The current issue and full text archive of this journal is available at
www.emeraldinsight.com/0263-5577.htm

Developing an agent-based workflow management system for collaborative product design

Ching-Jen Huang
Department of Industrial Engineering and Management, National Chinyi Institute of Technology, Taiping, Taiwan, Republic of China

Amy J.C. Trappey
Department of Industrial Engineering and Engineering Management, National Tsing Hwa University, Hsinchu, Taiwan, Republic of China, and

Yin-Ho Yao
Department of Industrial Engineering and Management, Ta Hwa Institute of Technology, Hsinchu, Taiwan, Republic of China

Abstract

Purpose – The purpose of this research is to develop a prototype of agent-based intelligent workflow system for product design collaboration in a distributed network environment.

Design/methodology/approach – This research separates the collaborative workflow enactment mechanisms from the collaborative workflow building tools for flexible workflow management. Applying the XML/RDF (resource description framework) ontology schema, workflow logic is described in a standard representation. Lastly, a case study in collaborative system-on-chip (SoC) design is depicted to demonstrate the agent-based workflow system for the design collaboration on the web.

Findings – Agent technology can overcome the difficulty of interoperability in cross-platform, distributed environment with standard RDF data schema. Control and update of workflow functions become flexible and versatile by simply modifying agent reasoning and behaviors.

Research limitations/implications – When business partners want to collaborate, how to integrate agents in different workflows becomes a critical issues.

Practical implications – Agent technology can facilitate design cooperation and teamwork communication in a collaborative, transparent product development environment.

Originality/value – This research establishes generalized flow logic RDF models and an agent-based intelligent workflow management system, called AWfMS, based on the RDF schema of workflow definition. AWfMS minimizes barriers in the distributed design process and hence increases design cooperations among partners.

Keywords Work flow, Intelligent agents, Product design, Product design systems

Paper type Research paper

1. Introduction

With the development of internet technology, competitive pressures are forcing companies to consider strategies to reduce costs and compress time between each stage of the value chain (Humphreys *et al.*, 2005). The concept of product design collaboration over the internet has been widely adopted to reduce time-to-market,

This research is partially supported by the National Science Council and ITRI research grants, which are greatly appreciated by the authors. The authors also wish to thank W-J. Tzeng's technical supports in applying JADE for AWfMS implementation.

Industrial Management & Data Systems
Vol. 106 No. 5, 2006
pp. 680-699
© Emerald Group Publishing Limited
0263-5577
DOI 10.1108/02635570610666449

project's life cycle and to take requirement into consideration for competitive advantages (Bouchlaghem *et al.*, 2004). Integration and collaboration are a common term in the enterprise systems (Gulledge, 2006). In collaborative product design, each team member must be responsible for one or more works and contribute what they can in different domains of expertise at various stages to overcome the major weakness of traditional face-to-face communication collaboration (Cheng *et al.*, 2006). From the perspective of supporting collaborative activities, research on the collaborative design can be divided into two main purposes, i.e. to assist users solve complex problems intelligently and to mediate cooperative activities intelligently. These researches are categorized into the domain of computer supported cooperative work (CSCW), where workflow systems play a critical role (Aknine, 1998). These workflow systems constitute the coordinate mechanisms for a collaborative design environment with the benefits of flexible process definition, easy tracking of activities, and effective process management.

Recently, the term "agent" has been widely used to build computer support for works with collaborative design nature (Trappey *et al.*, 2004). The integration of workflow management with agent technology has attracted attention of researchers in the related area. Researchers apply agent technology in the workflow system to achieve the benefits of autonomy, social ability, pro-activeness, and reactivity (Xia and Li, 1999; Kuo, 2004). Further, using software agent technology to provide flexible and autonomous solutions for workflow management is proposed by Yan *et al.* (2001) for business process coordination. However, current workflow management technologies have difficulties in solving the challenges of collaborative product design in a distributed environment with dynamic nature of product development, distributed knowledge and resources, and risks attached to design collaboration (Xu and Wang, 2002). The risks of collaboration occur significantly when an organization losses direct control over the product development process, incomplete information is disclosed with lack of transparency (Parker, 2000). To solve the problems, one of the major issues is to deal with the interoperability and communication between agents and agent-based platforms. The cooperation and coordination among agents becomes the most important issue in agent systems research (Xia and Li, 1999).

In this research, we propose an agent-based workflow management system (AWfMS) with RDF (resource description framework) ontology schema for collaborative design flow definition. To enhance cooperation and coordination among agents, we constructed our work with two main focuses. First, we define a generic workflow metadata model (i.e. data schema) for the development of collaboration functionalities and for the interoperability between agents platforms. This led us to design a standardized flow logic by using XML/RDF ontology schema. On the other hand, we define agent-based workflow architecture using the workflow data schema to perform the desired functions. The purposes of this research is to establish an agent-based intelligent workflow system, which applies the RDF ontology schema as a common representation of workflow logics and elements to enable design chains cooperation, design knowledge reuse and coordination in real time. Further, a case study in collaborative system-on-chip (SoC) design is presented to demonstrate how the AWfMS works in the domain of integrated circuit (IC) design to bring benefits to IC corporations and project managers.

The remainder of this paper is organized as followings. Section 2 reviews the background and previous literature. Section 3 defines the agent-based workflow system architecture in details. Section 4 uses a SoC collaborative design case as an example to demonstrate AWfMS collaboration at work. The last section concludes our contributions and addresses suggestions for future works.

2. Literature review

In this section, we give an overview of related applications of agent technology, present a brief description of current research works on AWfMS, and discuss current limitations and our intended break through in agent-based workflow management for product design collaboration. Finally, we describe XML/RDF schema for enabling structural and semantic description of workflow template.

2.1 Related applications of agent technologies

An agent is a software that can autonomously perform routine tasks with a degree of intelligence (Turban *et al.*, 2004). Agents can be applied to filter data, interpret information, monitor activities, decision support, etc. Agents in application domains typically possess four key characteristics, i.e. autonomy, reactivity, communication and goal driven. They are capable of acting autonomously, cooperatively, and collectively. Parker (2000) described the risks of product development using collaborative mode. The main risks include loss of direct control over the product development process, lack of transparency, incomplete information disclosure, ambiguous roles/responsibilities specification and allocation. Agent technologies are intended to reduce risks of collaborations for concurrent, distributed and collaborative information processing, particularly for the concurrent and distributed design and manufacturing (Tang, 2004). Overall speaking, there are various applications of agent technologies reported in many engineering fields in recent years as shown in Table I.

Table I.
Agent-based technology applications

Authors	Application areas
Chen *et al.* (2000)	Automate tasks in e-commerce applications
Anumba *et al.* (2002)	Collaborative structures design
Xu and Wang (2002)	B2B workflow monitoring
Mitkas *et al.* (2003)	Control concurrent engineering tasks
Beetz *et al.* (2004)	Support collaborative design in architecture domain
Domazet (2000)	Enable engineering collaboration in inter-enterprise design chains
Huang (2004)	Support modular product collaborative design
Jia *et al.* (2004)	Coordinated product development and manufacture
Liu *et al.* (2004)	Dynamically create and manage design tasks in widely distributed and ever-changing design environments
Liu (1998)	Support strategic environmental scanning and interpretation
Shih *et al.* (2002)	Recommendation systems in e-commerce applications
Sugumaran and Bose (1999)	Support complex data analysis, data mining and decision-making activities
Tang (2004)	Integrate die-marker's activities into customer product development process
Trappey *et al.* (2004)	Support online global logistics management
Trappey *et al.* (2006)	Automate SIP online trades
Ying and Dayong (2005)	3PL in e-commerce
Zha (2002)	Concurrent intelligent design and assembly planning

Jia *et al.* (2004) presented an adaptive and upgradeable agent-based system for coordinated product development and manufacturing. The system consists of a central managing agent and several other functional agents such as the manufacturability evaluation agent, process-planning agent, and scheduling agent. Xu and Wang (2002) proposed a collaborative multi-agent system for B2B workflow monitoring. Further, Anumba *et al.* (2002) discussed the use of intelligent agents in the collaborative design of structures. Domazet (2000) discussed the use of collaborative agents to enable engineering collaboration in inter-enterprise design chains. In specific domain application, Tang (2004) developed a multi-agent-based system to integrate die-maker's activities into customer product development process within a distributed, collaborative and concurrent environment. Chen *et al.* (2000) applied agent technology for automating tasks in e-commerce applications. Shih *et al.* (2002) also proposed a generic mobile agent framework for recommendation applications in e-commerce to help merchants to make suitable decisions and provide personalized information to customers. Sugumaran and Bose (1999) applied intelligent software agent to support complex data analysis, data mining and decision-making activities. Liu (1998) developed an agent-based strategic scanning and interpretation system in the context of strategic management and organizational learning. Liu's system consists of three types of software agents, i.e. the information resource agent, the field intelligent agent and the strategic issue attendant. In the logistics management, Trappey *et al.* (2004) proposed a mobile-agent online order-tracing system, which enables the real-time tracking of logistic services in the network without causing much internet loading. Ying and Dayong (2005) also presented a multi-agent framework for third party logistics (3PL) in e-commerce. Zha (2002) developed a knowledge intensive multi-agent framework for concurrent intelligent design and assembly planning. Beetz *et al.* (2004) proposed a framework for multi-agent system for the support of collaborative design in the architectural domain. Huang (2004) presented an agent-based system to support modular product collaborative design. Mitkas *et al.* (2003) discussed how software agent technology can be used to control a large number of concurrent engineering tasks and present a multi-agent development framework for constructing multi-agent systems. Liu *et al.* (2004) proposed a multi-agent collaborative design system for dynamically creating and managing design tasks in widely distributed and ever-changing design environments. Trappey *et al.* (2006) applied mobile agent technology for the enabling of silicon intellectual property (SIP) knowledge trade mechanisms on the internet.

2.2 Related works on agent-based workflow management systems

The workflow management coalition (WfMC) defines workflow management as:

> The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules (WfMC, 1999).

Workflow management system (WfMS) is a software application that supports specifications and executions of workflow. Workflow design is very much related to the formulation and analysis of the value chain activities (Lau *et al.*, 2003). WfMS provides the:

> ... procedural automation of a business process by management of sequence of work activities and the invocation of appropriate human and/or IT resources associated with various activity steps (Hollingsworth, 1995).

It contains a set of tools providing supports for the necessary services of workflow enactment services, process definition, administrative and monitoring tasks, workflow client applications and other invoked applications.

An AWfMS is a set of software agents that managing and coordinating the flow of works through a business process (Odgers *et al.*, 1999). From this viewpoint, an agent-based system is a distributed and collaborative system, not a stand-alone knowledge-based system (Liu, 1998). The system aims to improve process integration, interoperability, reusability and adaptability. Using agent to perform WfMS functions is an emerging research field in recent years. Many researchers have been exploring the power of agent technology to provide flexible, distributed, and intelligent solutions to the current WfMS. Madhusudan (2005) proposed an agent-based approach for coordinating product design workflows. The agent-based approach enables facilitating design process knowledge reuse and supporting distributed dynamic process management. Yan *et al.* (2001) discussed the forms and benefits of integrating workflow and agent technology for business process management. Huang *et al.* (2000) proposed workflow management as a mechanism to facilitate the teamwork in a collaborative product development environment. Savarimuthu *et al.* (2004) described the monitoring and controlling aspects of agents that are embedded to an adaptive and distributed agent-based workflow systems. Savarimuthu *et al.* (2005) developed a multi-agent-based workflow system that supports distributed process models and the adaptability of executing processes. Stormer and Knorr (2001) described the use of PDAs for the execution of workflow tasks and presented a prototype implementation based on software agents.

The essential features of the above agent-based workflow systems are autonomous, and dynamic collaboration and communication. Agents must cooperate, communicate and negotiate with other agents for coordinating and controlling the flow of works and executing any tasks of the workflow from various locations. In order for agents to communicate effectively and enhance interoperability, they need to have mutually understandable and standard semantic constructs. Using semantic web to design intelligent agents has the following advantages, e.g. easy to understand, easy resource integration, and resource reuse (Turban *et al.* 2004). Despite the growing number of multi-agent workflow systems have been prototyped, still no research presented the semantic construct approach, such as RDF schema, for standard data modeling and development of agent-based WfMS for inter-operability among web-based application suites.

2.3 Resource description framework standard

The RDF is a semantic construct framework describing and interchanging metadata for the web application. It is developed by the World Wide Web Consortium (W3C) and provides the foundation for metadata interoperability across different resource description communities (W3C, 2005). The objective of RDF is to enhance the interoperability of metadata. It is a tool that provides a content presentation and organization standard so that content can be shared safely among different software applications (Turban *et al.*, 2004). Extensible markup language (XML) provides structural and semantic information and offers a standard approach for describing, capturing, processing information on web. This paper integrates the RDF/XML concept into agent-based WfMS. The workflow template and message could be organized and modeled by RDF/XML schema standard. With the well-defined standard format, knowledge on the web is easier for software agents to access, understand, and share.

3. Agent-based collaborative system framework

In this research, we develop an AWfMS for collaborative design environment as shown in Figure 1. The system operates based on a three-tier system architecture for security data communication and control including two main mechanisms, i.e. workflow management mechanism (WfMM) and agent communication mechanism (ACM). The WfMM is the WfMS kernel following the specification of WfMC workflow reference model (Hollingsworth, 1995). It includes a workflow user interface (UI) and four main sub-modules, i.e:

(1) system maintenance module;

(2) workflow monitoring and control module;

(3) workflow maintenance module; and

(4) workflow execution module.

The WfMM server uses WfMM agent group to communicate with ACM server. The ACM server executes the workflow instance and communicates with other agent-based workflow system modules.

3.1 Three-tier AWfMS architecture

Generally speaking, system architecture can be defined as 2-tier, 3-tier or *n*-tier model. The traditional client/server architecture is based on a 2-tier model, user layer and application layer with built-in database. Other than handling the graphical interface and the interactions with users, the user layer have to bear the load of data manipulation and logic calculation. This model will cause the client site to carry a

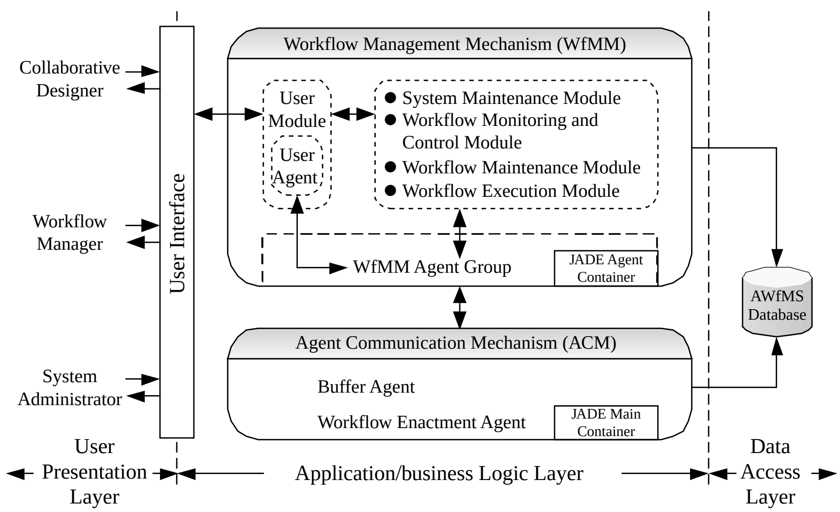

Figure 1.
AWfMS collaborative system framework

heavy load and occupy more system resources; therefore, reduce the program execution performance. In this research, AWfMS applies 3-tier system architecture to increase the system performance, security and maintainability. AWfMS is a type of distributed workflow system, as described by Li and Shi (1999). Thus, workflow engineers can explain and execute processes on different sites.

3.1.1 User presentation layer. User presentation layer is responsible for the presentation of data, receiving user events, controlling the UI, and interactive with the user. For interoperability consideration, this research develops a JAVA-based graphic user interface (GUI). The client site can just use browser to communicate with the applications without any additional hardware/software installation. When users login in the system, the system will present different UI based on the user's role and authority, such as collaborative designer UI, workflow manager UI, and system administrator UI.

3.1.2 Application/business logic layer. The application/business logic layer mainly handles the business objects with pre-defined functional purposes in order to proceed the data manipulation and logic operation. Functional objects are encapsulated with methods and data logics for the usage of user presentation layer. It protects the data from direct access by the clients. In this research, the application objects mainly provide services for the enabling of WfMM and ACM.

3.1.3 Data access layer. The data access layer is responsible for data storage, including workflow template data, workflow and system execution data, and users data. The data access layer can only be accessed through the application logic layer with user authentication checking. It is non-transparent to users to prevent unauthorized data access.

3.2 Workflow management mechanism

The WfMM constructed by following the WfMC workflow reference model, including components such as workflow engine, process definition tools, administrative and monitoring tools, and client application. Table II shows the system module, agents and related workflow components. The WfMM is divided into five main system modules. They are:

(1) system maintenance module;
(2) workflow monitoring and control module;
(3) workflow maintenance module;
(4) workflow execution module; and
(5) user module.

Each module has its corresponding agent to execute the functional behavior.

The system maintenance module is performed by system maintenance agent (SMA). The SMA has two main functions, i.e. user management and function-authority management. In user management function, SMA can add, modify, and delete a user. In function-authority management, SMA will define corresponding function depending on the user's role. The system defines three users roles, i.e. system administrator (super user), workflow manager (workflow template designer/manager), and collaborative designer (end-users). System administrator has the highest authority and is permitted to perform all system functions. A workflow manager can define and modify a workflow and the workflow's task details. Finally, collaborative designer (end-users) are allowed to perform the functions of modify personal data, show work list, and execute designated task.

The workflow monitoring and control module is performed by workflow monitoring and control agent (WfM&CA). The WfM&CA tracks the real-time status of flow progress, e.g. start, suspend, resume, abort, and query. Figure 2 shows the task status and status linkage diagram in the system.

The workflow maintenance module (agent) can add, modify, delete, and query any workflow template and its associating tasks. It includes three main functions, i.e. design task, design workflow template, and download workflow RDF definition. The "design task" function can add, modify, and delete a task. The "design workflow template" function can add, modify, change version, and delete a workflow template by putting associating tasks into the flow template. The "download workflow RDF" function supports the RDF document management. The user module (agent) can modify personal data, show personal work list waiting for execution, and highlight the delayed tasks to corresponding user. Finally, the workflow execution module uses the workflow execution agent (WfEA) to pass the attributes, which outputted by tasks, to the buffer agent (BA) and workflow enactment agent (WfEnA) at ACM server for sub-sequential tasks.

3.3 Workflow metadata model

Essentially speaking, workflow is an automatically running process among multi-participants to transfer information or tasks according to some previous

System module	Agent type	Workflow component
System maintenance module	System maintenance agent	Administration and monitoring tools
Workflow monitoring and control module	Workflow monitoring and control agent	Administration and monitoring tools
Workflow maintenance module	Workflow maintenance agent	Process definition tools
Workflow execution module	Workflow execution agent Buffer agent Workflow enactment agent	Workflow engine/client application
User module	User agent	UI tools

Table II. The corresponding system modules, agent types and workflow components

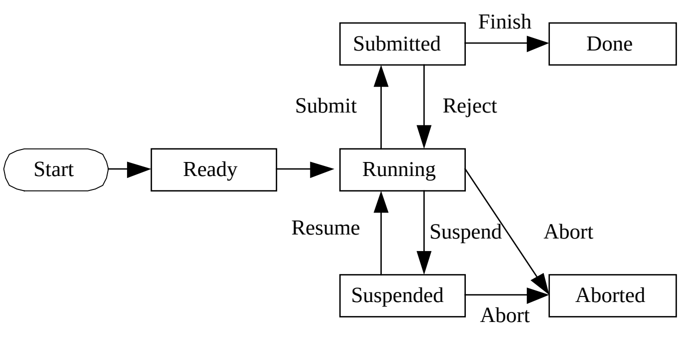

Figure 2. Task status and linkage diagram

defined rules or sequences. The integration and cascade of tasks formulate a typical workflow model, and thus, form the basis of WfMS. Figure 3 shows the ontology of workflow model. The involved objects in this model are workflow template, workflow instance, task, task instance, role, rule, application, message, and document.

The workflow metadata model can essentially and effectively describe the information relevant to a workflow, such as pre/post conditions for the operation of a task, data structure, control parameters, invoked application and operational modes. This information will be applied by related agent models. The AWfMS RDF metadata schema is shown in Figure 4.

3.4 Agent architecture

In the AWfMS framework, we propose a system constructed with intelligent multi-agent support in a distributed collaborative design environment. The system works on JADE agent platform, respectively. The agent hierarchy of AWfMS is shown in Figure 5. The proposed system consists of a number of agents, which operate autonomously and cooperate with each other to accomplish their predefined goals.

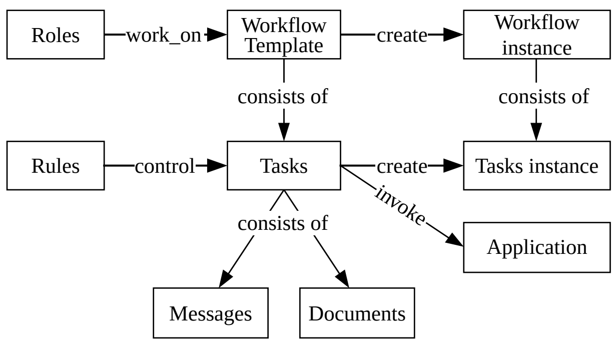

Figure 3. Workflow model's ontology

```
<!ELEMENT workflow ( wf_id, task_data, yes_to, no_to, application ) >
<!ELEMENT wf_id (#PCDATA)>
<!ELEMENT task_data (task_id, receiver, dueday, task_desc, message, document)>
<!ELEMENT yes_to (#PCDATA)>
<!ELEMENT no_to (#PCDATA)>
<!ELEMENT task_id (#PCDATA)>
<!ELEMENT receiver (#PCDATA)>
<!ELEMENT dueday (#PCDATA)>
<!ELEMENT task_desc (#PCDATA)>
<!ELEMENT message (MsgID, msg)>
<!ELEMENT MsgID (#PCDATA)>
<!ELEMENT Msg (#PCDATA)>
<!ELEMENT document (DocID, doc_desc, doc_path)>
<!ELEMENT DocID (#PCDATA)>
<!ELEMENT doc_desc (#PCDATA)>
<!ELEMENT doc_path (#PCDATA)>
<!ELEMENT application (app_name, app_parameter, app_path)>
<!ELEMENT app_name (#PCDATA)>
<!ELEMENT app_parameter (#PCDATA)>
<!ELEMENT app_path (#PCDATA)>
```

Figure 4. Workflow metadata schema

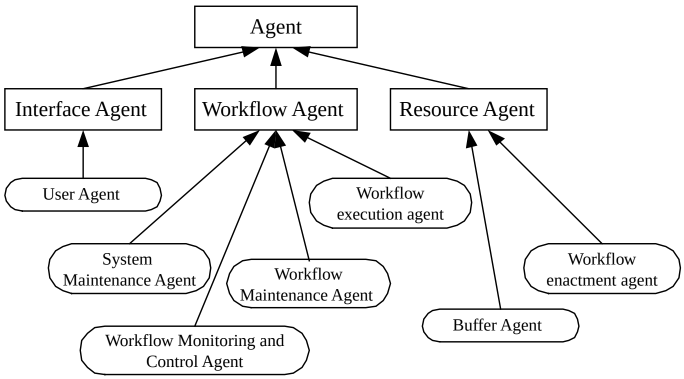

Figure 5.
The agent hierarchy in three sub-groups

In our architecture, the intelligent agents can be classified into three sub-groups according to their functional hierarchy:

(1) Interface agent is associated with a user in an organization. Its function is to manage human resource, such as access user's authority data and unfinished jobs. User agent (UA) is a type of interface agent.

(2) Workflow agent handles process definition, controls a process instance (including create, activate, suspend, abort, etc.), maintains control data of workflow and supervises a workflow execution. They include WfM&CA, SMA, workflow maintenance agent (WfMA) and WfEA.

(3) Resource agent is associated with the other resource except human resource. WfEnA and BA are the types of resource agent.

3.4.1 The behavior model of workflow monitoring and control agent. When a user with the role of collaborative designer login in the system successfully, system will automatically create a UA containing the basic data affiliating with this user. The user can then modify personal data through UA. In the meantime, the system will trigger workflow monitoring and control module, and then the module will automatically generate a WfM&CA. Figure 6 shows the life cycle and behavior model of WfM&CA. The WfM&CA perform tasks as follows:

- After successfully acquired the basic data of the user from UA, system will automatically trigger workflow monitoring and control module, which will create a WfM&CA.
- The UA will forward the user's basic data to WfM&CA.
- WfM&CA will make a request to the WfMM server for workflow data. The WfMM server finds the user's related workflow data and sends them to WfM&CA.
- WfM&CA then forward the data to UA.
- UA will show the workflow data as a work list for the specific user. After the task completetion, WfM&CA will be eliminated from the system.

3.4.2 The behavior model of system maintenance agent and workflow maintenance agent. When a user as the system administrator logins to the system successfully,

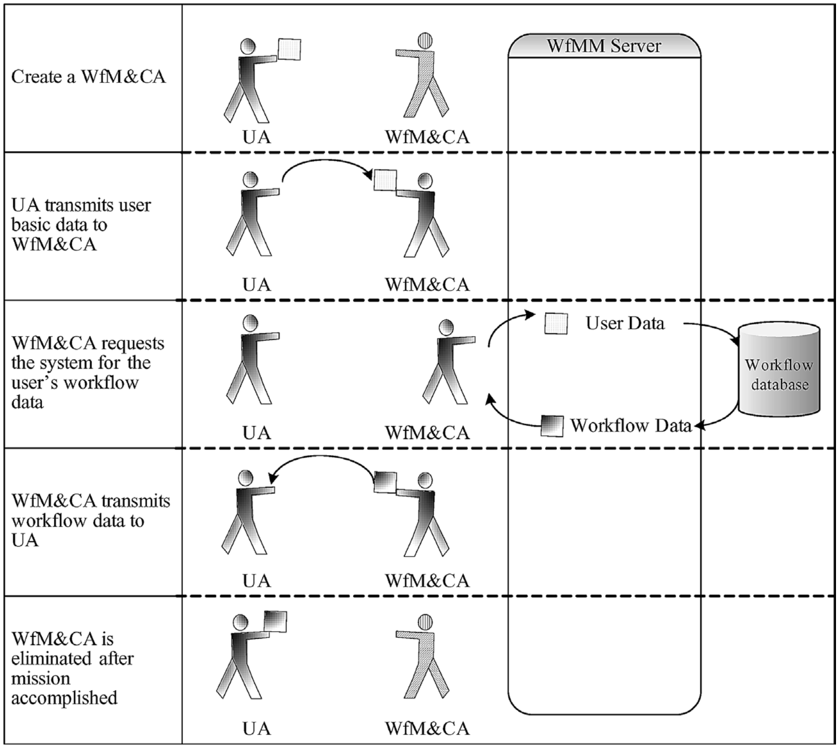

Figure 6.
The lifecycle behavior model of WfM&C Agent

system will automatically create a system administrator's UA. Instead of interacting with WfM&CA, UA will interact with WfMA or SMA depending on what the desired function is invoked. Figure 7 shows the life cycles and behavior model of WfMA and SMA. The tasks are performed sequentially as follows:

- After UA successfully sends workflow maintenance requests or system maintenance requests to WfMM server, system will automatically trigger workflow maintenance module or system maintenance module with WfMA or SMA up running.
- The UA passes user's requests to WfMA or SMA.
- WfMA or SMA requests the system for workflow or system's maintenance status.
- After acquiring the data, WfMA or SMA will transmit the data to UA.
- UA displays the data in a proper form at through UI. Meantime, WfMA or SMA will be eliminated from the system.

A workflow usually consists of many tasks with logical order for execution. Every task is corresponding to a single WfEA. WfEA holds the task status and deals with inputs and outputs of all task attributes. When a task is running, WfEA acquire the output attributes of the preceding task as input attributes of current task. When a task is

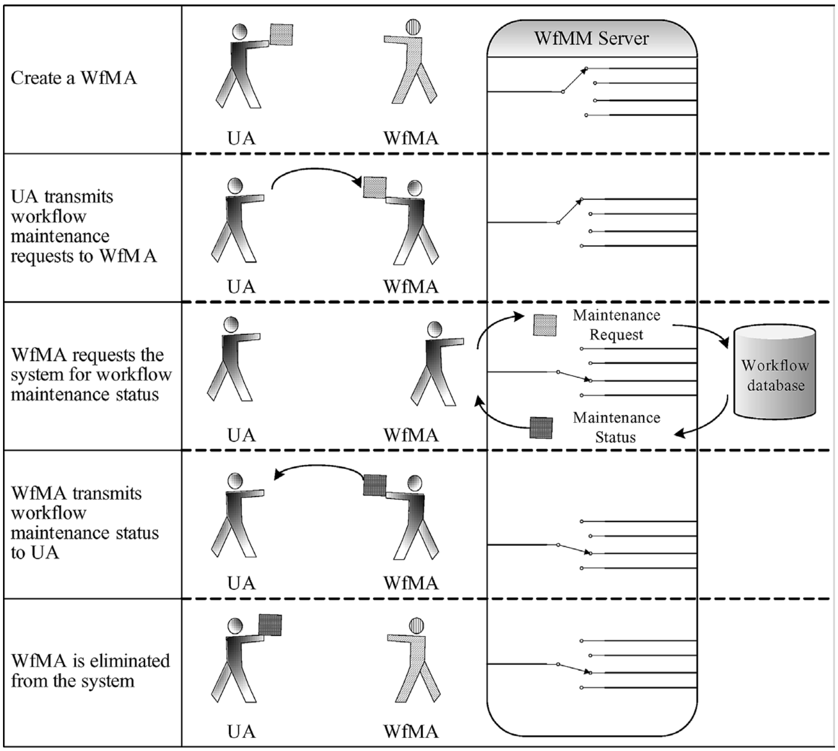

Figure 7.
The life-cycles behavior model of WfMA and SMA

submitted, WfEA of current task delivers output attributes to WfEA of the following task or BA awaiting for another parallel task's outputs. If WfEA delivers output attributes to the BA, which means there is a non-sequential workflow, which needs multi-inputs from multiple WfEAs of the previous tasks (Figure 8).

When BA in the ACM server acquires complete attributes according to workflow definition, it will deliver the attributes to a WfEnA. The WfEnA will process these

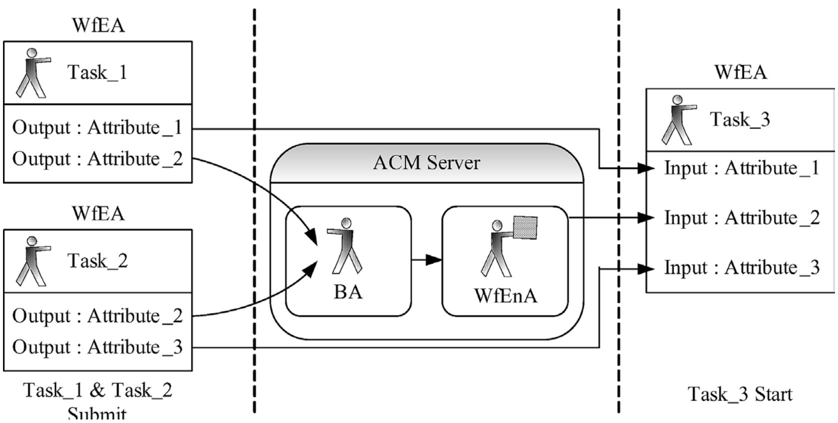

Figure 8.
The behavior model of WfEA, BA, and WfEnA

attributes using its own rules and deliver final results to WfEA of the following task. If a BA does not receive complete attributes before the due date, it will automatically send default attributes or remind responsible users by using e-mails.

4. Case study

In this section, we present a real case of SoC collaborative design scenario as an agent-based workflow application. The design workflow takes place in SoC design is demonstrated with multiple agent collaborative support. Finally, the conceptual framework is developed and the prototype (AWfMS platform) is implemented for the complex SoC collaborative design processes.

4.1 Industrial SoC collaborative design scenario

Industrial SoC collaborative design is a typical case of product collaborative design. SoC design project is a complicated process, which needs a number of people with complementary skills working as a team to achieve common objectives, such as reducing cycle times and lowering product cost. All participants contribute their expertise in different domains at various stages to integrate multiple functionalities on a single chip.

Traditionally, under the consideration of R&D expense and technology capability, the design company usually purchase or outsource some design know-hows, such as intellectual property (IP), from alliances or IP mall. Therefore, engineers of IP providers usually involve in new SoC design projects of their clients. The SoC design team, which may be in different geographical locations across the continents, must get the required IP data to proceed the design tasks. These scenarios make the design processes conducted work on a distributed environment over the internet.

When an industrial SoC design process is defined, we can translate it into a design workflow diagram, as shown in Figure 9. The diagram shows the detailed collaborative workflow tasks and their corresponding team members. The design team consists of project manager, IP provider, analog IC designer, digital IC designer and layout engineer. Based on the diagram, the workflow designer can establish a workflow template. Table III highlights the definition of workflow template and an instance of SoC design task. The description of the flow details is stored in the database server to be used by the agents.

4.2 SoC collaborative design AWfMS framework and implementation

Following the agent-based approach, a prototype WfMS for SoC product design collaboration has been developed. Figure 10 shows the conceptual model implemented in the case of industrial SoC collaborative design. The 3-tier architecture is implemented in the SoC product developer's site. The system includes two servers, namely WfMM server, working as a WfMM, and ACM server, acting as ACM. All clients access the design data from the servers. This ACM agent platform consists of a JADE main container where Message Transport System, Agent Management System (AMS) and Directory Facilitator (DF) reside. The AMS manages the life cycles of the agents such as starting and stopping agents. Every machine has its own JADE agent container built on its Java Virtual Machine (JVM) in order to manage the agents for specific behavioral mechanisms. The JVM provides a complete run time environment for agent execution and allows agents to execute on the same host concurrently.

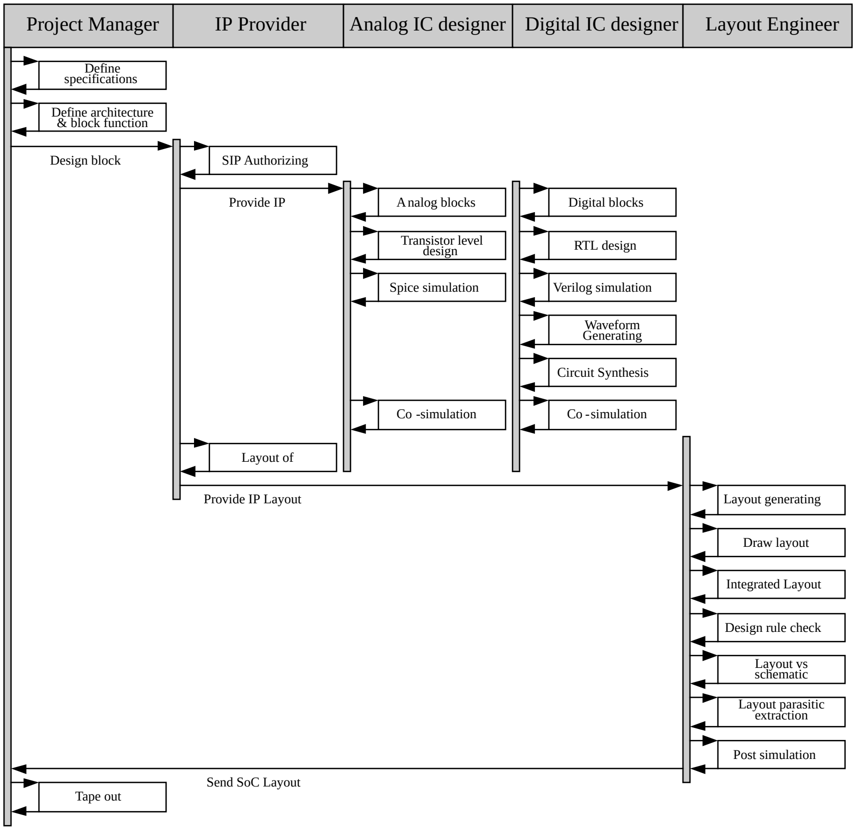

Figure 9.
A SoC design workflow diagram

Every JADE agent container will register itself to the Remote Method Invocation in JADE main container. Therefore, agent finds any other agent in this agent platform through DF in the JADE main container.

As shown in the conceptual framework, the primary responsibilities of the SoC design project manager are to submit request, review design specification and confirm design specification. IP provider staff is in charge of designing product based on design company request, verifying product integration and redesigning product. The SoC engineer is in charge of designing product, evaluating design request and confirming design specification. With the implemented AWfMS, it allows the design team members to collect, merge and response the requests coming from other members interactively and cooperatively. The agent-based system enables collaboration between design team and facilitates concurrent development and easy integration of IPs with SoC design.

4.3 Workflow and agents cooperation for starting a new SoC design

Figure 11 shows the sequence diagram for a workflow execution and the internal interactions between agents for starting a new SoC design job. The operation details and messages exchange of these agents are illustrated as follows.

Elements Level 1	Level 2	Level 3	Level 4	Value	Description
Workflow				SoC	The workflow for SoC design
	wf_id			SoC01	SoC design workflow ID
	task_data				A task in SoC design workflow
		task_id		SoCT07	SoC design task ID
		receiver		Steven	Participant user of SoC design task
		dueday		3	Due day of SoC design task
		task_desc		RTL design	Description of SoC design task
		message			Message for SoC design task
			msg_id	SoCM05	Message ID
			msg		Message content
		document			Document for SoC design task
			doc_id	SoCD02	Document ID
			doc_desc	Notes for RTL design	a brief notation for SoC RTL design
			doc_path	\\SoC\F:\SocD02.doc	file location of the document
	yes_to			SoCT08	To next job
	no_to			SoCT06	To previous job
	application				Invoked application for SoC design
		app_name		SoCP03	Application name for SoC design
		app_para		layout11, Task11	parameters for SoC design
		app_path		\\SoC\F:\SoCCAD.exe	Application path for SoC design

Table III.
An example of workflow task element for SoC design

The process workflow starts with internal or external customers' requests to SoC design. When a user of design team logins successfully, the system creates a UA and gives to the UA the user's information. The UA then creates a WfM&CA and sends the user data to WfM&CA. The WfM&CA then gets the workflow RDF definition data, accesses the jobs waiting for execution and displays jobs list to the UA. Afterward, the user selects the assigned job to execute from the job list. UA will send the data to WfEA subsequently for WfEA to start executing the selected job. When WfEA finishes the job, it will send all the relevant results to BA. If there is no any parallel job waiting for execution, BA will send all the job data to WfEnA and WfEnA will activate the next task of the workflow instance.

Figure 12 show the AWfMS web interfaces of SoC collaborative design workflow. When a collaborative designer logins the system, the system will display the designated work list for the user. Figure 12(A) shows the task definition of the entire SoC design workflow. In real time, a project leader can review all statuses of task-execution as shown in Figure 12(B). The system can also provide statistical analysis of execution performance as shown in Figure 12(C).

4.4 The managerial implications of AWfMS

The case study has demonstrated AWfMS for complex SoC design with collaborative team efforts. For the design team to work closely following a strict time constraint,

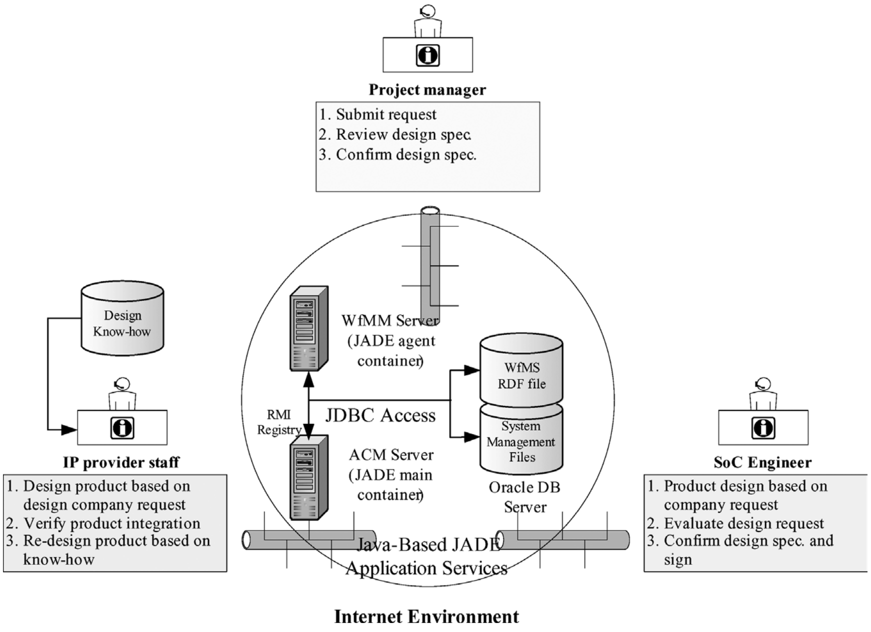

Figure 10. Conceptual framework of SoC collaborative design AWfMS

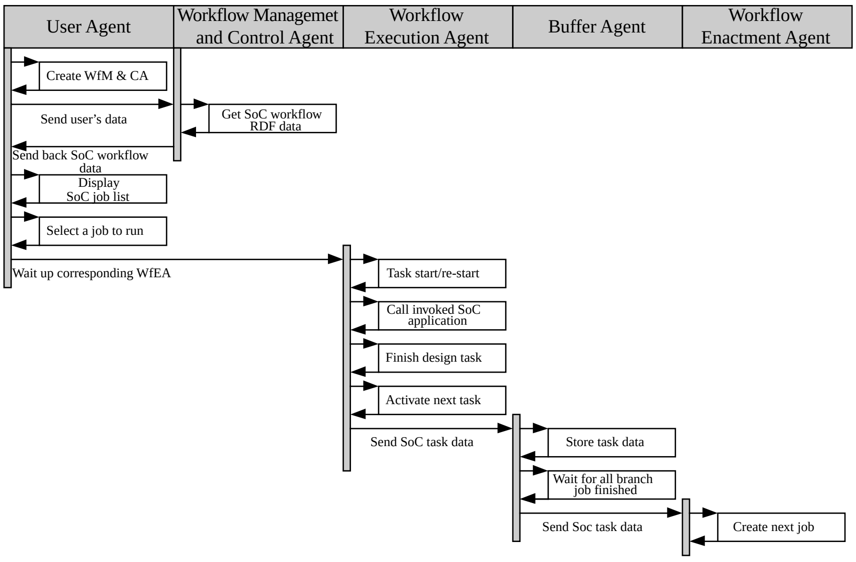

Figure 11. Sequence diagram of agents cooperation for starting a new Soc design job

effective communication, design knowledge sharing/reusing, dynamic control of the whole design progress, and identification of discrete contribution of individual participants are critical issues. The AWfMS implementation provides a promising solution to handle these issues. AWfMS also demonstrates benefits from the perspectives of design chain participants and managers. These benefits are enabled

(A)

Task ID	Running Manpower	Required Document	Required Resource
Define Specifications	admin	No	No
Tape Out	Engineer19	VerificationList	No
Define Architecture	Engineer1	No	Xilinx
Digital Blocks	Engineer2	No	Altera
Analog Blocks	Engineer3	No	No
SIP Authorizing	Engineer4	No	Altera
Verilog Programming	Engineer5	SocSpecification	No
Draw Circuit Schematic	Engineer6	SocSpecification	No
Verilog Simulation	Engineer7	SocSpecification	Cadence
Spice Simulation	Engineer8	SocSpecification	Xilinx
Waveform Generating	Engineer9	VHDLCode	HSPICE
Circuit Synthesis	Engineer10	VHDLCode	HSPICE
Co-Simulation	Engineer11	VHDLCode	Dracula
Draw Layout	Engineer12	CaseList	Xilinx
Intergrated Layout	Engineer16	SocSpecification	No
DRC	Engineer17	CaseList	HSPICE
LPE	Engineer18	VerificationList	HSPICE, Cadence

(B)

Task ID	Running Status	Running Manpower	Time	Cost	Factor
Define Specifications	Processing	admin	0	0	No Data
Define Architecture	Not Start	Engineer1	Not Start	Not Start	No Data
Digital Blocks	Not Start	Engineer2	Not Start	Not Start	No Data
Analog Blocks	Not Start	Engineer3	Not Start	Not Start	No Data
SIP Authorizing	Not Start	Engineer4	Not Start	Not Start	No Data
Verilog Programming	Not Start	Engineer5	Not Start	Not Start	No Data
Draw Circuit Schematic	Not Start	Engineer6	Not Start	Not Start	No Data
Verilog Simulation	Not Start	Engineer7	Not Start	Not Start	No Data
Spice Simulation	Not Start	Engineer8	Not Start	Not Start	No Data
Waveform Generating	Not Start	Engineer9	Not Start	Not Start	No Data
Circuit Synthesis	Not Start	Engineer10	Not Start	Not Start	No Data
Co-Simulation	Not Start	Engineer11	Not Start	Not Start	No Data
Draw Layout	Not Start	Engineer12	Not Start	Not Start	No Data
Intergrated Layout	Not Start	Engineer16	Not Start	Not Start	No Data
DRC	Not Start	Engineer17	Not Start	Not Start	No Data
LPE	Not Start	Engineer18	Not Start	Not Start	No Data
Tape Out	Not Start	Engineer19	Not Start	Not Start	No Data

(C)

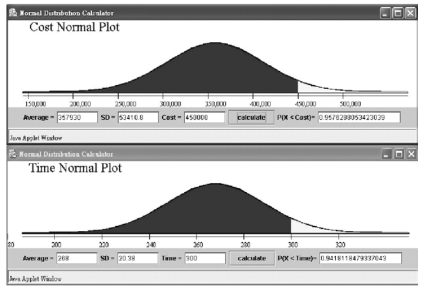

Figure 12.
An example of SoC design flow showing (A) tasks of workflow; (B) status report of task-execution and (C) flow performance analysis

due to the continuous monitoring, control and update of design knowledge and outcome by agents enacting their missions in AWfMS. Thus, the benefits of implementing AWfMS, from a manager's point of view, are summarized as follows.

- manage design workflow dynamically and flexibly in dispersed locations;
- solve interoperability problems for product design knowledge and job sharing;
- enable communication between design partners in heterogeneous platforms and systems; and
- support performance auditing and evaluation of design team in real time.

5. Conclusion

As the rapid development of computer network and related technologies, many organizations perform their works on a distributed computing infrastructure based on internet/intranet, which provides a heterogeneous platform for workflow system application. One of the challenges on workflow system development is the enabling of high interoperability. A promising solution approach to resolve the issue is to integrate RDF/XML concept and agent technology to develop the workflow system. Based on the workflow ontology, this research develops a 3-tier AWfMS for collaborative teamwork of complex SoC product design. The major contribution in this paper is that we have designed a novel approach and system architecture of agent-based WfMS for collaborative product design. AWfMS applies agent technology to establish the workflow handling and information communicating mechanism in distributed environments. RDF ontology schema is used as the data modeling standard of AWfMS. In summary, the system has achieved the following advantages as a collaborative design workflow management tool:

- facilitate design cooperation and teamwork communication in a collaborative, transparent product development environment;
- enable design process/knowledge sharing and reuse;

- overcome the difficulty of interoperability in cross-platform, distributed environment with standard RDF data schema; and
- control and update of workflow functions become flexible and versatile by simply modifying agent reasoning and behaviors.

References

Aknine, S. (1998), "Using multi-agent systems to support workflow telecommunications applications", *Proceedings of the 1998 IEEE International Symposium on Intelligent Control (ISIC)*, September 14-17, pp. 537-42.

Anumba, C.J., Ugwu, O.O., Newnham, L. and Thorpe, A. (2002), "Collaborative design of structures using intelligent agents", *Automation in Construction*, Vol. 11, pp. 89-103.

Beetz, J., Leeuwen, V. and Vries, B.D. (2004), "Towards a multi agent system for the support of collaborative design", in Leeuwe, J.P. and Timmermans, H.J.P. (Eds), *Developments in Design & Decision Support Systems in Architecture and Urban Planning*, Eindhoven University of Technology, Eindhoven, pp. 269-80.

Bouchlaghem, D., Kimmance, A.G. and Anumba, C.J. (2004), "Integrating product and process information in the construction sector", *Industrial Management & Data Systems*, Vol. 104 No. 3, pp. 218-33.

Chen, Q., Dayal, U., Hsu, M. and Griss, M. (2000), "Dynamic agents, workflow and XML for e-commerce automation", available at: http://xml.coverpages.org/qchen_EC2000.pdf

Cheng, E.W.L., Love, P.E.D., Standing, C. and Gharavi, H. (2006), "Intention to e-collaborate: propagation of research propositions", *Industrial Management & Data Systems*, Vol. 106 No. 1, pp. 139-52.

Domazet, D. (2000), "Enabling engineering collaboration in inter-enterprise design chains using collaborative agents", SIMTech Technical Report.

Gulledge, T. (2006), "What is integration?", *Industrial Management & Data Systems*, Vol. 106 No. 1, pp. 5-20.

Hollingsworth, D. (1995), "The workflow reference model", *Workflow Management Coalition*.

Huang, C.C. (2004), "A multi-agent approach to collaborative design of modular products", *Concurrent Engineering: Research and Applications*, Vol. 2 No. 1, pp. 39-47.

Huang, G.Q., Huang, J. and Mak, K.L. (2000), "Agent-based workflow management in collaborative product development on the internet", *Computer-Aided Design*, Vol. 32, pp. 133-44.

Humphreys, P., Huang, G. and Cadden, T. (2005), "A web-based supplier evaluation tool for the product development process", *Industrial Management & Data Systems*, Vol. 105 No. 2, pp. 147-63.

Jia, H.Z., Ong, S.K., Fuh, J.Y.H., Zhang, Y.F. and Nee, A.Y.C. (2004), "An adaptive and upgradable agent-based system for coordinated product development and manufacture", *Robotics and Computer-Integrated Manufacturing*, Vol. 20, pp. 79-90.

Kuo, J.Y. (2004), "A document-driven agent-based approach for business process management", *Information and Software Technology*, Vol. 46, pp. 373-82.

Lau, H.C.W., Lee, W.B. and Lau, P.K.H. (2003), "Flexible workflow integration: an object technology approach", *Industrial Management & Data Systems*, Vol. 103 No. 3, pp. 167-76.

Li, H. and Shi, M. (1999), "Application of agents in workflow management system", paper presented at APCC/OECC '99. Fifth Asia-Pacific Conference on Communications and Fourth Optoelectronics and Communications,Vol. 2, pp. 1068-72.

Liu, S. (1998), "Strategic scanning and interpretation revisiting: foundations for a software agent support system – Part 2. Scanning the business environment with software agents", *Industrial Management & Data Systems*, Vol. 98 No. 8, pp. 362-72.

Liu, H., Tang, M. and Frazer, J.H. (2004), "Supporting dynamic management in a multi-agent collaborative design system", *Advances in Engineering Software*, Vol. 35, pp. 493-502.

Madhusudan, T. (2005), "An agent-based approach for coordinating product design workflows", *Computers in Industry*, Vol. 56, pp. 235-59.

Mitkas, P.A., Symeonidis, A.L., Kehagias, D. and Athanasiadis, I. (2003), "Application of data mining and intelligent agent technologies to concurrent engineering", paper presented at the Tenth International Conference on Concurrent Engineering, Madeira.

Odgers, B.R., Shepherdson, J.W. and Thompson, S.G. (1999), "Distributed workflow co-ordination by proactive software agents", *Proceedings of Intelligent Workflow and Process Management, IJCAI-99 Workshop*, August.

Parker, H. (2000), "Interfirm collaboration and the new product development process", *Industrial Management and Data Systems*, Vol. 100 No. 6, pp. 255-60.

Savarimuthu, B.T.R., Purvis, M. and Cranefield, S. (2005), "Agent-based integration of web services with workflow management systems", The Information Science Discussion Paper Series, University of Otago, Dunedin.

Savarimuthu, B.T.R., Purvis, M. and Fleurke, M. (2004), "Monitoring and controlling of a multi-agent based workflow system", paper presented at the Australasian Workshop on Data Mining and Web Intelligence, Dunedin.

Shih, T.K., Chiu, C.-F., Hsu, H.-H. and Lin, F. (2002), "An integrated framework for recommendation systems in e-commerce", *Industrial Management & Data Systems*, Vol. 102 No. 8, pp. 417-31.

Stormer, H. and Knorr, K. (2001), "PDA – and agent-based execution of workflow tasks", available at: http://citeseer.ist.psu.edu/476011.html

Sugumaran, V. and Bose, R. (1999), "Data analysis and mining environment: a distributed intelligent agent technology application", *Industrial Management & Data Systems*, Vol. 99 No. 2, pp. 71-80.

Tang, D. (2004), "An agent-based collaborative design system to facilitate active die-maker involvement in stamping part design", *Computers in Industry*, Vol. 54, pp. 253-71.

Trappey, A.J.C., Trappey, C.V. and Lin, F.T.L. (2006), "Automated silicon intellectual property trade using mobile agent technology", *Robotics and CIM*, Vol. 22, pp. 189-202.

Trappey, A.J.C., Trappey, C.V., Hou, J-L. and Chen, B.J.G. (2004), "Mobile agent technology and application for online global logistic services", *Industrial Management & Data Systems*, Vol. 104 No. 2, pp. 169-83.

Turban, E., King, D., Lee, J. and Viehland, D. (2004), *Electronic Commerce: A Managerial Perspective*, Pearson Education International.

W3C (2005), "Resource description framework (RDF)", available at: www.w3.org/RDF/

WfMC (1999), "Workflow management coalition terminology & glossary", document number WfMC-tc-1011, issue 3.0.

Xia, M. and Li, H. (1999), "Cooperative software agents for workflow management system", paper presented at APCC/OECC '99. Fifth Asia-Pacific Conference on Communications and Fourth Optoelectronics and Communications,Vol. 2, pp. 1063-7.

Xu, D. and Wang, H. (2002), "Multi-agent collaboration for B2B workflow monitoring", *Knowledge-Based Systems*, Vol. 15, pp. 485-91.

Yan, Y., Maamar, Z. and Shen, W. (2001), "Integration of workflow and agent technology for business process management", paper presented at the Sixth International Conference on CSCW in Design, July 12-14, London, pp. 420-6.

Ying, W. and Dayong, S. (2005), "Multi-agent framework for third party logistics in e-commerce", *Expert Systems with Applications*, Vol. 29, pp. 431-6.

Zha, X.F. (2002), "A knowledge intensive multi-agent framework for cooperative/collaborative design modeling and decision support of assemblies", *Knowledge-Based Systems*, Vol. 15, pp. 493-506.

Further reading

Huang, C.J., Trappey, A.J.C. and Yao, Y.H. (2003a), "Design and develop WfMC-based modularized network collaborative design system", *Journal of Chinese Institute of Industrial Engineer*, Vol. 20 No. 4, pp. 422-34.

Huang, C.J., Trappey, A.J.C., Yao, Y.H., Yeh, S.C. and Cheng, Y.H. (2003b), "Adopting contact center approach to collaborative product development - using TFT-LCD design chain as case study", *International Journal of Electronic Business Management*, Vol. 1 No. 2, pp. 120-8.

Corresponding author

Ching-Jen Huang can be contacted at: cjhuang@cjhuang.idv.tw

To purchase reprints of this article please e-mail: **reprints@emeraldinsight.com**
Or visit our web site for further details: **www.emeraldinsight.com/reprints**