PAPER

An RFID-Based Manufacturing Control Framework for Loosely Coupled Distributed Manufacturing System Supporting Mass Customization

Ruey-Shun CHEN[†], *Nonmember*, **Yung-Shun TSAI**[††a)], *Student Member*, *and* **Arthur TU**[†††], *Nonmember*

SUMMARY In this study we propose a manufacturing control framework based on radio-frequency identification (RFID) technology and a distributed information system to construct a mass-customization production process in a loosely coupled shop-floor control environment. On the basis of this framework, we developed RFID middleware and an integrated information system for tracking and controlling the manufacturing process flow. A bicycle manufacturer was used to demonstrate the prototype system. The findings of this study were that the proposed framework can improve the visibility and traceability of the manufacturing process as well as enhance process quality control and real-time production pedigree access. Using this framework, an enterprise can easily integrate an RFID-based system into its manufacturing environment to facilitate mass customization and a just-in-time production model.

key words: *radio-frequency identification (RFID), mass customization, shop-floor control, production pedigree*

1. Introduction

Many manufacturing companies are currently facing severe global competition, a shorter life cycle of new products, and changing customer demands. Therefore, they must change their business operations to provide product variety and customization through flexibility and quick responsiveness [1], [2]. Thus, employing mass customization and a just-in-time (JIT) production model enables a firm to achieve this goal and survive in today's hypercompetitive environment [3].

To successfully achieve mass customization and the JIT production model, automatic identification technology (AIT) is one of the critical factors [4], [5]. Many manufacturing companies adapt AIT to manage manufacturing activities and take immediate action to resolve any events that disrupt production or cause customer dissatisfaction. For firms manufacturing high-priced and highly customized products, such as bicycles, a combination of AIT and a loosely coupled manufacturing process plays a key role in enabling flexible manufacturing [6].

Ubiquitous computing can be combined with existing information systems to create loosely coupled distributed systems that can be dynamically located and invoked to accomplish a complex manufacturing task [7], [8]. Even though methods that enable the implementation of ubiquitous computing applications have been proposed [9], they cannot effectively support the integration requirements in a ubiquitous and real-time environment. Issues related to radio-frequency identification (RFID) usage in ubiquitous computing applications have been widely studied [9]. RFID is a very important AIT. RFID provides the function of data carriers that can be read and written, thus it can record the identity and current status of a product as it is being manufactured [10]. Therefore, applying RFID to production systems will help manufacturing companies to realize mass customization and a JIT production model.

In this study we propose an RFID-based manufacturing control framework and a distributed information system to track and control the manufacturing process flow. Furthermore, a bicycle manufacturer is used to demonstrate the proposed system. This study revealed that the proposed system should improve the visibility and traceability of the manufacturing process. Additionally, this system should also enhance process quality control and real-time production pedigree access. Ultimately, an enterprise should facilitate mass customization and the JIT production model by integrating an RFID-based system into its manufacturing environment.

The rest of this paper is organized as follows: In Sect. 2 we review related literature. In Sect. 3 we describe related problems and demand analysis. The framework of the proposed system is outlined in Sect. 4. In Sect. 5 we describe the actual implementation and performance evaluation of the proposed system. Finally, in Sect. 6, we present our conclusions.

2. Literature Review

RFID is an automatic object identification and data collection technology that utilizes radio waves [11], [12]. RFID tags can be categorized into either active or passive types. Active tags are powered by an internal battery, can typically function in a read/write mode, and have longer reading ranges. Passive tags do not rely on an internal power source. Their operating power is obtained from a transceiver. Consequently, passive tags are much lighter and can operate

---

Manuscript received January 18, 2008.

Manuscript revised May 26, 2008.

[†] The author is with the Department of Information Management, China University of Technology, Taiwan.

[††] The author is with the Institute of Information Management, National Chiao Tung University, Taiwan.

[†††] The author is with the Institute of Information Management, National Chiao Tung University and Identification and Security Technology Center, Industrial Technology Research Institute, Taiwan.

a) E-mail: ystsai@ms72.hinet.net

DOI: 10.1093/ietisy/e91–d.12.2834

over much longer periods [11], [13]. An RFID framework for ubiquitous computing applications should usually handle data such as location, neighborhood, time, linkage between the physical and virtual world, history, context, name, and address [14].

RFID technology has had a tremendous impact on education, healthcare, manufacturing, transportation, retailing, services, and even war [15]. In the field of manufacturing, a number of researchers have utilized RFID to improve the core activities of the manufacturing supply chain including production, warehouse management, distribution, and quality assurance. For example, in the field of production, Qiu [16] utilized RFID to bridge the gap between shop-floor automation and factory information systems for more effective and efficient factory system integration. Huang et al. [17] utilized RFID for the collection and synchronization of real-time field data from manufacturing workshops located in a walking-worker, fixed-position, flexible assembly line. In addition, for warehouse management, Chow et al. [18] presented an RFID-based resource management system to help warehouse users handle orders by retrieving and analyzing useful knowledge from a case-based data warehouse. Finally, in the field of quality assurance, Parlikad and McFarlane [19] proposed RFID-based product identification technologies to improve product recovery decisions by providing the necessary information associated with the product after its sale.

However, few studies have reported on the application of RFID to both mass customization and the JIT production model. Therefore, in this study we propose an RFID-based manufacturing control framework to help smooth manufacturing flows and enhance real-time information visibility and traceability for enabling mass customization and JIT production. In this study, instead of only using message or service exchanging to perform integration in an enterprise, we used RFID to “hook” the physical objects in an enterprise with applications that traditionally have not been easy to integrate.

3. Statement of Problem and Demand Analysis

The case study, XMbike (a fictitious name chosen to preserve the anonymity of the manufacturer), is a leading global bicycle manufacturing company based in Asia. XMbike sells middle- and high-end bicycles to Asia, Europe, and North America and has manufacturing plants across Taiwan, China, and South East Asia. The overall bicycle manufacturing process from receiving frame tube to packing and shipping is shown in Fig. 1.

3.1 Statement of Problem

JIT production is a system that aims to eliminate the need to maintain a large inventory of parts for production. XMbike has adopted a JIT system to produce their various bicycle models. JIT has improved XMbike’s production efficiency and helped reduce its inventory for the last few years.

However, since the company used paper travelers with barcode labels for tracking and identifying the thousands of frames that are moved around the plant each day, traceability was not accurate. The lack of accurate real-time work-in-process information can seriously affect the effectiveness of JIT and supply chain planning, particularly if the company wishes to achieve mass- customized production. By analyzing a model of this process, various potential problems and inefficiencies have been identified and are discussed below.

1. The barcode system can only cover the processes in the final assembly stage after the painting process since barcodes cannot be applied to parts subjected to heat or surface treatment owing to the harsh production environment in these processes. Therefore, the firm must employ manual tracking before the final assembly stage.
2. The job card system of data tracking contained only the product assembly information needed by the production operator at a particular station. The system did not have the capacity for real-time data tracking. Thus, such a system could not be relied on for quick access to information on the assembly process, item location, assembly problems, and failure analysis.
3. Finding lost frames was time-consuming owing to the large facility space and the manual tracking process.
4. The job card is normally attached to frames; however, in some production processes, such as heat treatment or painting, the job card and the frame must be separated

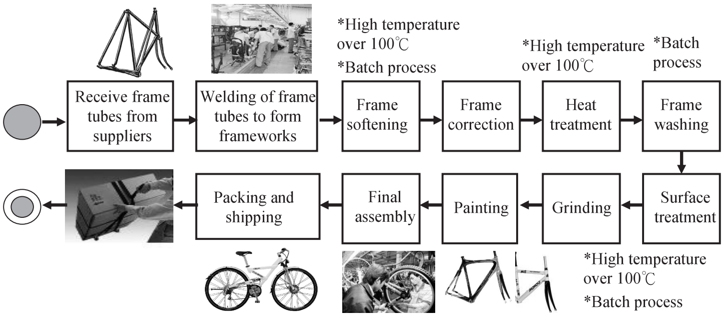


Fig. 1 Major bicycle manufacturing processes.

and then correctly matched after the process. All these actions are performed by the field operator, thus are prone to error.

5. Rework processes take place from time to time on the first floor of the facility. Since frames in this area are manually tracked, rework processes sometimes lead to the loss of frames.
6. Any missing frames would affect JIT and just-in-sequence (JIS) assembly processes in the final assembly stage, and it could result in much higher costs if a mass-customization system was adopted.

3.2 Demand Analysis

Most of XMbike's plants are designed for the manufacture of limited-choice mass-produced models with price ranging from US $1000 to $2000. However, some plants manufacture high-end models in small quantities with price above US $3000. Faced with strong competition from other bike manufacturers, XMbike plans to increase its market share of the high-end market segment by targeting customers who want bicycles configured to their own specifications. The firm plans to allow the customer to select among a variety of model types, frame sizes, colors, and other features. The company estimates that the customer can choose from about 6 million possible variations when ordering a custom-made bike.

The process employed to produce such custom-made bicycles requires not only highly trained and skilled workers but also restructuring of the manufacturing information system to accommodate such a change. Therefore, a mass-customization manufacturing strategy naturally leads to the development of a system that promotes attention to detail and encourages operators to achieve 'zero mistakes' in every step of the production process. In light of their strategic goal, XMbike plans to launch a mass-customization production line within one of its most advanced plants that is currently manufacturing a variety of high-end models in small quantities and special orders.

4. Design of RFID-Based Manufacturing Control Framework

4.1 Proposed RFID Framework

On the basis of the nature of manufacturing control, we propose an RFID-based manufacturing control framework by utilizing RFID tags as the physical objects that integrate the manufacturing processes and the applications. The proposed framework includes RFID tags with a special data structure containing event identifiers and software modules to process these events, as shown in Fig. 2. The event identifiers are a string of information occupying a small amount of memory in an RFID tag. Information can be read from and wrote into this memory space, thus making the tag a mobile data carrier with its own memory.

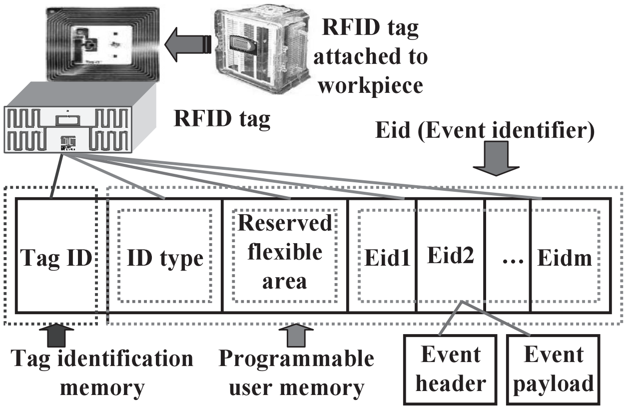


Fig. 2 Data structure of RFID tag used as a mobile data carrier.

The memory data structure comprises three main areas. They are the tag ID, the tag object type and attribute, and the event identifiers. These data types are explained in detail below:

1. Tag ID: The tag ID is a unique code of the RFID tag representing a physical object, and the coding can follow an industry standard (e.g., EPC Global or ISO standards) or be unique to the company.
2. Tag object type and attribute (ID type): The ID type is used for recording the class of the product and associated information related to the product to facilitate the rapid classification of tagged physical objects.
3. Event identifier (Eid): Each Eid contains either a pointer or encoded information to represent actual routing or service instructions. Each Eid comprises an event header and an event payload. The event header is used to record various process flags and status information, and the event payload is used to storage various event-related setting data. These Eid lists can form a type of product pedigree that contains the history of events that have taken place for a tagged product.

4.2 Description of Components in the Framework

Figure 3 shows a schematic view of the overall proposed RFID-based manufacturing control framework. This framework also illustrates the processes of application integration and processing interface information. The major components of application integration and interface information are RFID middleware, the tag handler, session controller, service dispatcher, event manager, session data, tag activity log, and event interactions. The various roles of software components and other constituents of our framework are shown in Fig. 3. The functions of these components are described in Sect. 4.2. The system operations of these components used for processing application integration and interface information are described in Sect. 4.3.

1. RFID middleware: The middleware is mainly responsible for managing RFID readers and filtering raw RFID

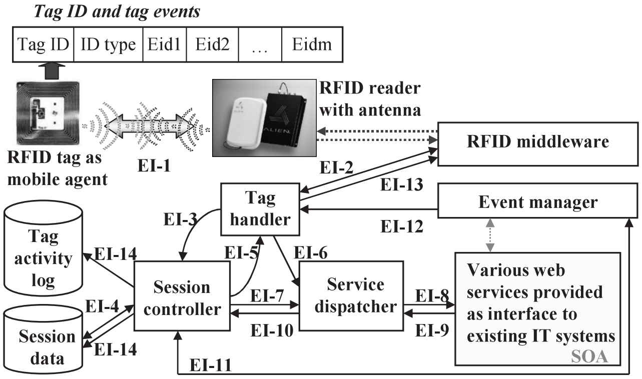


Fig. 3 Software components and event interactions.

data, thus providing a logical interface allowing application users to perform physical RFID reader management and low-level raw RFID data handling. Further RFID event processing is carried out by the tag handler in our proposed framework.

2. Tag handler: The tag handler is responsible for the tag read/write process and parsing and analyzing the incoming tag event information. It also facilitates the tag-handling process based on location-specific tag-processing logic. It usually requires the RFID middleware to facilitate its tag read/write processes.
3. Session controller: The session controller primarily performs a data access and discovery role in the framework. It also provides a data cache for sets of parameter data accepted from a tag handler and location- and/or operation-specific information retrieved from session data (requested by the same tag handler). When an RFID event-processing cycle (initiated by the tag handler) is completed, it synchronizes the processing information with the session data and releases the memory resource for use in a computer server. These caching and RFID event session cycle management operations provided by the session controller help improve the overall RFID event-processing speed and data integrity.
4. Service dispatcher: The service dispatcher is responsible for receiving an event-processing notification from the tag handler, communicating with the session controller, and directing an event to corresponding services. The service dispatcher is also responsible for application integration between the framework and external applications including enterprise resource planning (ERP) system and manufacturing execution system (MES). It receives related interface information such as job orders from ERP and working orders from MES, and it also sends related interface information to them.
5. Event manager: The event manager is responsible for preparing new events and producing a write or clear instruction for the data of the RFID tag through the tag handler. It retrieves new event information from session data or receives overwriting event-handling instructions from various services and then prepares the information required by the new event and produces a new event code, before requesting the RFID tag handler to write the correct associated event information to the corresponding RFID tag. Additionally, the event manager can be customized to include some decision-making logic in computing the optimal production route or check product quality.
6. Session data: In the manufacturing environment, this data repository usually resides in the line-control PC for a processing area or a local production line. The following types of information are stored:
    (1) Mapping information for the RFID reader/antenna and the area/location covered by it.
    (2) Master data regarding each specific processing area or production line.
    (3) Recipe-related information of local machine/tools in a processing area or production line.
    (4) System parameters and other data specific to the local processing area or production line.
7. Tag activity log: This is an operational data store that keeps not only all the tag transaction data as a tagged item moves from one location to another but also some historical information. This data repository can be located in the central system that monitors all the distributed line-control PCs. It can also reside in local line-control PCs and then periodically share its information with the central system.
8. Event interactions: The functions of these Eid lists focus on the interface information. Information stored in these Eid lists is first made available through the initialization process. A complete RFID tag-handling cycle is illustrated by event interactions (EI-1 through EI-14) for a single location/processing area. When a tag enters a new location/processing area, this cycle is repeated.

4.3 Description of Interface-Information-Handling Logic

Here we describe the system operations of the components used for processing the application integration and interface information. The interactions and information flows among these components of the framework used for processing RFID interface information are referred to as event interactions (EIs). The details of these interactions and the corresponding interface-information-handling logic are described below:

1. EI-1: A tagged object transmits its data to an RFID reader whenever it enters the reading range of a reader's antenna. The RFID reader then passes this data to the RFID middleware which constantly monitors several reader devices. The RFID middleware, upon receiving the raw tagged data, performs data filtering and obtains the tag ID, the product type and attributes, and the event content from the RFID reader. This middleware then relays them to the target tag handler. In the case that the tag data need to be parsed,

the tag handler must also incorporate a data parser to perform the parsing operation.

2. EI-2: The tag handler is coupled to the RFID middleware, and the tag data can either be sent to the tag handler by the middleware or be cached in the middleware first and then stored by the tag handler when it has capacity to process more tag data (in the case of processing multiple tags).
3. EI-3: After receiving tag data, the tag handler immediately parses and analyzes the tag ID string and transmits the associated parameters (information such as tag ID, ID type, and Eid) to the session controller.
4. EI-4: The session controller obtains the information associated with a certain tag as well as location- and/or operation-specific information from the session data using the parameters passed from the tag handler and then caches these data in its memory.
5. EI-5: The session controller sends the key parameters regarding a certain mobile agent back to the tag handler.
6. EI-6: Using the tag data obtained from the RFID middleware and the parameters obtained from the session data, the tag handler compares the analyzed tag ID and the event parameters with the associated parameters provided by the session manager, so as to determine whether the event represented by the RFID tag can be processed at this location and at this time. If yes, the tag handler sends a notification with associated parameter data to the service dispatcher to process the event; otherwise, it informs the service manager to handle the problem and display an error message and relevant correcting information for the user.
7. EI-7: The service dispatcher, upon receiving an event-processing notification and the associated parameter data from the tag handler, retrieves the location- and/or operation-specific (event process logic or method) information cached in the session controller in accordance with the tag ID, location ID, Eid, and other associated information given by the parameters. Alternatively, if detailed event data are predefined and preloaded in the tag memory bank during the initialization of the tagged object, the service dispatcher performs its operation directly in accordance with the associated setting data and instructions associated with the tag sent from the tag handler.
8. EI-8: The service dispatcher transmits the event process logic or interface information to the external systems such as ERP and MES, and it drives the automatic process flow (it either directly drives or communicates with the external system through the network service or waits for a manual process).
9. EI-9: When the external process flow is completed, the external interface information is sent back to the service dispatcher by the external systems such as ERP and MES (directly or through the network service).
10. EI-10: The service dispatcher then relays the external processing information to the session controller.
11. EI-11: The session controller registers the external processing information in its cache memory and sends a completion notice and the associated parameters to the event manager.
12. EI-12: The event manager reads the associated event process flow information from the session data (through the session manager) in accordance with the associated parameters, and it obtains the latest information regarding the status of a mobile agent (tagged object) and its environment (such as resource or routing information). Finally, data comparison and computing are performed in accordance with the information, and the associated information of the event to be completed in the next stage of the tag (workpiece) is obtained. New event information (e.g., ID type and Eid) is generated. Alternately, if the Eid is predefined and preloaded in the tag during the initialization of the tagged object, then the event manager directly reads out the new event information from the Eid lists residing in the tag memory. Finally, the event manager transmits the new event information to the tag handler.
13. EI-13: The tag handler writes the new event into the RFID tag (i.e., the corresponding mobile agent) through the RFID middleware and RFID reader, and it also updates the associated parameters of the current event.
14. EI-14: After the tag event is written successfully, the tag handler informs the session controller so that the associated data registered in its cache memory is written to the system database (session data and tag activity data), thus releasing the corresponding cached event data associated with the workpiece from the memory session controller during this event-handling session.

4.4 Proposed RFID System Layout

The factory layout for XMbike's mass-customization production line is a two-story building with a large floor space for the production and storage of bicycles. Its shop-floor control environment is loosely coupled and managed by field operators. The RFID system layout in XMbike's plant should include RFID readers positioned at strategic points and RFID tags attached to the frame at an appropriate position. A local PC/workstation must also be deployed to control RFID readers and provide a client-side shop-floor control service. The overall RFID system layout from receiving the frame tubes to packing and shipping is shown in Figs. 4 and 5.

4.5 Using RFID Tag to Maintain Production Pedigree

Each mass-customized bicycle has its own mobile production pedigree during the production process from welding to the final assembly. Furthermore, the information stored in the production pedigree can include not only static information such as bicycle ID, routing, or method information

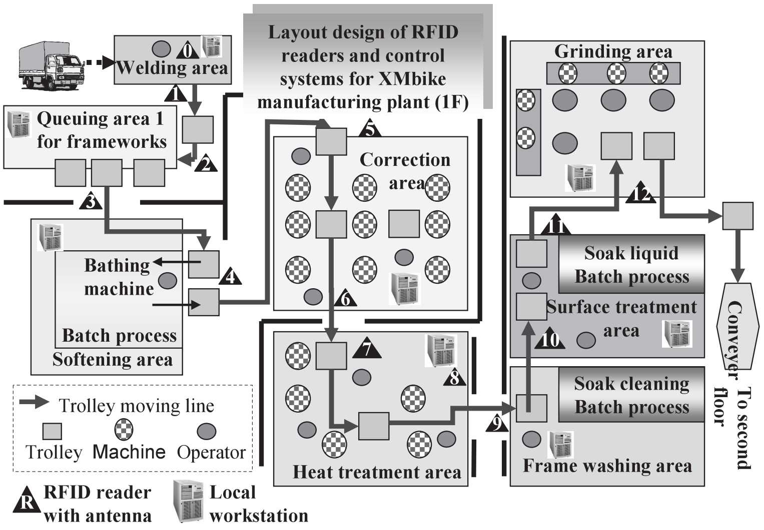


Fig. 4 RFID system layout for XMbike -1st floor.

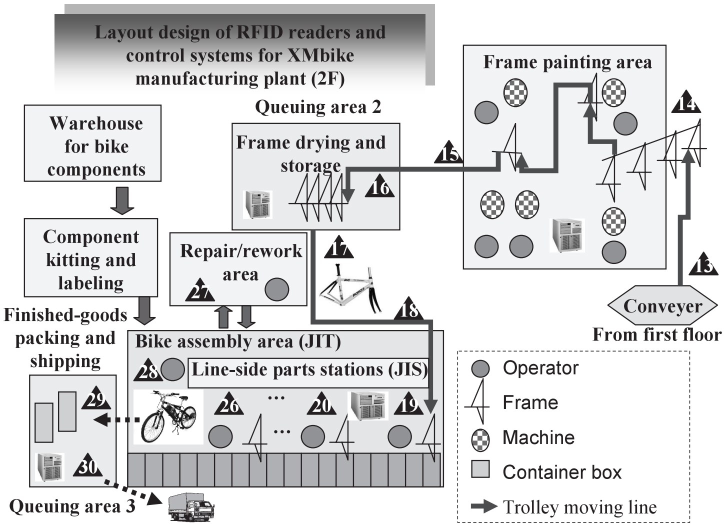


Fig. 5 RFID system layout for XMbike -2nd floor.

but also dynamically changing production state information. XMbike can extract and update this pedigree information through a control workstation or a PC installed independently for each production line or processing area. Work-in-process and production status information can be collected at certain sink point with some time delay as long as XMbike knows the whereabouts of all customized bicycles and ensures that each bicycle has been correctly processed.

With its read/write capability, RFID can be used as a dynamic electronic manifest, allowing users to reduce traffic in networks, to link remote production stations, and to backup host PCs or PLCS. In XMbike's case, each tag is attached to a frame, and routing and assembly instructions are written to the tag. As the frame approaches the first station, the tag is read by a reader/writer to determine whether or not the frame should be processed at a certain processing area. If affirmative, the assembly information is read off the tag and transferred to the local workstation and displayed on an LCD screen, where the operator can follow the instructions. After the operations are performed, key quality data and/or production results can be stored on the tag. This allows users to later investigate any quality issues in various batches.

In the case when the operation is unsuccessful, this failure is also written to the tag. Then, prior to reaching the next station, the frame is removed from the main production line and transferred to a remote rework station. At the rework station, the tag is read to determine how the frame must be repaired or what processes had been successfully performed before the failure. Finally, the production pedigree stored in the RFID tags can be extracted by a local PC/workstation at any time.

4.6 RFID Security

Security risks in RFID have become a crucial issue and need to be dealt with carefully [20]. On the basis of the Federal Information and Security Management Act, which regulates the security of information technology systems, the National Institute of Standards and Technology (NIST) has issued security guidelines for RFID systems. These guidelines give organizations many checklists and specific recommendations regarding potential RFID security risks, and they also lay the foundation for managing RFID security. To secure our proposed RFID manufacturing control system, we have applied these guidelines to this system. Some security measures included in our proposed system are as follows:

1. Firewall: We utilized the firewall to separate the Intranet environment from the Internet environment. The RFID system is located in a secure Intranet environment. This can prevent the RFID system from attacks by hackers.
2. Virus detection: We installed virus detection software in the server and workstations, which protects these computers against the threat of viruses and spyware.
3. Authentication: When shop-floor operators login to the RFID manufacturing control system, they must key in their user identification and password. After they login to this system, they must use the RFID reader to read their own RFID tag, which stores personal identification information. After this two-step operation they can initiate and operate the RFID system. The two-step operation can correctly authenticate approved shop-floor operators.
4. Confidentiality: The RFID tags only store production control data, which are not very sensitive. The confidential data are separated and stored in a well-protected central database.
5. Access control: We block tag signals with metal shields to prevent production control data from unauthorized access to some important processing areas.
6. Standard operation procedures: We establish standard operation procedures to manage the RFID tags. These procedures focus on tag initiation, tag disposal, tag recycling, and tag data cleaning.

5. Implementation and Results

To demonstrate the applicability and usefulness of the proposed framework, we developed a prototype system based on the framework, including software components and hardware devices, to validate the design of the framework.

5.1 Information Infrastructure

The processing of the real-time RFID data can be divided into four layers, as illustrated in Fig. 6. In this study we construct the information infrastructure of the proposed enterprise application integration (EAI) framework as a multi-tiered, web-based structure under an Internet/Intranet network system to provide a real-time information transaction environment. The architecture consists of three parts — the hardware devices include passive RFID tags and RFID readers with antennas; the RFID middleware provides a logical control interface to manage RFID readers and to preprocess the large volume of raw RFID data; finally, the RFID application contains most of the software components and application modules as well as a system database.

We deployed the RFID middleware system and shop-floor control application to a local workstation and the central database to the central server. The RFID readers are connected to the local workstations and communicate with the RFID middleware through a local-area network (LAN) system. By mapping the information infrastructure of the prototype system (Fig. 7) to the layered RFID information-processing architecture, we can see that the RFID readers and tags correspond to the bottom layer of the processing architecture, RFID middleware corresponds to the RFID event-handling layer, and the RFID application implements both business logic processing and business process integration layers.

The development environment of the RFID prototype

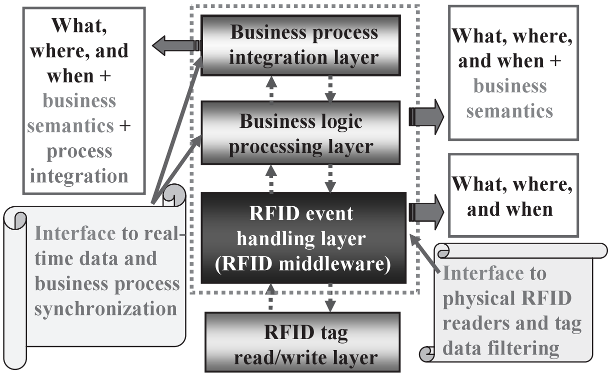


Fig. 6 Layers of RFID information-processing architecture.

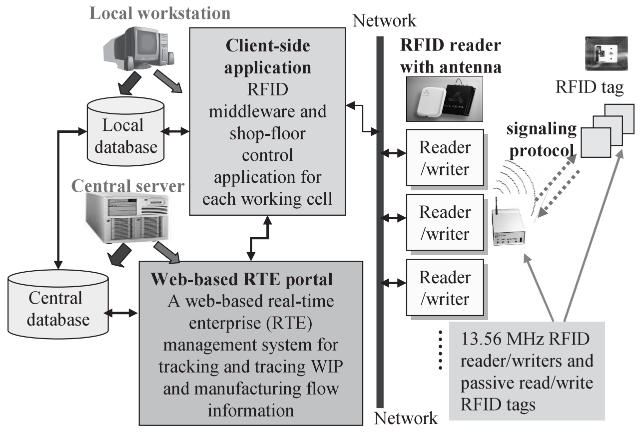


Fig. 7 Information infrastructures of the prototype system.

system has the following specifications:

1. OS: server: WinXP, client: WinXP.
2. Database: SQL server.
3. Application program: J2EE technology stack (JSP, JDBC, Java Beans . . . etc).
4. Application server: Tomcat.
5. RFID reader: ISO standard 13.56 MHz RFID reader/writer.
6. RFID tag: ISO standard passive read/write tags with memory capacity up to 8 kb.

For the information infrastructure illustrated in Fig. 7 and the development environment of the RFID-based manufacturing control system, the major costs for implementing this system are the RFID tags, RFID readers, RFID software, a server machine, workstations, LAN, and application development. The total cost is US $113100, the breakdown of which is given in Table 1.

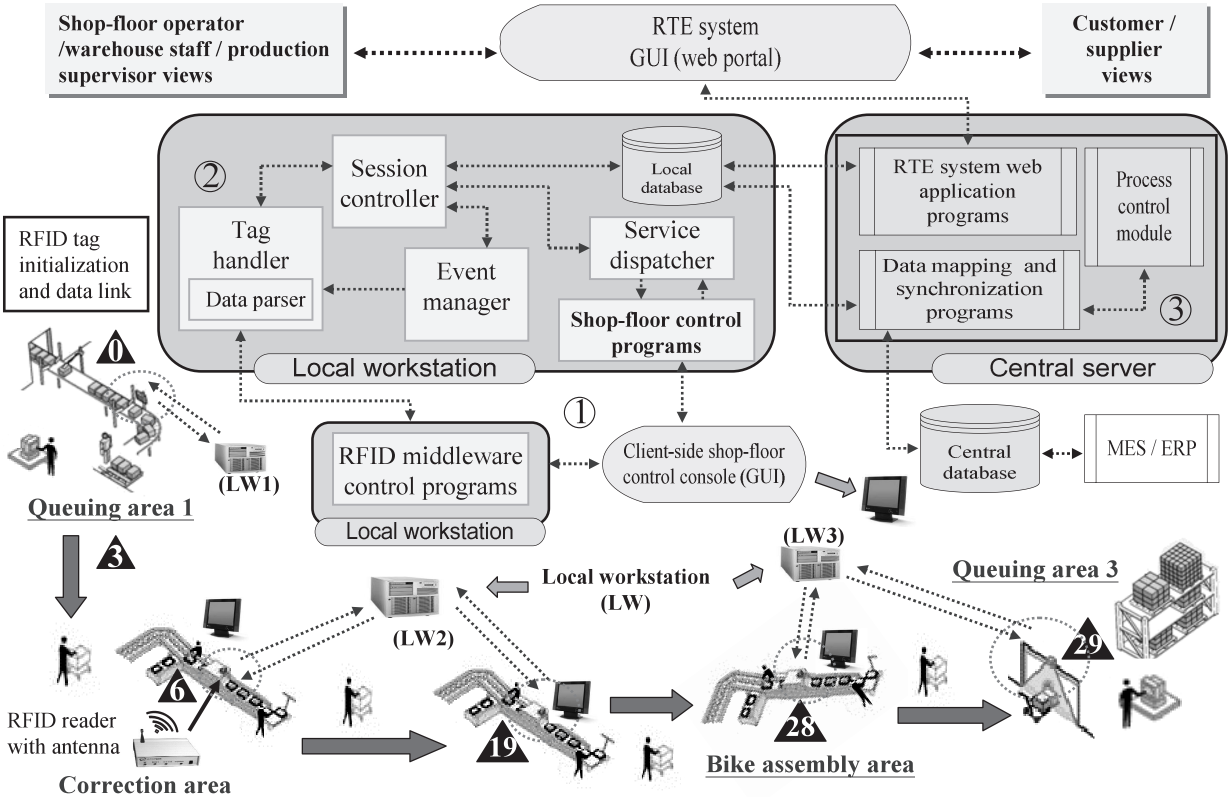
5.2 System Implementation

The prototype system is an integrated information system used in a shop-floor control environment to enable the real-time tracking and tracing of manufacturing process flow information for better process visibility and product quality. This prototype system is divided into three main parts, each consisting of several components, as shown in Fig. 8.

1. RFID middleware system: The system includes RFID middleware control programs that directly control the RFID readers and perform raw RFID data preprocessing tasks. Built upon the control programs is an RFID

Table 1 Implementation cost.

Item	Quantity	Amount (US $)
A server machine	1	2000
Workstations	12	12000
Local-area network	1	1300
RFID readers	30	81000
RFID tags	1000	800
RFID software	1	2000
Application development	1	14000
Total		113100

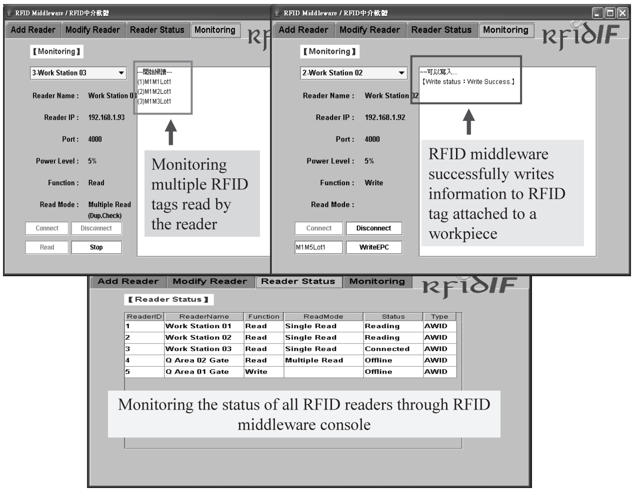


* **Solid lines** represent **the flow of physical materials** while **dotted lines** represent **the flow of virtual information**. **Dotted circles** indicate the coverage areas of the RFID readers with an antenna (usually deployed at a gate or at entrance and exit points of a working/testing station).

Fig. 8 Implementation of the prototype system.

reader client management console, which is a Java client application providing GUI interfaces for configuring reader settings and monitoring reader activities. A demonstration of the client application is shown in Fig. 9.

2. RFID event logic-processing integration system: This system is installed in each local PC/workstation. It includes the tag handler and the dual tag data parser within the tag handler, the session controller, event manager, service dispatcher, shop-floor control programs, client-side shop-floor control console, and a local database. These software components are implemented in Java programming language, and each has its own execution logic. For example, the tag handler was created to handle the frame checkout scenario at a processing area. The information stored in the RFID tag can be interpreted by the dual tag data parser in the tag handler. The architecture of the parser is depicted in Fig. 10. The parser is implemented using pushdown automata (PDA) for encoding and decoding the tag data. Furthermore, we made the event manager more versatile by implementing a quality assurance function for each bicycle frame. The event manager references quality lookup tables and relevant accept/reject parameters to determine whether the frame should continue to the next processing area or start the rework procedure. Finally, the operator interacts with the system through the client-side shop-floor control console.



【Monitoring】

3-Work Station 03

Reader Name : Work Station 03

Reader IP : 192.168.1.93

Port : 4000

Power Level : 5%

Function : Read

Read Mode : Multiple Read (Dup.Check)

---開始掃讀---
(1)M1M1Lot1
(2)M1M2Lot1
(3)M1M3Lot1

Monitoring multiple RFID tags read by the reader

【Monitoring】

2-Work Station 02

Reader Name : Work Station 02

Reader IP : 192.168.1.92

Port : 4000

Power Level : 5%

Function : Write

Read Mode :

M1M5Lot1

~~可以寫入...
【Write status：Write Success.】

RFID middleware successfully writes information to RFID tag attached to a workpiece

【Reader Status】

ReaderID	ReaderName	Function	ReadMode	Status	Type
1	Work Station 01	Read	Single Read	Reading	AWID
2	Work Station 02	Read	Single Read	Reading	AWID
3	Work Station 03	Read	Single Read	Connected	AWID
4	Q Area 02 Gate	Read	Multiple Read	Offline	AWID
5	Q Area 01 Gate	Write		Offline	AWID

Monitoring the status of all RFID readers through RFID middleware console

Fig. 9 Demonstration of RFID middleware system.

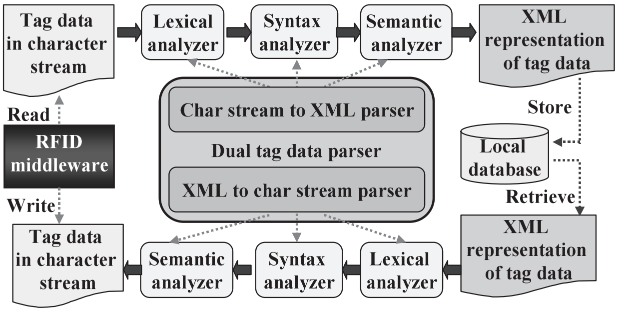


Fig. 10 RFID tag event data two-way parsing process.

3. High-level real-time enterprise (RTE) management system: This part is divided into three submodules to carry out specific system operations. The data mapping and synchronization programs are responsible for most of the data-intensive operations such as data extraction, consolidation, and synchronization. The RTE system web application programs support most of the front-end, user interface operations of the web portal. It also receives requests from the service manager and performs certain tasks accordingly. The process control module updates and maintains the manufacturing process and routing information that allows the supervisor or manager to access accurate and real-time process information.

5.3 System Operations

The prototype system simulates the shop-floor operation for the XMbike manufacturing environment. When receiving customer orders, the company's ERP system generates a corresponding job order and transfers it to the MES system. Then, the MES system transforms the job order into working orders and notifies our prototype system of the arrival of new working orders. The prototype system then extracts the working orders from the MES database and consolidates the information into the central database. A production supervisor can release the working orders. The manufacturing processes are initiated in the preprocessing stage, and relevant information is written to the RFID tags attached to bicycle frames at this stage.

When a frame has completed its initialization process, its information is immediately relayed to the first processing area and the shop-floor operators. Additionally, the required parts or materials are also prepared in this area and are delivered to the processing area on demand. When the frame arrives at the processing area, a series of events related to the processing of the frame occur, as shown in Fig. 11 and described below.

1. When a frame with an RFID tag passes a check-in point at a processing area, the tag becomes active.
2. The tag handler then receives the tag data from the RFID middleware, and then the dual tag data parser in the tag handler reads out the tag data and parses the tag information.
3. The tag handler decides whether a bicycle frame has been successfully processed in all the preceding operations and thus can be handled at this time and this location.
4. If the frame can be processed, the service dispatcher instructs the shop-floor control programs to display the production pedigree of the bicycle frame on the client-side shop-floor control console. Otherwise, if the frame cannot be processed at this processing area, the client-

side shop-floor control console displays and provides an error message and suitable guidance to operators.

5. After the operator confirms the completion of a task on the client-side shop-floor control console, the event manager compares frame information with lookup tables and accept/reject parameters in the local workstation database to determine whether this operation has been successfully completed. The event manager determines the rework/reentrance routing path for the bicycle frame. Finally, the event manager instructs the tag handler to update information to the RFID tag on the frame.
6. The tag handler first communicates with the RFID reader deployed at the checkout point to confirm that the correct frame is in place. If the correct frame is in place, the tag handler can start the updating operation; it updates the next operation and next processing area parameters in the RFID tag. The tag handler then notifies the session controller of the completion of the production event-handling cycle.
7. Finally, the session controller stores the associated production event-handling data in the local database. The central server is scheduled to extract information periodically from the local database at each workstation.

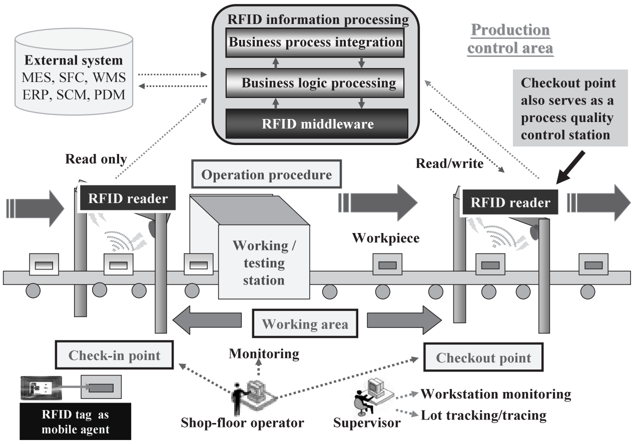


Fig. 11 Manufacturing operation in production control area.

5.4 Evaluation and Benefits

The surveyed company used paper travelers (run cards) with barcode labels for tracking and identifying the thousands of workpieces (frames) moved around its production facility each day. Tracking and tracing the manufacturing process information involved human intervention that often resulted in many errors and delays in obtaining information. Furthermore, these problems affected process quality control.

Three criteria were applied to compare the performance of the proposed RFID system and the conventional run cards with a barcode system over six months of observation. They were throughput time, productivity, and quality failure rate. The throughput time means the total manufacture time for a customized bike, which consists of the time to receive an order, prepare parts, manufacture the bike, perform a quality check, and the packing and shipment. The throughput time is measured in hours. Productivity means the numbers of bikes completed per minute in a production line, measured as production quantity per minute. The quality failure rate means the percentage of bikes in each working day that are defective.

Figure 12 shows the results generated from the RFID system and the conventional run cards with a barcode system. The results reveal that the production performance was significantly improved using the proposed system. The throughput time was reduced from 10.8 hours to 6.2 hours (42.6% decrease). The productivity was increased from 1.1 per minute to 1.5 per minute (36.4% increase). The quality

Table 2 Comparison of manufacturing process tracking.

Aspect	Run card and barcode-based tracking system	RFID-based tracking system
Convenience	Requires line-of-sight scan	Automatic scan without line of sight
Efficiency	Cannot support batch reading	Can read multiple tags simultaneously
Accuracy	Susceptible to misreads and human error	Reduced human error and misreads improves data accuracy
Traceability	Limited traceability. Some processes in harsh environments (such as baking) make barcode tracking impossible.	Allows detailed tracking and tracing of process status, inputs/outputs, and the time at which each processing step was performed
Speed	Process information not in real-time	Real-time process information
Reliability	Barcode are easily dirtied or scratched in harsh manufacturing environment	RFID tag can survive harsh environments involving dirt or high temperatures
Automation	Needs more human labor to collect and track process data	Reduced human labor required for data collection and tracking
Information	Limited process information	Large amount of detailed process information
Storage (Data)	Allows only centralized data storage	Both centralized and decentralized data storage

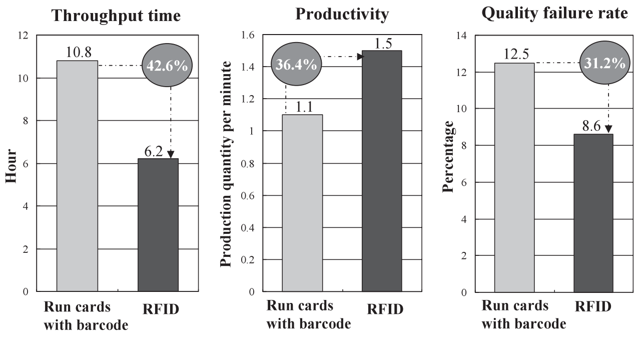


Fig. 12 Comparison of performance.

failure rate was reduced from 12.5% to 8.6% (31.2% decrease).

Table 2 shows a comparison of the information- tracking capability of the company's current tracking technology and our proposed system. It is clear that RFID technology can significantly close the gap between product flow and information flow. In addition to improved tracking and tracing capability, the prototype system also greatly enhanced the company's quality control of its manufacturing process because our system's checkout function helped shop-floor operators detect quality problems immediately and resolve these problems before the next stage of manufacturing.

In summary, as shown by the results of our research, we found that overall manufacturing process monitoring and control could be improved by the application of RFID technology and our prototype system.

6. Conclusion

The transition of manufacturing enterprise into mass-customization production with a ubiquitous and real-time organization is one of the major trends in globalization. However, to become a real-time and flexible enterprise, it is necessary to use information technology to integrate related information and physical objects at different times and different locations. In summary, this study makes the following contributions to improving the traceability and visibility of mass-customization manufacturing processes.

1. We have proposed an RFID-based manufacturing control framework for tracking and controlling the manufacturing process flow.
2. On the basis of the framework, we implemented an integrated prototype system that consists of RFID middleware and a shop-floor control information system to process RFID data in real time.
3. The findings of this study reveal that the prototype system can provide both the shop-floor operators and the production supervisor with real-time production process information, helping them to respond to the status of the production line in real time and to make better decisions when handling production events.
4. The proposed framework can markedly improve the manufacturing process and quality control.

References

[1] M. Fletcher and R.W. Brennan, "Designing holonic manufacturing systems using the IEC 61499 (function block) architecture," IEICE Trans. Inf. & Syst., vol.E84-D, no.10, pp.1398–1401, Oct. 2001.

[2] T. Miyamoto, D. Ichimura, and S. Kumagai, "A multi-agent based manufacturing resource planning and task allocation system," IEICE Trans. Fundamentals, vol.E86-A, no.4, pp.806–812, April 2003.

[3] X. Du, J. Jiao, and M.M. Tseng, "Understanding customer satisfaction in product customization," International Journal of Advanced Manufacturing Technology, vol.31, no.3–4, pp.396–406, 2006.

[4] R.D. Banker, I.R. Bardhan, H. Chang, and S. Lin, "Plant information systems, manufacturing capabilities, and plant performance," MIS Quarterly, vol.30, no.2, pp.315–338, 2006.

[5] N. Chokshi, A. Thorne, and D. McFarlane, Routes for integration Auto-ID systems into manufacturing control middleware environments, White Paper, Auto-ID Center, 2004.

[6] A. Tu, J.S. Jwo, and J.Y. Kuo, "An enterprise application integration framework using RFID," Proc. 36th CIE Conference on Computers & Industrial Engineering, pp.2750–2760, 2006.

[7] A. Yamaguchi, M. Ohashi, and H. Murakami, "Autonomous decentralized control in ubiquitous computing," IEICE Trans. Commun., vol.E88-B, no.12, pp.4421–4426, Dec. 2005.

[8] R. Kotorov, "Ubiquitous organization: Organizational design for e-CRM," Business Process Management Journal, vol.8, no.3, pp.218–232, 2002.

[9] C. Floerkemeier and M. Lampe, "Issues with RFID usage in ubiquitous computing applications," Lect. Notes Comput. Sci., vol.3001, pp.188–193, 2004.

[10] K.H. Doerr, W.R. Gates, and J.E. Mutty, "A hybrid approach to the valuation of RFID/MEMS technology applied to ordnance inventory," International Journal of Production Economics, vol.103, no.2, pp.726–741, 2006.

[11] D. Paret, RFID and contactless smart card applications, Wiley, 2005.

[12] K. Tanaka, Y. Kimuro, K. Yamano, M. Hirayama, E. Kondo, and M. Matsumoto, "A supervised learning approach to robot localization using a short-range RFID sensor," IEICE Trans. Inf. & Syst., vol.E90-D, no.11, pp.1762–1771, Nov. 2007.

[13] K. Finkenzeller, RFID handbook: Fundamentals and applications in contactless smart cards and identification, Wiley, 2003.

[14] K. Romer, T. Schoch, F. Mattern, and T. Dubendorfer, "Smart identification frameworks for ubiquitous computing applications," Proc. 1st IEEE International Conference on Pervasive Computing and Communications, pp.253–262, Nov. 2003.

[15] A. Gunasekaran, E.W.T. Ngai, and R.E. McGaughey, "Information technology and systems justification: A review for research and applications," Eur. J. Oper. Res., vol.173, no.3, pp.957–983, 2006.

[16] R.G. Qiu, "RFID-enabled automation in support of factory integration," Robotics and Computer-Integrated Manufacturing, vol.23, no.6, pp.677–683, 2007.

[17] G.Q. Huang, Y.F. Zhang, and P.Y. Jiang, "RFID-based wireless manufacturing for walking-worker assembly islands with fixed-position layouts," Robotics and Computer-Integrated Manufacturing, vol.23, no.4, pp.469–477, 2007.

[18] H.K.H. Chow, K.L. Choy, W.B. Lee, and K.C. Lau, "Design of a RFID case-based resource management system for warehouse operations," Expert Systems with Applications, vol.30, no.4, pp.561–576, 2006.

[19] A.K. Parlikad and D. McFarlane, "RFID-based product information in end-of-life decision making," Control Engineering Practice, vol.15, no.11, pp.1348–1363, 2007.

[20] Y. Cui, K. Kobara, K. Matsuura, and H. Imai, "Lightweight privacy-preserving authentication protocols secure against active attack in an asymmetric way," IEICE Trans. Inf. & Syst., vol.E91-D, no.5, pp.1457–1465, May 2008.

Ruey-Shun Chen was an associate professor at the Institute of Information Management at National Chiao-Tung University, Taiwan. He is now an associate professor at the Department of Information Management at China University of Technology, Taiwan. He received his Ph.D. from the Department of Computer Science and Information Engineering, National Chiao-Tung University. His research interests are radio-frequency identification, data mining, genetic algorithms, reliability, performance evaluation, business networks, and distributed systems.

Yung-Shun Tsai is a Ph.D. student at the Institute of Information Management at National Chiao-Tung University, Taiwan. His research interests are radio-frequency identification, data mining, management information systems, ontology, knowledge management, optimization, and artificial intelligence.

Arthur Tu is a Ph.D. student at the Institute of Information Management at National Chiao-Tung University, Taiwan. He is also a researcher at Identification and Security Technology Center at the Industrial Technology Research Institute, Taiwan. His research interests are radio-frequency identification, real-time enterprise, ontology, management information systems, and artificial intelligence.