An Infrastructure to Construct an Individualized Manufacturing Information System for Small and Medium Manufacturing Enterprises

Joseph Oh, Bo-Hyun Kim and Jae-Yong Baek

Abstract Small and medium manufacturing enterprises (SMMEs) face many difficulties when introducing their manufacturing information systems because of realistic limitations such as the large initial investment needed to buy the system, the burden of maintaining staff to operate it, and continuous payment of maintenance costs, etc. However, most systems are not used properly due to the characteristic difference between SMMEs' shop-floors and their functions, the lack of understanding concerning their many complex functions, and so on. To overcome the situation, we propose an infrastructure to support the construction of the individualized manufacturing information systems for SMMEs. The infrastructure consists of a Manufacturing Application (MfgApp), Plug-and-Play (PnP) Platform, and a Manufacturing Application Store (MfgAppStore) such as Apple's business model. The MfgApp is a small-sized application software to treat specific manufacturing tasks and is performed only on a PnP Platform. The PnP Platform includes a PnP Developer which provides developers of MfgApps with a development toolkit containing a common database, user interface, and so on, and a PnP Browser which manages and executes MfgApps. The MfgAppStore is the online marketplace on the Internet to register new MfgApps that have passed the peer review process and are made available to purchase by users of SMMEs. Each SMME is able to construct its own manufacturing information system using a PnP Browser with MfgApps selectively downloaded from the MfgAppStore. We are applying the prototype infrastructure to some SMMEs of the mold industry to validate the proposed concept.

J. Oh · B.-H. Kim (✉) · J.-Y. Baek
IT Convergence based Manufacturing System Research Group, Korea Institute
of Industrial Technology, Gyeonggi-do, Ansan-si, Sangnok-gu, Sa 1-dong 1271-18,
Cheonan 426-171, South Korea
e-mail: bhkim@kitech.re.kr

A. Azevedo (ed.), *Advances in Sustainable and Competitive Manufacturing Systems*,
Lecture Notes in Mechanical Engineering, DOI: 10.1007/978-3-319-00557-7_36,

1 Introduction

Small and medium manufacturing enterprises (SMMEs) in Korea, which include most domestic manufacturing enterprises, are supporting the substructure of value chains. Therefore, the growth of SMMEs should be accompanied for the sustainable growth of the manufacturing industry. However, until now, the manufacturing business in Korea has been developed centered mainly on large manufacturing enterprises. Due to that, SMMEs are relatively weak in technological power and productivity compared to larger manufacturing enterprises. Recently, in Korea, with employment and shared growth becoming the leading topic of conversation, universities, research institutes, and governments are attempting to improve the productivity and technological competitiveness of SMMEs. One of these attempts is to support SMMEs utilizing Korea's outstanding information technology (IT) [1].

Since utilization of the IT is closely related to national economic development, major countries of the world are promoting industrial innovation as a core policy task. That is, OECD countries are emphasizing the utilization of IT as one of four main propelling engines to improve the productivity of manufacturing enterprises, and the United States Department of Commerce recognizes the IT industry as a prime mover of the new economy (High Growth, Low Unemployment, and Low Price). As such, despite efforts for the reinforcement of IT related infrastructure at the national level, most SMMEs are still utilizing IT at an elementary level such as documentation or simple tasks, and most CEOs are still skeptical concerning the productivity improvements by utilizing IT. This is because when SMMEs introduce IT, they should consider various burdensome factors that inevitably follow, such as securing fully responsible personnel, the cost of continuous maintenance, and the adaptation to new IT systems. In particular, even though SMMEs introduce a high-priced and package-typed IT solution, they restrictively utilize only a part of the functions of the introduced system [2, 3].

As a method to overcome such realistic limits, this study proposes an information infrastructure with which SMMEs are able to establish an individualized information system (IIS) by selectively combining desired IT functions. With the infrastructure, SMMEs are able to establish the IIS by purchasing the functions required to perform actual work without purchasing a high-priced and package-typed IT solution.

This paper is composed as follows: Section 2 presents the definition and configuration elements of the infrastructure for constructing the individualized customized information system (ICIIS), and the model to be applied to SMMEs, which are dealt with in this study. Sections 3 and 4 describe the *Plug-and-Play* (*PnP*) *Platform and Manufacturing Application* (*MfgApp*), which are configuration elements of the ICIIS for SMMEs. Lastly, Sect. 5 presents the significance and effects of the result of this study.

2 Infrastructure for Constructing Individualized Information System

2.1 Components of ICCIS

The *Infrastructure for Constructing Individualized Information System* (*ICIIS*) proposed by this study signifies a hardware and software base required for individuals or enterprises to realize an IIS shown in Fig. 1. That is, the information infrastructure consists of: (1) IT devices, (2) application software as IT content; (3) operation platform which manages and operates the application software, (4) development tool of application software for developers, and (5) a business model as an intermediary connecting the user and the developer.

The IT device signifies the hardware in which the application software is installed and utilized, and the application software refers to a small scaled software function used by individuals or users in various enterprises. The operation platform is a supporting tool which comprehensively operates and manages the installation, modification history, interconnection, etc., of the application software. The development tool refers to the software development kit (SDK) which allows the application software to be operated at the operation platform consistently in the aspects of forms and functions. Lastly, the business model is an online market where the application software is able to be purchased and sold, where users are able to buy the necessary application software, and where developers are able to sell application software as a product.

2.2 ICIIS for SMMEs

The *ICIIS for SMMES* proposed in this study has been constructed based on the infrastructure shown in Fig. 1. The category and name of each configuration

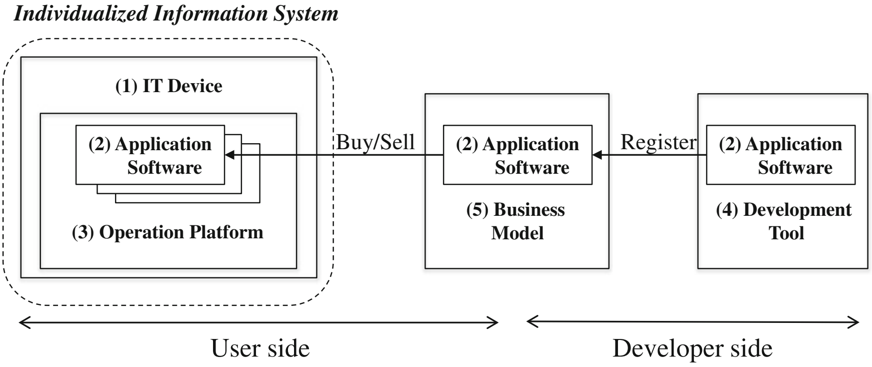


Fig. 1 The infrastructure for constructing individualized information system (ICIIS)

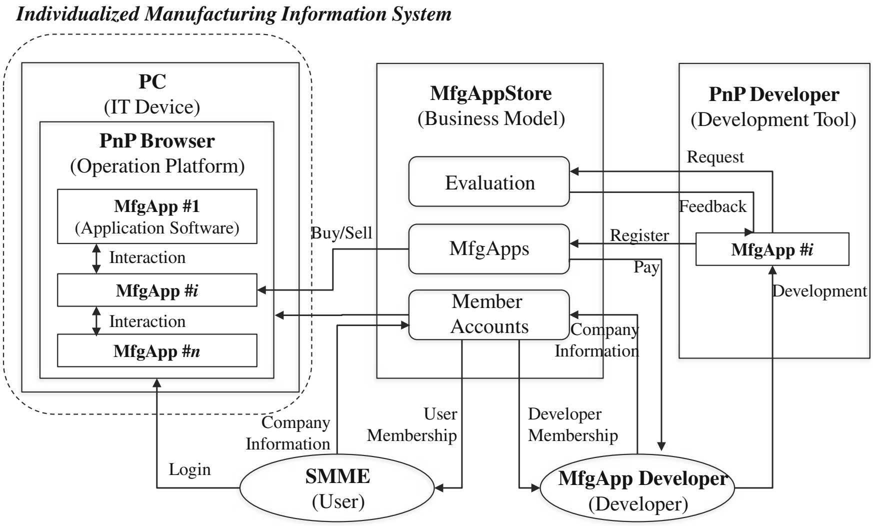


Fig. 2 The infrastructure for constructing individualized information system (ICIIS) for small and medium manufacturing enterprises (SMMEs)

element are as follows. The IT device is a general PC owned by most SMMEs. The Application Software is referred to as the *Manufacturing Application* (*MfgApp*) taking into account the characteristics of the manufacturing enterprise which serves as a customer. The range of the MfgApp is the collection of functions necessary for executing specific tasks inside an enterprise. As such, SMMEs are able to select only those MfgApps that are necessary according to the task execution range and utilize them as customized functions. The *Plug-and-Play* (*PnP*) *Browser* performing the role of the operation platform stores their history such as installation and upgrade, operates them in a consistent environment, and manages data related to them. The *Plug-and-Play* (*PnP*) *Developer*, a development tool for developers, allows the MfgApp to be developed easily and consistently. The aforementioned PnP Browser and PnP Developer are commonly called the *PnP Platform*, which will be described in greater detail separately in Sect. 3 since it is an important item in this study. The last element, the Business Model, is referred to as the *Manufacturing Application Store* (*MfgAppStore*), which is an online open market where the MfgApp is able to be purchased and sold. The MfgAppStore helps developers verify and register the MfgApp, and provides the functions by which SMMEs are able to easily search or purchase an MfgApp [4, 5].

As shown in Fig. 2, the basic flow of the ICIIS for SMMEs is as follows: A developer registers the MfgApp developed using a tool called PnP Developer to the MfgAppStore, and SMMEs run the MfgApp purchased from the MfgAppStore on the PnP Browser. In addition, the developer signs up to be a member of the MfgAppStore. Then, the feasibility and effectiveness of the MfgApp developed by the developer are evaluated. If the MfgApp passes evaluation, it is registered to the

MfgAppStore. On the other hand, SMMEs sign up to be the member of the MfgAppStore and use the MfgAppStore and PnP Browser with the account provided from the MfgAppStore. SMMEs perform login to the MfgAppStore to buy the desired MfgApps and login to the PnP Brower to utilize the purchased MfgApps freely.

2.3 *Difference of Information Infrastructures Between the Individual and the Enterprise*

In the case of the information infrastructure described in the previous section, the details of the configuration elements differ depending on the object, even though the overall concepts are the same. The representative infrastructure for individuals is a mobile-based platform, which corresponds to Apple's App Store and Google's Android Market model. The common point of the two cases is that an infrastructure is constructed which allows an individual to buy and download the desired application software from an online market.

Table 1 shows the comparison and description of the aforementioned information system construction environment for individual users and SMMEs in the aspects of five configuration elements. As shown in Table 1, the two infrastructures exhibit difference in three aspects. Firstly, the configuration and characteristics of the application software vary due to the fact that the users to be applied are different. That is, in the case of the application software for an individual, its contents are configured according to the preference and personality of a target individual customer, while in the case of the application software for an enterprise, its contents are configured taking into account the tasks and production site characteristics of a manufacturing enterprise. Secondly, the IT devices used by users are different. The information infrastructure for an individual is constructed based on the mobile device owned by an individual while the application software for an enterprise is constructed based on a PC used most commonly by SMMEs.

Table 1 Difference in configuration elements for the customized informatization

	Customized informatization for individual users		Customized informatization for SMMEs
	Apple	Google	
(1) IT device	Smart device (iPhone, iPAD)	Smart device (Android Phone, Tab)	PC
(2) Application software	Application	Application	Manufacturing applications
(3) Operation platform	iOS	Android	Plug-and-play browser
(4) Development tool	iPhone SDK	Android SDK	Plug-and-play developer
(5) Business model	AppStore	Google Play	Manufacturing application store

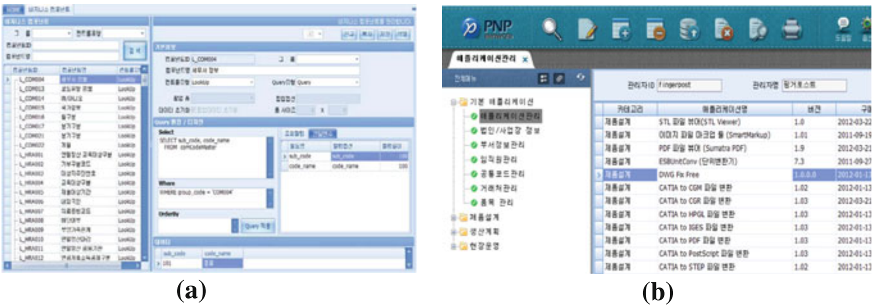
Thirdly, there is a problem with information links between application software. In the case of the information infrastructure for an individual, since there are not many links between application software in the mobile device, most application software is independent, while in the application software for an enterprise, the link between applications should be considered importantly due to the close relation between the internal tasks of individual enterprises [6, 7].

3 PnP Platform

As mentioned in the previous section, the PnP Platform is divided into the PnP Developer, which is a platform for MfgApp developers, and the PnP Browser, which is a platform for MfgApp users and SMMEs. Figure 3 shows the image screens of the PnP Developer and PnP Browser implemented as an example by this study. This section provides a rough description of the PnP Developer and PnP Browser.

3.1 PnP Developer

The PnP Developer is a development tool kit providing a development library and function to allow the MfgApp to be developed consistently. Table 2 shows four major modules provided by the PnP Developer. The UI development module is used when designing the user interface (UI) of the MfgApp. Through that module, the source code of the program is automatically created according to the basic screen design. In other words, if a developer defines the major information of the MfgApp and designs the UI, the UI development module automatically creates the source code by binding the major defined information and UI information. In addition, since it is possible to process database input/output using that module, a developer is able to easily implement the initial screen of the target MfgApp. The



관리자ID fingerpost 관리자명 핑거포스트

카테고리	애플리케이션명	버전	구
제품설계	STL 파일 뷰어(STL Viewer)	1.0	2012-03-22
제품설계	이미지 파일 마크업 툴 (SmartMarkup)	1.01	2011-09-19
제품설계	PDF 파일 뷰어 (Sumatra PDF)	1.9	2012-03-21
제품설계	ESBUnitConv (단위변환기)	7.3	2011-09-27
제품설계	DWG Fix Free	1.0.0.0	2012-01-1
제품설계	CATIA to CGM 파일 변환	1.02	2012-01-13
제품설계	CATIA to CGR 파일 변환	1.03	2012-03-21
제품설계	CATIA to HPGL 파일 변환	1.03	2012-01-13
제품설계	CATIA to IGES 파일 변환	1.03	2012-01-13
제품설계	CATIA to PDF 파일 변환	1.03	2012-01-13
제품설계	CATIA to PostScript 파일 변환	1.03	2012-01-13
제품설계	CATIA to STEP 파일 변환	1.02	2012-01-13

Fig. 3 PnP platform: **a** PnP developer, **b** PnP browser

Table 2 Major functions of the PnP developer

Major module	Detail function
UI development module	Creates internal screen information of the MfgApp Screen UI design UI information binding Creates the program source code automatically
Common API module	MS Excel interlocking method Data format processing method
Screen template management module	Provides the basic screen template Connects the UI development module
Integrated development management module	Manages development environment Manages terminology for work Manages database objects

common Application Programming Interface (API) module provides a set of the functions of the MfgApps frequently used or complicated to improve the software development efficiency. The PnP Developer provides approximately 100 common APIs including MS Excel interlocking method, data format method, etc. The Screen Template Management Module is a function which provides MfgApp screens by patternizing some of them. It improves the development standardization and productivity, and provides a consistent program user environment. Lastly, the Integrated Management Module inputs the MfgApp development process of 'Analysis → Design → Development → Distribution → Post Management' in a database for its systematic management. It is usefully utilized for the development and post management of the MfgApp.

3.2 *PnP Browser*

The PnP Browser is a tool which performs integrated operation and management of the selected MfgApps. All MfgApps are run by the PnP Browser. The PnP Browser has three major functions: (1) MfgApp Operation Management, (2) MfgApp Version Management, and (3) Basic MfgApp Management. The MfgApp Operation Management Function allows all MfgApps purchased from the MfgAppStore to be used under the same environment. If the MfgApp purchased by a customer is updated, the MfgApp Version Management Function automatically connects the MfgApp to the MfgAppStore to update it to the latest version. Therefore, it is not necessary for a user to check version information one by one in order to update the MfgApp purchased. Lastly, the PnP Browser provides embedded MfgApps necessary for SMMEs to perform basic work. Currently, it provides 10 types of basic MfgApps. Those basic MfgApps were developed to prevent duplication of a basic function and to perform integrated management of master data for common use. Table 3 shows the list of basic MfgApps supported by the PnP Browser.

Table 3 List of basic MfgApps

No.	Name	Description
1	Application management	Performs integrated management of MfgApps purchased from the MfgAppStore
2	User management	Manages the information of logged in users who utilize the PnP Browser
3	Common code management	Performs integrated management of the common code information used by the MfgApp
4	Corporation/ workplace management	Manages information on corporations and reported workplaces
5	Department information management	Manages department information including department name, department manager, etc
6	Employee management	Manages basic information including the name, address, position, etc., of officers and staff
7	Customer information management	Manages the basic information including the company name, contact point, address, etc. of customers
8	Item information management	Manages item information including the item name, model, unit, size, etc
9	Project management	Manages basic information including the project name, period, PM, etc
10	Warehouse management	Manages warehouse information including warehouse name, as well as information as to whether it is used or not

4 Interaction Between MfgApps

4.1 MfgApp and Database

Since various works performed inside an enterprise interact with each other according to a specific process, the information interaction between MfgApps is very important to construct a customized information system for SMMEs. That is, MfgApps need the process to share and reprocess some pieces of information in order to perform work inside an enterprise. The interaction between material procurement management and stock management works is an example of the information interaction between MfgApps. In general, after purchasing materials, the person in charge of procurement management hands over the information including the name, purchase quantity, purchase date, warehoused date, etc. The person in charge of Stock Management checks for the storage, release, and return of the materials with the information, and performs inquiry about materials in shortage.

To achieve information interaction between MfgApps, it is necessary to share information at the database level. To do so, the database should be first established. However, it is significantly burdensome for SMMEs to establish a database independently. For these SMMEs, this study proposes a cloud service structure that can provide such a database server online. As a matter of fact, it may be required for enterprises to maintain an independent server for reasons of security.

Table 4 Database type

Database name	Name of internal table
Core database	MfgApp information, customers, user accounts, etc
Basic database	Departments, employees, customers, items, etc
Extended database	(Added and named by a user as necessary)

Therefore, this study designs a cloud service so that a database server is able to be separately installed inside an enterprise. Table 4 shows the types of databases designed under such a structure. The databases including MfgApp information are divided into three types: The *Core Database* stores information necessary for classifying the MfgApp purchased by specific users among MfgApps. Therefore, user account and MfgApp table purchased by the user, etc., are located in the database. Here, in order for the MfgApp to be run on the PnP Browser, these tables should be referred to. The *Basic Database* includes master data which is able to be used in duplication in multiple MfgApps, where tables of departments, employees, items, etc., are located. Lastly, the *Extended Database* stores characteristic information needed by the MfgApp in addition to the master data included in the Basic Database. Accordingly, in the *Extended Database*, when a new MfgApp is registered, a new table is created additionally, or some of existing tables are updated. The newly created table is named by the developer according to the Naming Rule guideline.

Table 5 shows the way the internal table of each database type is referred to the MfgApp. The types of the MfgApps shown in Table 5 are divided into two: Firstly, the *Basic MfgApp* is, as described in Sect. 3.2, the MfgApp built in the PnP Brower, which does not need to be purchased from the MfgAppStore. On the contrary, the *Selective MfgApp* is the MfgApp purchased from the MfgAppStore by SMMEs. Those two types of MfgApps refer to all tables of the Core Database in order to identify specific MfgApps. In the case of the Basic Database, the two types of MfgApps refer only to some of the tables needed to run them. The *Selective MfgApp* refers only to the tables of the Extended Database created together when it is developed. The MfgApp has more than two tables which are able to be managed directly. In the case of such tables which are able to be directly managed, a user is able to read and write data in the corresponding tables through specific MfgApps. In addition, even in the case of tables which are not to be managed directly by the MfgApp, it is possible to read the corresponding data

Table 5 Tables referred to for each MfgApp

	Basic MfgApp	Selective MfgApp
Internal table of core database	Refer to all tables (possible to read data)	Refer to all tables (possible to read data)
Internal table of basic database	Refer to some tables (possible to read and write data)	Refer to some tables (possible to read data)
Internal table of extended database		Refer to some tables (possible to read and write data)

through the API opened within the allowable range so that the aforementioned information interlocking between MfgApps is able to be achieved. The Basic and Selective MfgApps directly manage the designated tables in the Core and Extended Databases, respectively.

4.2 Plan for Interaction Between MfgApps

As shown in Fig. 4, the interaction between MfgApps is divided into two. The α and τ in Fig. 4 show the tables of the MfgApp and Database, respectively. In the case of the plan for *the Type 1 Interaction*, the Selective MfgApp refers to the tables of the Basic Database managed by the Basic MfgApp. That is, if the basic information on departments, employees, customers, items, etc., is stored in the corresponding tables through the Basic MfgApp, the Selective MfgApp refers to the information. Developers refer to the desired basic information through the open APIs of the PnP Developer. In the case of the plan for *the Type 2 Interaction*, one of the two Selective MfgApps developed by the same company refers to the other MfgApp. This means that a developer internally uses the APIs through which the information of a specific table can be referred to, without opening it externally. That is, a developer can sell several MfgApps together by emphasizing that

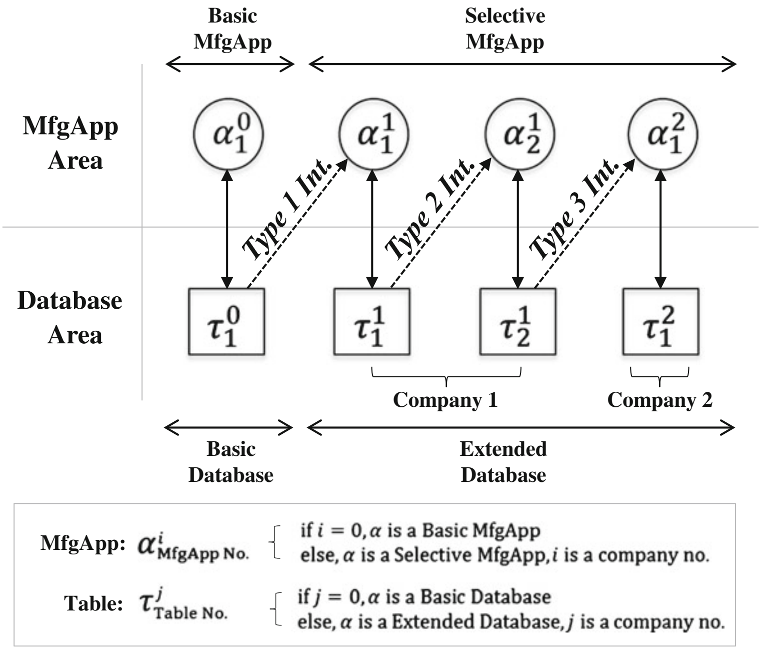

Basic MfgApp
Selective MfgApp
MfgApp Area
Database Area
α_1^0
α_1^1
α_2^1
α_1^2
Type 1 Int.
Type 2 Int.
Type 3 Int.
τ_1^0
τ_1^1
τ_2^1
τ_1^2
Company 1
Company 2
Basic Database
Extended Database
MfgApp: $\alpha^i_{\text{MfgApp No.}}$
if $i = 0$, α is a Basic MfgApp
else, α is a Selective MfgApp, i is a company no.
Table: $\tau^j_{\text{Table No.}}$
if $j = 0$, α is a Basic Database
else, α is a Extended Database, j is a company no.

Fig. 4 Three plans for information interaction between MfgApps

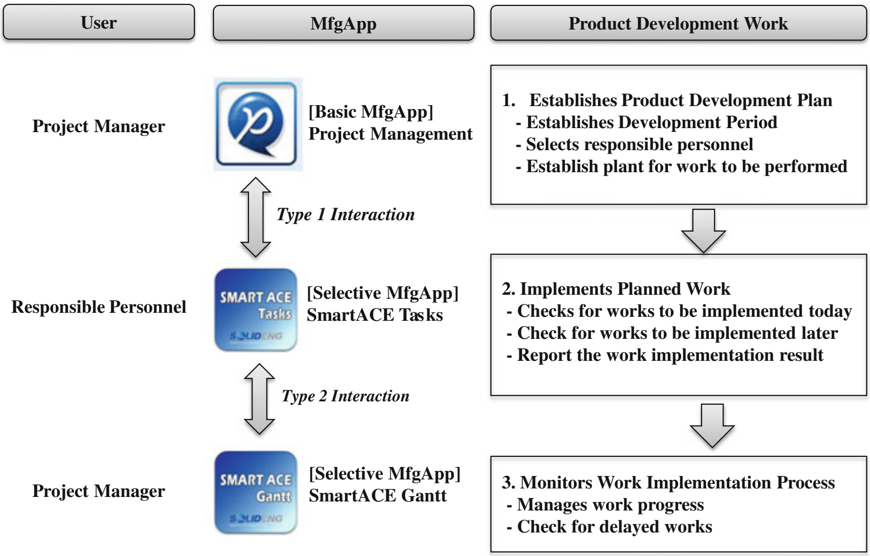

User
MfgApp
Product Development Work
Project Manager
[Basic MfgApp]
Project Management
1. Establishes Product Development Plan
- Establishes Development Period
- Selects responsible personnel
- Establish plant for work to be performed
Type 1 Interaction
Responsible Personnel
SMART ACE Tasks
[Selective MfgApp]
SmartACE Tasks
2. Implements Planned Work
- Checks for works to be implemented today
- Check for works to be implemented later
- Report the work implementation result
Type 2 Interaction
Project Manager
SMART ACE Gantt
[Selective MfgApp]
SmartACE Gantt
3. Monitors Work Implementation Process
- Manages work progress
- Check for delayed works

Fig. 5 MfgApp utilization scenario at the time of product development

multiple MfgApps it developed can be interlocked. In the case of the plan for *the Type 3 Interaction*, a specific Selective MfgApps refers to an MfgApp developed by other companies. Since the plan has problems of security and disclosure of development know-how, it requires effort to solve these problems as well as the strategic tie-up between developers. These three plans for the interaction between these MfgApps have many problems to be solved technically and strategically as the interaction plan moves from Type 1 to Type 3.

4.3 Scenario Using MfgApps

Figure 5 shows a virtual scenario where a manufacturing enterprise utilizes the MfgApp for product development. In the virtual scenario, the product development work is divided into: (1) Establishment of Product Development Plan; (2) Execution of Planned Work; and (3) Work Execution Process Monitoring. The project manager (PM) and responsible personnel can utilize the MfgApps for each of their tasks divided into three types. First, the project manager establishes a product development plan including establishment of development period, selection of responsible personnel, a plan for work to be implemented, etc., with the Project Management, a Basic MfgApp. If the project manager establishes a product development plan, detail tasks are implemented according to the established plan. At that time, the person responsible for each task checks the tasks to be performed today and later with *SmartACE Tasks*, a Selective MfgApp and inputs the task implementation results to report it to the project manager. The project manager checks the task implementation process of each responsible person and its result with *SmartACE Gantt,* a Selective MfgApp. With that, the project manager checks the progress and delay of tasks and performs project management so that product development work can be implemented smoothly. As such, the information interlocking between MfgApps is necessary in order for work at the time of product development to be implemented successively. The Type 1 Interaction was made between the MfgApps, Project Management, and SmartACE Tasks and the Type 2 Interaction was made between the SmartACE Tasks and SmartACE Gantt.

5 Conclusions

This paper proposed an ICIIS as a method to improve the productivity of SMMEs. The proposed infrastructure consists of five elements: General PC, MfgApp, PnP Browser, PnP Developer, and MfgAppStore. In addition, unlike the information infrastructure based on existing mobile systems, in the case of the proposed infrastructure, information interlocking between MfgApps was dealt with as an important issue. As such, this study presented the structure and three information interlocking plans by which work data between the MfgApps is able to be shared at the database level.

Since SMMEs are able to construct an IIS by selecting only the necessary IT functions through these infrastructures, the investment cost is relatively low and little time is required compared to existing package based IT solutions. The result of this study will have a direct influence on SMMEs and a positive influence on the IT industry due to the expansion of the pool of designers and developers who develop MfgApps. As a result, this study has a great significance in that both SMMEs using MfgApps and developers developing MfgApps are able to create a new form of informatization ecosystem to be profitable.

References

1. Black SE, Lynch LM (2001) How to compete: the impact of workplace practices and information technology on productivity. Rev Econ Stat 83(3):434–445
2. Reuther D, Chattopadhyay G (2004) Critical factors for enterprise resources planning system selection and implementation projects within small to medium enterprises. Engineering management conference 2004, proceedings of 2004 IEEE international, 2, 2004
3. Markets and Markets, Industrial Controls (SCADA, PLC, DCS) & Factory Automation (Field Devices, MES, ERP) Market: Global Forecast and Analysis Till 2016, 2012
4. http://mfg-app.co.kr. Accessed 20 July 2012
5. http://mfg-app.co.kr:8088/index.htm. Accessed 20 July 2012
6. https://developer.apple.com/devcenter/ios/index.action. Accessed 20 July 2012
7. http://source.android.com. Accessed 20 July 2012