Development of Construction Planning Tool by Integration of Application Data

Takanori Terashima, Assistant Professor
School of Project Design, Miyagi University, Japan
terashima@myu.ac.jp

Nami Amarume, Undergraduate Student
School of Project Design, Miyagi University, Japan

Koji Makanae, Professor
School of Project Design, Miyagi University, Japan
makanae@myu.ac.jp

Nashwan Dawood, Professor
School of Science and Technology, Teesside University, UK
n.n.dawood@tees.ac.uk

ABSTRACT: *This paper reports the implementation of a system that visualizes the construction-planning process using 3D modeling data and schedule data to analyze construction planning. In the past, studies have reported the benefits of visual 4D planning in which the 3D modeling data combined with process schedule data for work progress control. The proposed methodology offers rapid visualization of the work performance with scheduled activity and simplifies the construction planning and schedule inspection. Consequently, it is expected to increase productivity and reduce the need for reworking the plan. However, even an innovative construction company will not readily adopt the new methodology unless the benefits outweigh the cost and effort. There are case examples that use 4D simulation or animation to show how a building looks or show the progress of the construction. A new model or even a new data structure might be required for the 4D demonstration. We aim to develop a system that integrates several types of data and enables simulation of the construction progress by displaying 3D models according to the activity schedule. The system has the following requirements: it must 1) import and display 3D modeling data, 2) import the project schedule, 3) link each model and activity, 4) give material data for each object to enhance the realism, and 5) show the cost accumulation. These functions should use previously reserved resources. Thus, the system should be able to import DXF-format files for 3D modeling data, and access MDB-format database for the project schedule and the costs. These data format are apparently the most widely used in this context.*

KEYWORDS*: construction planning, 4D CAD, virtual reality, data integration, congestion control*

1. INTRODUCTION

This paper presents the implementation of a construction-planning tool using 3D modeling data and schedule data as an application of ICT technology in the field of construction. In recent times, the construction industry has attempted to improve its production system by integrating procedures ranging from design to construction and therefore promoting the use of 3D CAD systems (Nagao et al. 2005). The aim of the recent 3D CAD technology is to provide a platform to unify the data types shared by project members, including data related to design, construction, and facilities (Yabuki et al. 2004); earlier CAD technology could be used to obtain only data related to the intended design at an early planning stage or to obtain a consistent drawing at the final planning stage.

Technical issues at the construction stage are now considered from the early design stage. Combining design with production in such a manner, which is referred to as *production design*, allows the user to demonstrate at an early stage the estimated progress in the work as well as the amount of completed work. It is also possible to reduce the reworking and redesigning at the post-processing stage since visualization at an early stage helps to identify technical problems. This methodology requires various types of applications, including a CAD system, to design, plan, manufacture, and manage the process. However, it should be noted that the interchangeability of shared data between the software tools is not sufficiently high, and thus, the overall efficiency of the project is not improved considerably. The product model is being developed as a solution that allows interoperation among different systems (Yabuki et al. 2005a, 2005b; Kawanai 2005). The product model is a generalized data model that expresses 3D shapes, and it provides information about the attributes of each element in a structure. The data set

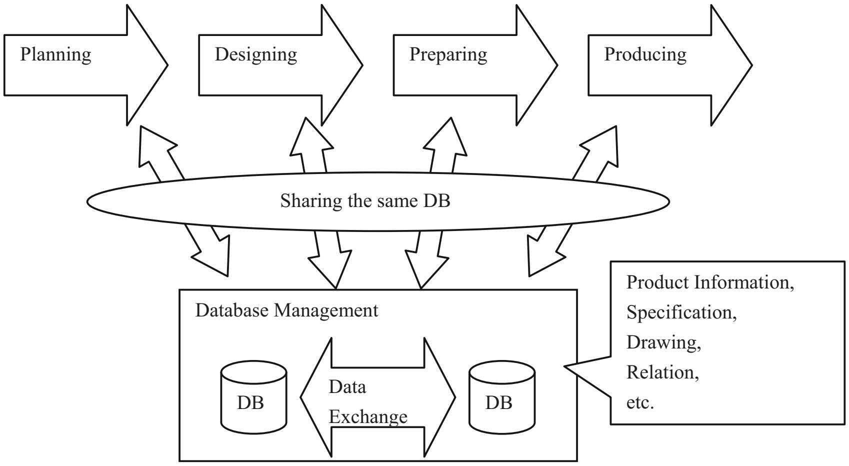


Fig. 1 Data Sharing Using Product Model.

can be shared by various systems or applications if the data set is created as a product model, as shown in Fig. 1.

Previous studies have reported the advantages of architectural methodologies that use 4D data to associate 3D modeling data with the schedule data for the construction. Despite the advantages of this method, even major construction companies do not use it. In fact, 2D drawings are still widely used for design and construction planning. Although a 2D drawing is sufficient for a designer, it is difficult for less-experienced people to visualize the 3D aspects using 2D drawing, and the problems cannot be simulated or analyzed on a PC. However, the conventional methods will not change unless the new approach can be clearly shown to offers benefits that outweigh the cost and effort involved. Considerable cost is involved in using the new methodology to produce a new set of data structures such as the product model, simulate the assembly process on an advanced CAD system, or animate the arrangement of objects in a high-performance 3D graphic designer. Even though each software tool provides multiple functionalities, the applications currently in use are all independent and specific, such as CAD for the design of 3D models and a project manager for the scheduling. The product model is very promising solution but developing a large amount of reusable data is essential otherwise specific data structure will have to be developed for each project all the time. Therefore, a system that integrates the outputs from each application is required for the implementation of the new methodology before the use of the product model.

Methods involving several applications have been introduced (Dawood et al. 2001, 2002, 2005, 2006; Sriprasert et al. 2003; Vries et al. 2007; Kang et al. 2006; Kano et al. 2002, 2003). Some require the original data structure and others simulate a specific project (http://www.bentley.co.jp/info/usercase/nishimatsu.htm). In this study, a system that integrates the data from several applications is proposed, whereas in other studies, the integration of the applications themselves has been suggested. Our approach is generally applicable for any case since the system accepts the data files created by several types of applications. The main component of the system is the visualization of the process control using a 3D shape linked to the schedule. The visualization not only shows the state of construction but also simulates the entire construction field. This is done by simulating heavy machinery and material carriers along with their route and temporary storages in order to examine the entire construction plan in advance. Although the data are created by different applications, the proposed system can link data from each dataset. It thus provides the same advantages as 4D CAD without the need to create a product model or a new type of data structure. Therefore, users can work with generic applications such as CAD and Microsoft Project (MS-Project) in the usual manner.

In this paper, we describe the implementation of software that simulates the progress in the work by visualizing the given drawing data and linking them with the schedule data.

2. DESIGN OF THE CONSTRUCTION-VISUALIZING TOOL

2.1 Software specification

Some tools implement the animation in AutoCAD using VBA or AutoLisp and others implement it in an advanced CAD system. The suggested system enables importing the model data contrary since data exchange without an expensive CAD system is widely required. This design is advantageous to both users and developers. Users can handle any implemented data type, and it is easy to supply a function to import other data formats since the system is independent of specific applications. It is assumed that the system needs to perform the following functions, integrating the resources by linking related material:

1. Reading drawing data
2. Reading schedule data
3. Linking components and schedule
4. 3D visualizing
5. Displaying progress in the work at any given time
6. Displaying the cost accumulation
7. Setting the textures for the components
8. Saving/reading work

Functions 1 and 2 involve importing data created by other applications. The other functions involve the processing of imported data. Functions 5–8 are the original functions. They show the construction status, the accumulating cost at any specified time, or illustrate the status transformation over time after linking the different types of imported data which is modeling data and schedule data, and additionally material/texture data in order to enhancing the reality. “Work” in function 8 implies the work on the proposed system and not the work in a construction project. Function 8 is important because any information in a data file is quite large and the users’ work should be reproducible. The proposed system is expected to use both the drawing data created by AutoCAD and the schedule data created by MS-Project. The system should look like a general Windows application; further, users should be able to navigate the 3D scene using a mouse. The system overview is shown in Fig. 2.

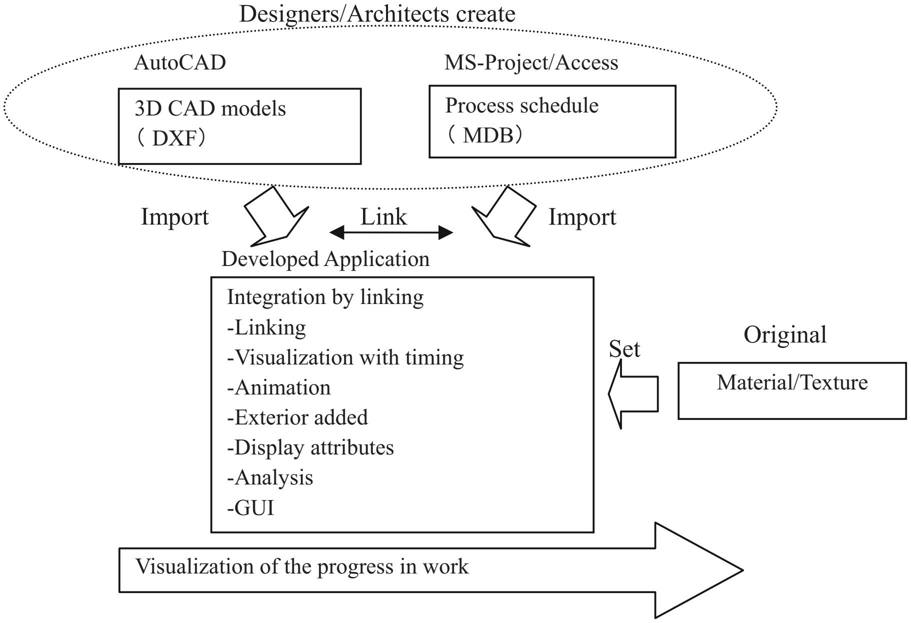


Fig. 2 System Overview.

2.2 Development Environment

The system is developed using Microsoft Visual C++ (hereafter VC) and DirectX SDK. Alternative languages include Visual Basic and Java, and alternative graphics engines include OpenGL and Java3D. Our choice was based on the need to handle huge data sets, and the fact that VC offers a Windows-like environment. The system is developed based on the FormView class so that additional functions can be represented by icons. It is also based on the Document-View architecture. The imported data are stored in the CDocument class and the CView class controls the visualization. Various data classes are defined for each data type and the system instantiates the appropriate data class in the CDocument class based on the type of the imported items. These data classes are referenced from the CView class and selectively rendered according to the time schedule. The relation among the classes is shown in Fig. 3. The principal functions are described in the following subsections.

2.3 Implementation

2.3.1 Importing the model data

AutoCAD creates two types of data: DWG and DXF. DWG is binary whereas DXF is text and therefore easier to handle. DXF is the drawing interchange file format; its specification has been published, and it is widely used for data exchange between applications that deal with models or drawings. The system must be able to process a DXF file.

Each DXF file consists of seven sections: HEADER, CLASSES, TABLES, BLOCKS, ENTITIES, OBJECTS, and THUMBNAILIMAGE. The ENTITIES section contains information about the shape. The data in this section are important for visualizing the model. There are entities that describe a specific figure such as a circle or a curve, with a layer name and the coordinates of the vertices. There are approximately 40 types of elements, as shown in Fig. 4. A preliminary survey indicates that the shapes defined in ENTITIES are usually one of four

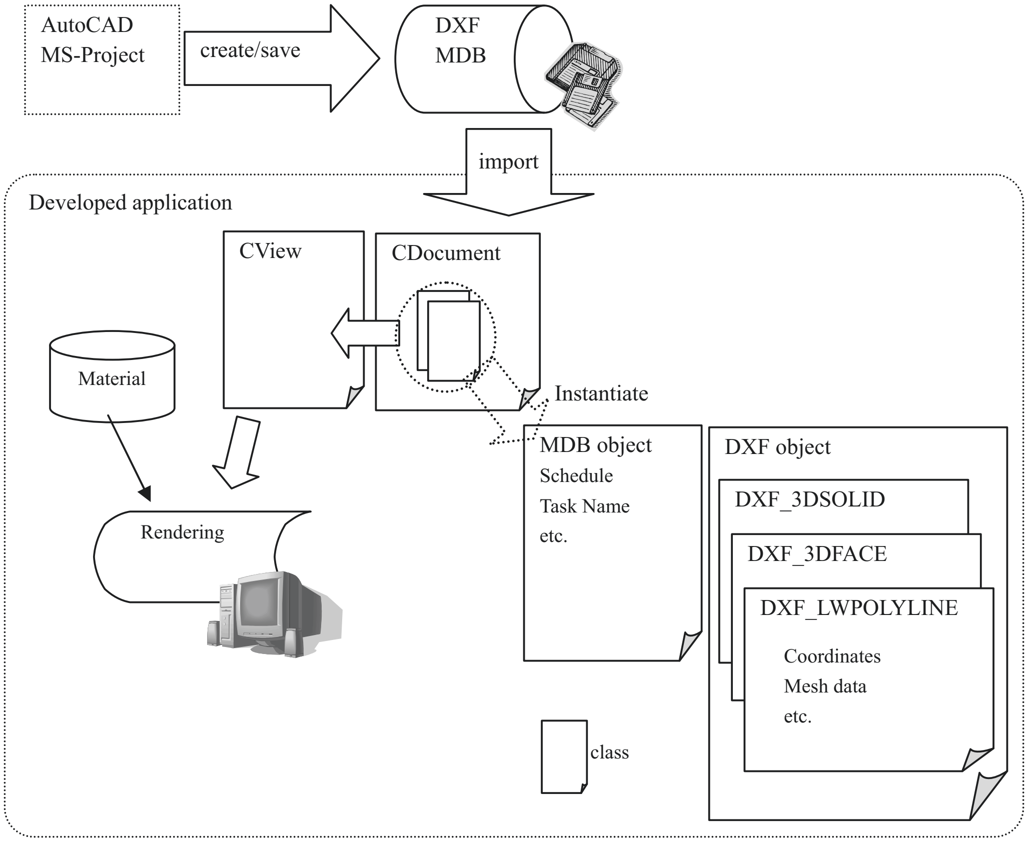
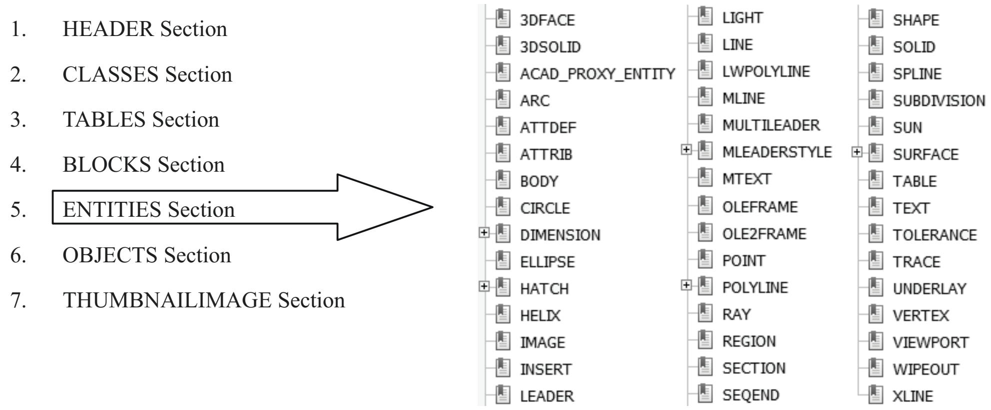


Fig. 3 Structure of Classes.



1. HEADER Section
2. CLASSES Section
3. TABLES Section
4. BLOCKS Section
5. ENTITIES Section
6. OBJECTS Section
7. THUMBNAILIMAGE Section

3DFACE
3DSOLID
ACAD_PROXY_ENTITY
ARC
ATTDEF
ATTRIB
BODY
CIRCLE
DIMENSION
ELLIPSE
HATCH
HELIX
IMAGE
INSERT
LEADER
LIGHT
LINE
LWPOLYLINE
MLINE
MULTILEADER
MLEADERSTYLE
MTEXT
OLEFRAME
OLE2FRAME
POINT
POLYLINE
RAY
REGION
SECTION
SEQEND
SHAPE
SOLID
SPLINE
SUBDIVISION
SUN
SURFACE
TABLE
TEXT
TOLERANCE
TRACE
UNDERLAY
VERTEX
VIEWPORT
WIPEOUT
XLINE

Fig. 4 Seven Sections in DXF File and Entity List in ENTITIES Section.

types: REGION, 3DFACE, 3DSOLID, and LWPOLYLINE; typically, the LWPOLYLINE shape is used. The processing of the drawing data is currently limited to these four shapes.

The system reads DXF files line by line and instantiates a class associated with each entity style when it finds the shape's definition within the ENTITIES section. Each class stores the coordinates, layer name, and so on. The imported shapes are rendered as a set of meshes obtained by defining these coordinates. The layer names are listed in an array (actually the vector class), and the classes are listed in an array together with the shapes, such as LWPOLYLINE or 3DSOLID. All the components belong to one of the layers. A set of components drawn on the same layer is considered to be the set needed by a particular construction process.

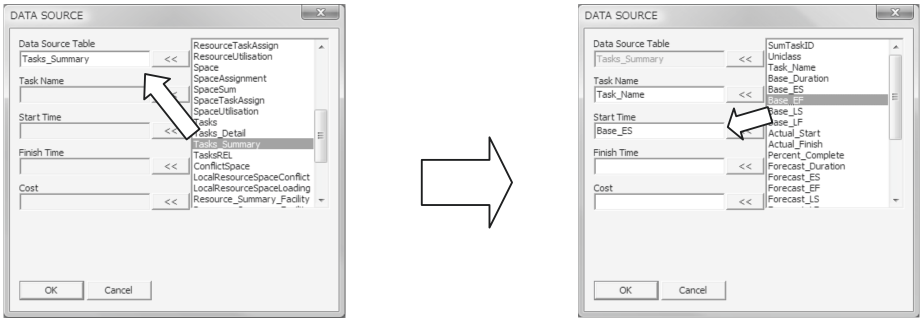
VPORT and LAYERS in the TABLE section are also essential. VPORT is used to determine the line of sight, and LAYERS is used to fetch the layer-name list for the shapes. All components of the structure belong to one of these layers. We assume that the component data that are related to a particular construction are drawn on a layer. When a layer is linked to a process in the construction plan, all the components drawn on the same layer are linked to that process. Therefore, an animation that changes process by process equivalently changes layer by layer.

2.3.2 Importing the schedule data

It is assumed that the process schedule is developed using MS-Project. The system currently requires a file exported by MS-Project that is in the Microsoft Access (MS-Access) format. The system accesses the elements stored in the database via an ODBC (open database connectivity) driver. The software is unable to access an MS-Project file directly since the relevant ODBC driver is not provided. The exported MS-Access file contains several databases, and one has a table that contains the process schedule. The table has fields that describe the task name, start date, finish date, and cost. The system can read them and modify them directly since it uses ODBC and can execute an SQL command. Screenshots of this operation are shown in Fig. 5.

2.3.3 Linking components and schedule

After the model data and schedule data are imported, the layer names and task names are listed. The component names are listed under their layer names. It is necessary to link task names in the schedule data and the modeling data. A task name should be reasonable since it is used for process management. Modeling data are created in the same layer for each process flow. The linking operation is as follows: choose one entry from the task-name list with a click and then one from the layer-name list, and then link them with a right-click. The layer that has to be scheduled, including the construction components, then appears under the process name. Fig. 6 shows this operation. This operation is carried out as many times as required by the process flow. A user should be able to save and reproduce these links because the number of processes is considerable when the structure to be built is large.



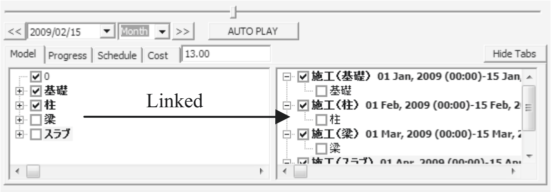
DATA SOURCE

Data Source Table: Tasks_Summary
Task Name:
Start Time:
Finish Time:
Cost:

DATA SOURCE

Data Source Table: Tasks_Summary
Task Name: Task_Name
Start Time: Base_ES
Finish Time:
Cost:

Fig. 5 Selection of Table (left) and Fields (right).



<< 2009/02/15 Month >> AUTO PLAY

Model | Progress | Schedule | Cost | 13.00

☑ 0
☑ 基礎
☑ 柱
□ 梁
□ スラブ

☑ 施工（基礎） 01 Jan, 2009 (00:00)-15 Jan,
□ 基礎
☑ 施工（柱） 01 Feb, 2009 (00:00)-15 Feb, 2
□ 柱
☑ 施工（梁） 01 Mar, 2009 (00:00)-15 Mar, 2
□ 梁

Fig. 6 Linking of Layer and Activity.

2.3.4 Visualization

When the system reads a DXF file, the coordinates are fetched for each component and constructed as a set of meshes. Each mesh is made up of triangles; hence, the displayed model is composed of a number of triangles because of the display mechanism of the hardware. A triangle mesh gives the best performance on the hardware used for the system.

The model displayed after the linking of layer and schedule shows the progress in the work at any given time. Some components are already constructed, others are in progress, and others are not yet started. They are distinguished by color. If a date is given by the user, the components that are not yet started on that date are not displayed, those under construction are displayed in red, and those completed are displayed in blue. The scene that displays the construction can be navigated by a mouse, and it is possible to examine the scene from any angle and any distance.

If the date is changed in a certain interval, the progress is animated as shown in Fig. 7. The resource list is shown on a tab. Clicking the other tabs shows other information such as the progress in the work, the cost, or the materials list, as shown in Fig. 8. The scene can be changed by changing either the display colors or the material attributes. When the tab is set to the materials list, the components and materials can be linked with the same operation as that used for linking components and tasks.

2.3.5 Setting the material for the components

DXF files have settings for colors, but they are not always used because they are not necessary at the design stage. Therefore, the system uses only mono color with gradation; however, the color differs according to the completion status. The displayed scene may be more realistic when each component has a material attribute. It is important to animate the design for process examination or analysis. If the system accomplished this, it would be a useful tool for the presentation of an area of interest. The DXF format does not support textures and material attributes so the material library must be stored in a different format.

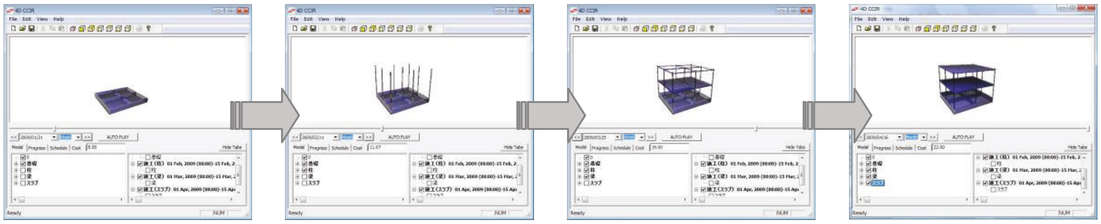


Fig. 7 Animation of Progress in Construction .

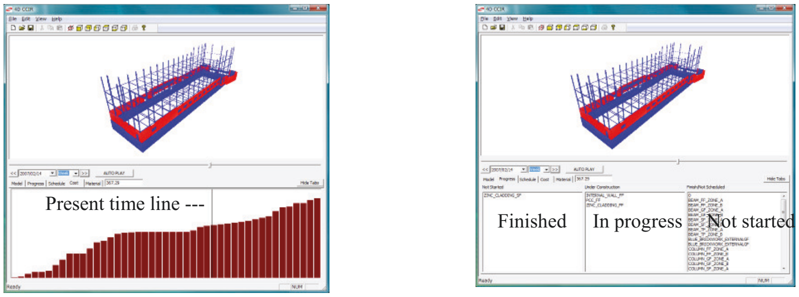


Fig. 8 Changing Tab to Show Cumulative Cost (left) and Progress in Work (right).

2.3.6 Saving and reading files

The system stores the source files, name of the database used, table names, field names, linkages of task names, and layer names in a text file. Each factor is separated by a text separator. When this file is read, the user changes are duplicated according to the information in the file. Currently, the text file lists each instruction performed by the user and the system carries out each of them.

3. DISCUSSION

The system can currently import model data and schedule data, visualize them, create an animation according to the process plan using materials/textures if the library is well organized, and display the cost accumulation. The construction status over time can be visually confirmed. All the construction processes can be displayed in the animation, and it is possible for the display to represent a specific time. The construction process plan can be examined and analyzed by performing simulations in advance. Smooth communication can be achieved among the designer, the purchaser, and the workers on the site. The display will be more realistic when the material of the components can be set.

The system can: 1) assess the completeness and accuracy of the plan, 2) communicate the progress, 3) visualize the progress, 4) visualize the direction of the work, and 5) identify conflicts (process and products). However, the biggest problem thus far is that the original processing of the model data cannot support all the elements that the DXF format defines. The system currently renders only a few shapes. Using a third-party SDK for DXF, e.g., Open Design Alliance’s OpenDWG (http://www.opendwg.org/), is probably appropriate. A material library is also important to make the model realistic, but this would require considerable work.

4. CONCLUSION

This paper has presented the implementation of an application for construction sites. The application provides an interface to link 3D model data with the schedule data so as to visualize a construction scheme in 4D without a new data structure such as that of the product model.

In future, we plan to implement a function that can visualize temporarily used heavy vehicles. We will also incorporate background images and terrain data since the MS-Project data usually includes such information. We will examine the correspondence with the surroundings and simulate an overall architectural plan, including an examination of the delivery vehicles, installation positions, space for heavy equipment, temporary storage of materials, and placement of orders. We aim to derive the benefits of 4D CAD using the resource files that are generally available at production sites.

5. REFERENCES

Dawood N., Scott D., Hobbs B., Mallasi Z. (2005). The Virtual Construction Site (VIRCON) Tools: An Industrial Evaluation, ITCon, Vol. 10.

Dawood N., Sikka S. (2006). The Value of Visual 4D Planning in the UK Construction Industry, Springer LNCS 4200, 127-135.

Dawood N., Sriprasert E., Mallasi Z., Hobbs B. (2001). Development of an Integrated Information Resource Base for 4D/VR Construction Process Simulation & Visualization, Proceedings of CONVR 2001.

Dawood N., Sriprasert E., Mallasi Z., Hobbs B. (2002). 4D Visualisation Development: Real Life Case Studies, Proceedings of CIB W78 conference.Kang L.S., Jee S.B. (2006). 4D System for Visualizing Schedule Progress of Horizontal Construction Project Including Earthwork, Proceedings of CONVR 2006.

Kano N., Yao T., Hosoda M. (2002). Virtual Construction Site: Development of a Prototype System and the Issues [in Japanese], Proceedings of 18th Symposium on Construction and Production, 67-72.

Kano N., Hosoda M., Kagawa Y., Tamura N., Miyamoto T. (2003). Virtual Construction Site: Visualized Simulation for Construction Sites [in Japanese], Proceedings of 19th Symposium on Construction and Production, 95-102.

Kano N. (2009). Construction Simulation Trend [in Japanese], Official Publication of Japan Construction Method and Machinery Research Institute, No. 708, 18-24.

Kawanai Y. (2005). The Present Position of Product Model [in Japanese], 7th Workshop of Japan Construction Information Center.

Nagao M., Fukushima K., Sone H., Fujii Y., Matsuba Y., Tsunakawa T., Toma K., Kayashima S. (2005). Cost Estimate and Possibility of Cost Control due to Production Design -- Using 3D-CAD in Production Design [in Japanese], Journal of Architecture and Building Science, Vol. 21, 373-377.

Sriprasert E., Dawood N. (2003). Multi-Constraint Information Management and Visualisation for Collaborative Planning and Control in Construction, ITCon, Vol. 8.

Vries B., Harink M.J.J. (2007). Generation of a Construction Planning from a 3D CAD Model, Automation in Construction, No. 16, 13-18.

Yabuki N., Kotani J., Shitani T. (2004). A Product Model-Based Design System for Steel Girder Bridges Using Virtual Reality [in Japanese], Journal of Applied Computing in Civil Engineering, Vol. 13, 211-220 (2004).

Yabuki N., Makanae K. (2005a). Product Model and 3D/4D CAD [in Japanese], Proceedings of JSCE, Vol. 90, No. 5, 23-25.

Yabuki N., Shitani T. (2005b). Development and Application of a Three Dimensional Product Model for Prestressed Concrete Bridges [in Japanese], Proceedings of JSCE, No. 784, 171-187.