

DEVELOPMENT OF A VISUAL TOOL FOR DYNAMIC SIMULATION PROGRAM HVACSIM+

Yosuke NISHI^{*1}, Xiangyang CHEN^{*2}, Nobuo NAKAHARA^{*3}

*1 Yamatake Corporation Fujisawa 251-8522, Japan *2 Yamatake Building Systems Co., Ltd. Tokyo 108-6030, Japan *3 Kanagawa University Yokohama 221-0802, Japan

ABSTRACT

This paper explains the development of a dedicated configuration tool for the HVACSIM+ simulation environment. This computer-based tool eases configuration by exploiting a visual metaphor of a simulation subject. This paper also explains the configuration tool's method of development, its functionality, and suggests some potential applications which could benefit from an object-oriented development.

The tool was developed with object-oriented technology in 1994. We used the Object Modeling Technique (OMT) to analyze the information of a HVACSIM+ simulation target, and developed the visual configuration tool on top of our existing object-oriented framework.

This new visual tool functions as a preprocessor, in the same way as HVACGEN did for HVACSIM+. And the new tool employs a visual metaphor, making the work of constructing the simulation model easier than was the case with HVACGEN.

INTRODUCTION

As computer technology shows rapid advancement today, all sorts of operation support environment have become available on personal computers (PC). Due to the development of user interface technologies represented in mouse or windowing environment, general-purpose applications such as word processor or spreadsheet now provide a consistent operating environment, as well as excellent operability. Furthermore, the appearance of OLE (Object Linking and Embedding) and other techniques has made data sharing and coordination of functions between different application programs possible. Faster processors speeds and large and inexpensive memory chips for personal computers have enabled application programs that once required costly computing

machines, to run on desktop PCs.

The HVACSIM+ general-purpose simulator is a technical software application that was developed using the FORTRAN programming language. Nearly ten years since it was developed, researchers continue to use this program, now on their desktop PCs. However, because HVACSIM+ was originally developed for use on various mainframe computers, no improvement was made to its human interface (character-based) or interoperability with application programs.

To achieve an ideal simulation environment, it was considered necessary to implement a more interactive human interface and also to enable integration with general-purpose application programs. We started by writing a program for a visual modeling tool that would allow user-friendly operation via a visual metaphor. Programming was also achieved using an object-oriented software technique as it was considered that this would reduce the development time needed, and would also provide us with a software architecture having excellent scalability and good maintainability. After completing this work, our tool was found to enhance the convenience of traditional users of HVACSIM+, and we also found that it was possible to further develop the software in order to realize a design and operating environment for integrated building systems. This report describes the HVACSIM+ simulator and the visual modeling tool, and the way that the object-oriented software method was used in our development project.

Among the software resources available today, there are a number of technical calculating programs that have a high potential for wider application, but which suffer from low usability and scalability. The application of object-oriented software techniques as described in this paper may become effective in leveraging existing software resources and

integrating them with modern computer environments.

VISUAL MODELING TOOL FOR HVACSIM+

1. Necessity of a Visual Modeling Tool

HVACSIM+ uses a pre-processor program HVACGEN to prepare a model description work file (named with the extension 'SIM'). The simulation data contained in this file is limited, however, since it is difficult for the user to easily grasp its contents and see the entire scope of the simulation contained in the file. For using HVACGEN to prepare a model description work file, the main shortcoming is a difficulty in distinguishing between components and seeing the links between them. HVACSIM+ identifies components by UNIT numbers. This structure forces a developer, even a person who had made the original file, to have at hand a UNITS/Component Names List. Links are also hard to grasp at first perusal because the HVACSIM+ links components using an index of variables. When the simulation becomes complicated, the task of reading the simulation very much depends on having at hand the system schematic or the information flow diagram prepared through the deskwork when the original model was created. Computer simulation is commonly thought to be easier by creating a new simulation using the old simulation model description work file. But in HVACGEN this is difficult as system units cannot be added or removed flexibly. Experts of HVACSIM+ prefer to edit a model description work file .SIM using a text editor. The problem with this method however, is that a slight

mistake would make the model description work file unusable. For example, if an index of one variable of a unit is changed, it becomes necessary for the developer to check every index of the variable of all units connected to this variable, in addition to checking all related data (boundary condition, reported variable, etc.). Therefore, it is not common practice to use the old simulation model description work file with HVACGEN. This has been the biggest bottleneck for users using HVACSIM+.

Other shortcomings exist for users using the preprocessor program HVACGEN. The program is designed for use on character-based displays, which means that if HVACSIM+ is run on this tool, all inputs to and outputs from the computer, from generation of the simulation to the results of the simulation, would be in numeric values and characters. It would be quite difficult to describe the simulated system to those not aware of the details of the simulation. The person who made the simulation also must take especial care to ensure that the work file corresponds at all times to the system schematic and the information flow diagram. Analyzing and designing a system through simulation, the repetitious tasks of structuring the system model, calculating, and modifying the model are required (Fig.1). As HVACGEN lacks user-operability, major modifications made during such tasks cause much trouble to users.

To assure correct configuration of a simulation, we considered it essential to develop a visual modeling tool which could graphically show the system schematic on a personal computer screen instead of depending upon paper as was the case with HVACGEN, and automatically create links of flow, pressure, temperature, and humidity between components via connecting pipes and ducts existing in the HVAC system.

2. Background of Development

In developing a visual modeling tool for HVACSIM+, we set goals as shown in Fig.2. To attain these goals, the object-oriented software technique was employed. Concretely, the system was analyzed and designed by the OMT (Object Modeling Technique), and actually mounted using the object-oriented

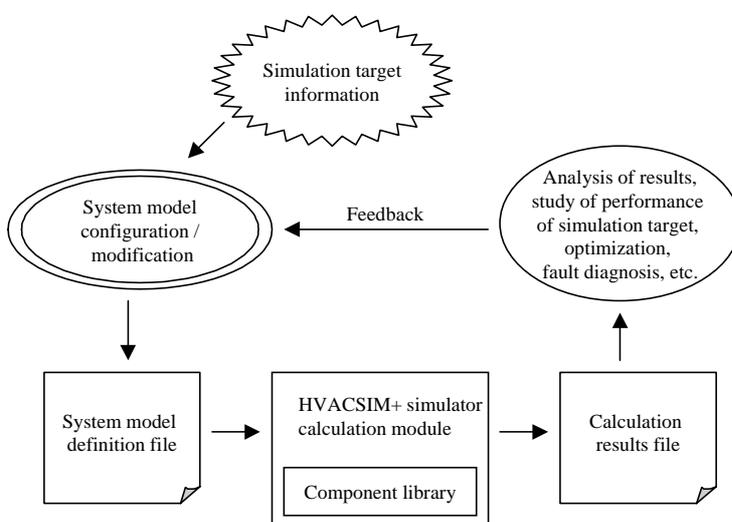


Fig.1 Simulation task flow

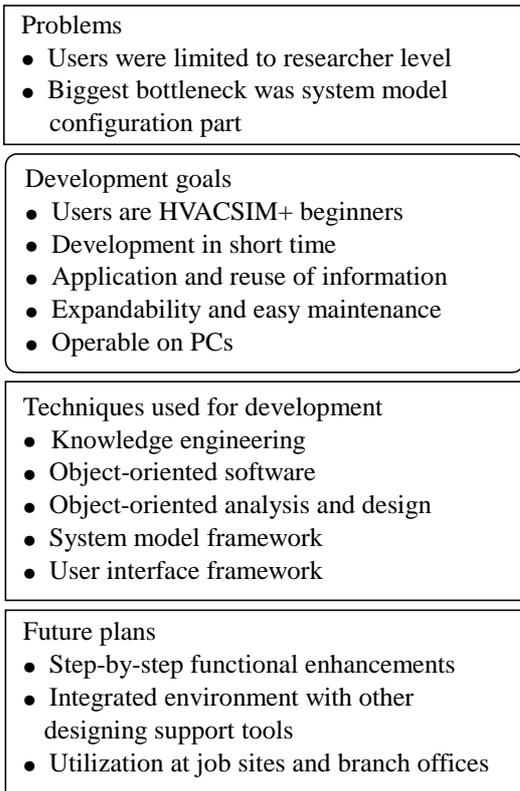


Fig.2 Background of development of HVACSIM+ visual modeling tool

program language Smalltalk and the object-oriented frameworks for supporting the design of control systems (system model framework and user interface framework) developed at Yamatake's Advanced Technology Center.

Including study of HVACSIM+ and problem

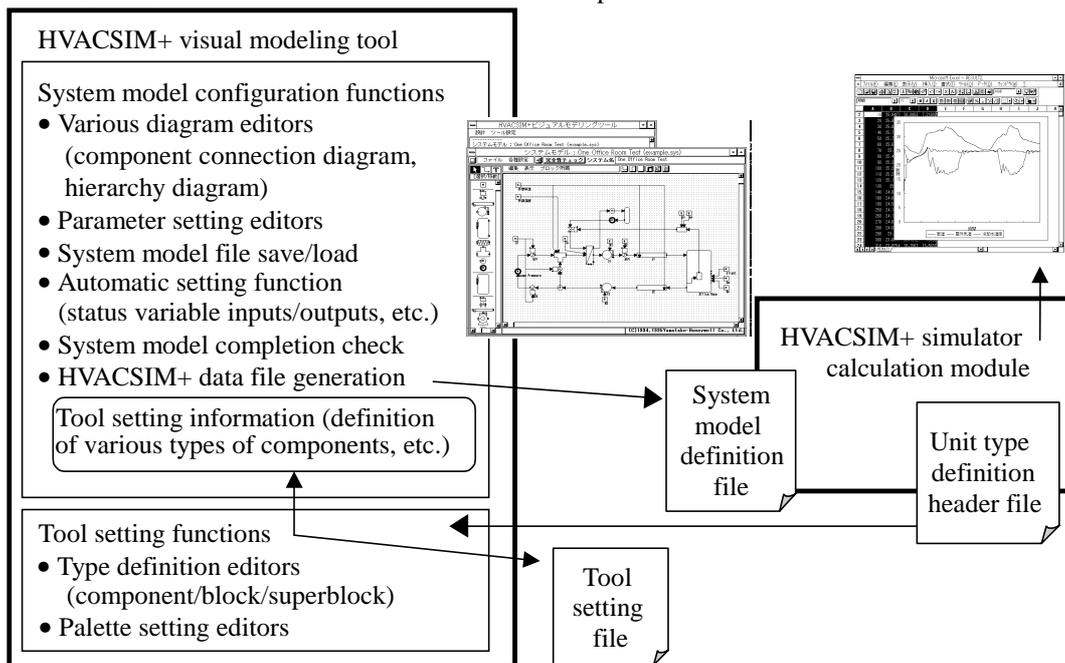


Fig.3 Outline of functions

analysis, we were able to complete the development of the tool for practical use in approximately six months. The details will be explained later in this paper.

3. Function of the Visual Modeling Tool

The visual modeling tool was developed without loading the function of compile utility program SLIMCON, so that it can generate a model description work file similar to that generated by HVACGEN, so allowing sharing of data with HVACGEN users. As well as having all the functions required to generate a model description work file, the tool also possesses functions to expand the tool itself. Fig.3 is an outline of the tool's functions, which are described below.

(1) Component diagram function

- a. Expresses a component as an icon.
- b. Insertion of a component into the simulation system by selecting of a component from the component palette and placement on the editing sheet; deletion of a component from the simulation system by removal of a component from the editing sheet; addition of an identical component to the simulation system by duplication of a component already on the editing sheet, and insertion into the sheet.
- c. Application of component tag name and modifying it on a pop-up window together with its parameters.

d. Drawing of lines to express refrigerants (air, water, etc.) and flow of control data.

(2) Automatic link function

a. Automatic indexing of variables from the flow of refrigerant in four categories: pressure, flow, temperature, humidity.

b. Automatic creation of the control variable index as a result of graphical connection of control signals.

(3) Hierarchical expression function

a. Description of a component connection diagram for each block.

b. Description of a hierarchy of superblocks and blocks.

(4) Various setting functions

a. Setting of initial values for variables

b. Setting of boundary variables

c. Setting of report variables

d. Setting of superblock property (tag name of superblock, assignment of report variable, and selection of convergence calculation options)

(5) Component link verification function

This function is absent in HVACGEN. The visual modeling tool expresses as lines all links between all components, which prevents users to neglect assigning boundary variables and connecting variables whose single output becomes an input to multiple variables.

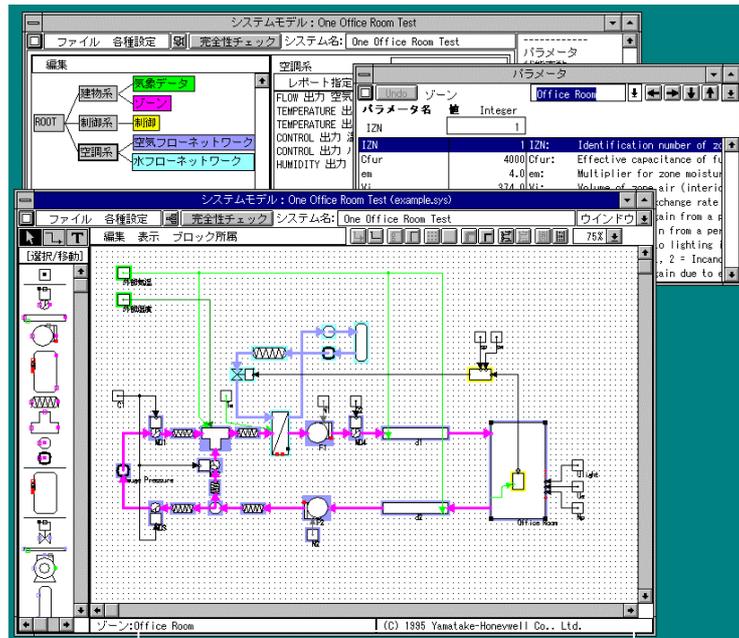
(6) Tool setting function

This tool is provided with functions of registering and editing components that are the basic units of a simulation model, and also has functions for setting and editing the component palette.

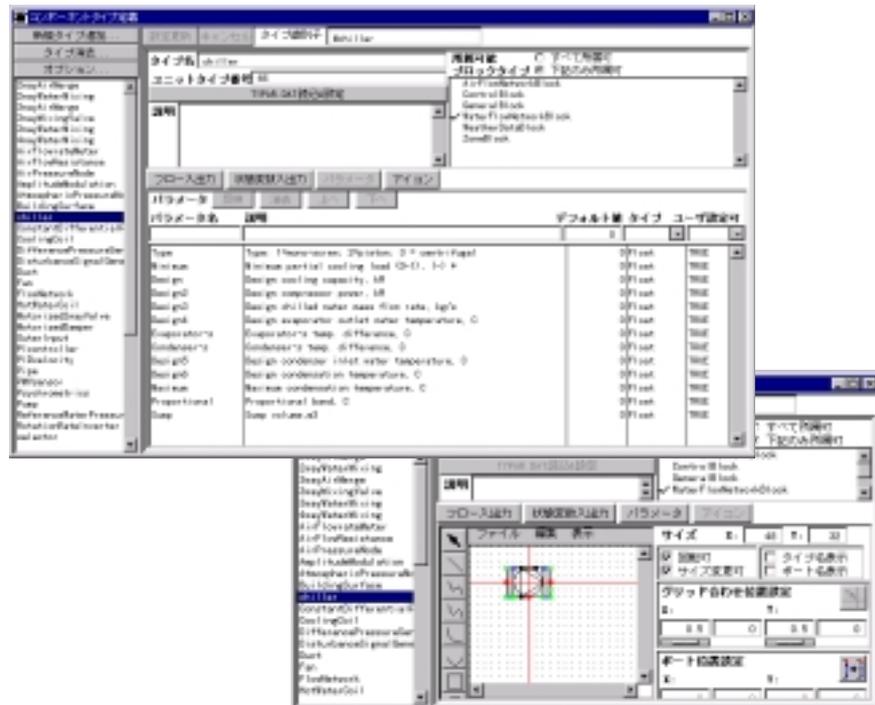
USING OBJECT-ORIENTED TECHNIQUE

1. Framework of object-oriented software for control system designing support

Fig.5 shows the software architecture of this tool. The system model framework and the user interface framework are based on experiences from developing various CASE (system designing support) tools for process control systems and building HVAC control systems.



Example of System Model Configuration



Example of Component Definition screen

Fig.4 Screen examples

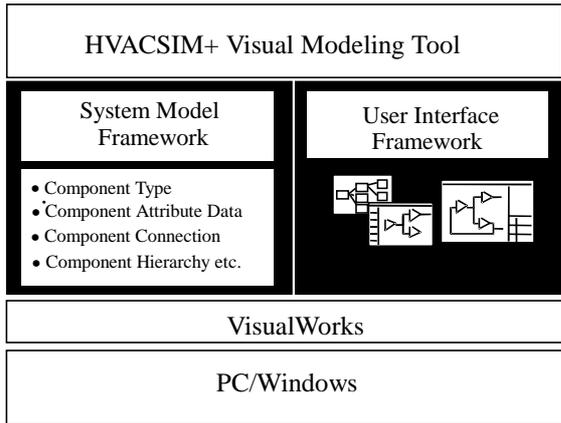


Fig.5 Tool architecture

The system model framework is the basis of a data structure for expressing design information where the system is comprised of a group of components. It is an object class group for describing design information that are thought to be common to various system design support CASE tools. Using this framework, it is possible to express connections between components and the hierarchy of components, as well as describe components from a variety of viewpoints. It is also possible to define various component types, and to set conditions regarding storable data or connections for each type.

In developing a software tool, the user interface is also very important. The user interface framework in Fig.5 is an object class group that provides screens which allow end-users to edit the

design information object group expressed by the system model framework. It has diagrams for displaying and editing connections and hierarchies, fields and tables for data display and input, window management function, etc.

With these frameworks as the basis for development of the tool, other functions peculiar to the tool were added, such as data file generation for simulator input, system model completion check, input/output automatic setting, etc.

2. Correspondence of HVACSIM+ visual modeling tool and frameworks

The results obtained from the object-oriented analysis regarding the system model information of HVACSIM+ simulator, are shown in Fig.6. HVACSIM+ is provided with unit types, such as a damper or a fan, having conditions for input/output of status variables, or parameters. When building a system model, the required units are generated as components and links are made between the status variables of the components. The resulting group of units is stratified into blocks and superblocks. This structure can easily be represented by the system model framework mentioned earlier.

The basic procedure for tool development can be described, as follows.

- (1) Analysis and mapping of design information (system and the component group constituting

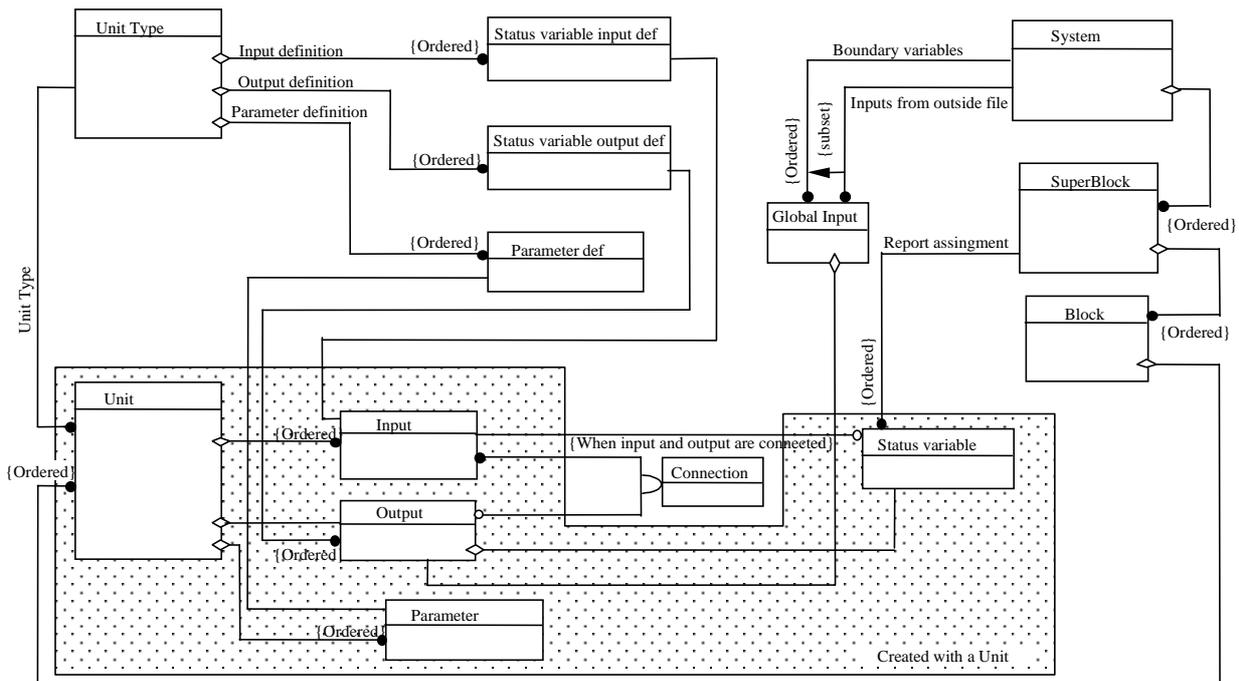


Fig.6 Example of analysis of system model information in HVACSIM+

that system) which are to be processed by the tool, onto the system model framework.

- (2) Design and application of data model for the design information to be processed by the tool, using the system model framework.
- (3) Design and application of screens required for the user to design a system, using the user interface framework.
- (4) Development of auxiliary programs related to knowledge required to structure design information or automatic design aspects.

(2) and (3) can be regarded as reuse of program code provided by the framework.

3. Benefits of tool development using frameworks

Although various class libraries are being gradually added to the object-oriented programming environment, few can handle object models within a specific problem field. However, the frameworks employed here provide the basis of development for a system design support tool---it is distinctive in that it not only provides a graphical user interface, but it also provides object models that exist within the system design information. In other words, the tool provides greater functionality than a typical CAD or a drawing tool, because it establishes architectures that possess meaningful substantive data residing 'behind' each drawing.

We were able to ascertain the following benefits of the frameworks employed in the development of the HVACSIM+ visual modeling tool.

- Significant man-hour reduction for development

By reusing the existing program of the framework, a great part of the programming work is removed by the tool's unique functions. Actually, two-thirds of the source code was derived directly from tool's framework.

- Visualization of design information

It is not easy to develop a tool that allows users to edit on diagrams, the information such as the connection or the hierarchy between components are difficultly expressed in text form.

- Reuse of design information

Apart from the user interface, information in the simulation target is expressed as data object groups that are defined by the system model framework. It is easy, therefore, to process or convert these data, which enables importing the information into other tools.

CONCLUSION

1. Development of HVACSIM+ integrated environment

The HVAC simulator HVACSIM+ had previously been used only by researchers, but by using a visual modeling tool, we were able to provide an operating environment of a complexity suitable for practical use. Users can now clearly and quickly grasp the structure and characteristics of the simulation target, and can analyze its actions thoroughly and effectively, by reading graphically expressed information and by using the input/ output automatic setting function provided by the tool.

We have confirmed that with further development this tool can be used to create an environment integrated with various support tools for optimal operational planning or system designing, we can arrange the basis for providing a more reliable BMS (building management system).

2. Development of framework

Through the development of the HVACSIM+ visual modeling tool, we were able to ascertain the efficiency of the system model framework and the user interface framework. These frameworks can be applied in the future for the development of other, similar, applications. Also, as the system model framework provides data models for representing design information, it may promote the integral use of data between various applications.

REFERENCES

- Clark, D., et al.: HVACSIM+ Users Guide, 1985
- Rumbaugh, D., et al.: Object-Oriented Modeling and Design (OMT), 1991