INDUSTRY FOUNDATION CLASSES AND INTEROPERABLE COMMERCIAL SOFTWARE IN SUPPORT OF DESIGN OF ENERGY-EFFICIENT BUILDINGS

Vladimir Bazjanac
Lawrence Berkeley National Laboratory
Berkeley, CA 94720 – U.S.A.

Drury Crawley
U.S. Department of Energy
Washington, DC 20585 – U.S.A.

ABSTRACT

Interoperability among software tools that are used in building design can make the use of building energy simulation tools more frequent and more effective. Industry Foundation Classes (IFC) are an object-oriented data model of buildings that provides an environment for such interoperability. This paper reports on a "live" demonstration of the use of five commercially available IFC-compatible software applications (including sophisticated CAD and energy simulation software) in support of schematic design of a small energy-efficient bank building.

The demonstration showed that it *is* possible to dramatically reduce the time (and cost) through automatic exchange of building geometry. The savings apply both to simulation and the building design processes, which could make the use of energy simulation in building design more frequent and earlier than usual.

INTRODUCTION

It is quite obvious that the earlier in building design energy efficiency is considered, the more likely it is that the building will be energy efficient when built. Early consideration of energy efficiency can result in fundamental early architectural design decisions that minimize the building's thermal loads [Bazjanac 1982]; it can also result in the early understanding of and commitment to an efficient HVAC design.

As the complexity of a building design increases, the proper understanding of its energy performance and the selection of energy-efficient alternatives increasingly depends on the quantitative analysis of building energy performance. Many software tools that support such analysis, including those developed with the support of the U.S. Department of Energy (such as DOE-2), are in use today.

Yet, such quantitative analysis is seldom performed early in the design process. Several reasons are often cited: The analysis costs too much; it takes too long (so that, when finished, it is no longer applicable to the status of the building design that had further developed substantially while the analysis was in progress); it requires informed guessing because crucial information (that is not yet developed in the design) is missing at the beginning of the analysis and thus its results may be questioned.

A major reason for the unacceptable cost and duration of early quantitative analysis is the process of entering of information that is needed to perform the analysis for a specific building. This is a process of transforming information that exists in different form in other documents and/or data bases into information that is needed by software tools that are used in the analysis. This process is typically manual (and must be repeated for each tool used) and can be further complicated due to possible misinterpretation and/or human error. In the case of whole-building annual simulation, the current way of transferring the information may account for as much as 80% or more of the resources needed to perform energy analysis. [Bazjanac & Crawley 1997]

Automated acquisition of the needed information can virtually eliminate this problem. To accomplish that, *all* participants in the building design process need an environment in which all software in use can share or directly exchange the information it needs. This is an environment of *software interoperability* that should be available to the building industry the way it is, at least partially, available to some other (such as the automobile, aircraft, and process plant industries).

SOFTWARE INTEROPERABILITY

Software interoperability in the building industry (which also includes facilities management) has been developed and promoted by the International Alliance for Interoperability (IAI) since 1995 [Bazjanac 1998]. IAI's Industry Foundation Classes (IFC) are the means of achieving interoperability in this industry: A general object-oriented data model of buildings that facilitates the sharing and exchange of information among IFC-compatible software applications [IAI 1999].

So far, the IAI has published three releases of the IFC object model; the latest is IFC 2.0, released in April 1999 [IAI 1999]. Each new model release contains new classes and schemata that enable interoperability among software applications serving additional segments of the industry. The IFC object model cannot provide interoperability by itself – software developers must provide interfaces to the object model so that their software can create a "project model" that defines a subject building and then share or exchange the information contained in it.

Thus, to achieve true interoperability, it is critical to have IFC-compatible software that is also *useful* to the industry. While the IAI had demonstrated the exchange of information among software *prototypes* at some of the past industry trade shows, the first IFC-compatible applications did not appear on the market until late spring 1998.

The IAI demonstrated IFC-compatible *commercial* software in support of two industry processes at the AEC Systems Show in Chicago in June 1998: The design of a small energy-efficient bank, and the redesign of a large floor in a commercial office building. Ten software applications were shown in that demonstration; eight of those, including all five used in the design of the small energy-efficient bank, are now commercially available. The remaining two are expected on the in the immediate future.

THE DESIGN OF A SMALL BANK

This process in the demonstration focuses on software support during schematic building design. The intent of the demonstration is to show that it is possible and cost-effective to:

- Support the complete schematic building design, starting from a building program;
- Use quantitative simulation of the annual energy performance that is based on "quality of information" typical for schematic design;
- Generate a preliminary prescriptive energy compliance check early in the design process.

The subject building is a bank hypothetically located in San Diego, CA. Besides the banking floor, it contains a vault, a lobby with automatic teller machines, bathrooms, a mechanical space and small storage. The total net building area is 355 m^2 (about 3,820 ft^2).

Five software applications are used in support of the design of the bank building:

- Alberti, an architectural "applied CAD" application from AcadGraph in Germany, used early in schematic design;
- Allplan FT, an object-oriented sophisticated CAD application from Nemetschek in Germany;
- Architectural Desktop from Autodesk, a sophisticated architectural CAD application from the U.S.;
- SMOG/RIUSKA, an application pair from Olof Granlund in Finland that simulates the whole-building annual energy performance and uses DOE-2.1E as its simulation engine;
- ComCheck, an application from the Pacific Northwest National Laboratories in the U.S. that checks compliance with prescriptive energy code.

All applications acquire definitions of building geometry developed by the application that preceded them in the design process. These are based on geometry specifications in IFC 1.5 (the most recent IFC release at the time). In addition, ComCheck imports energy performance data generated by RIUSKA. All information exchange is file-based, using the "*.ifc" file format, which is compliant with STEP Part 21 [ISO 1994].

PRELIMINARY SCHEMATIC DESIGN

The architectural design process starts by converting the building program, defined in a spreadsheet, into "bubbles" that each represent a programmed space.

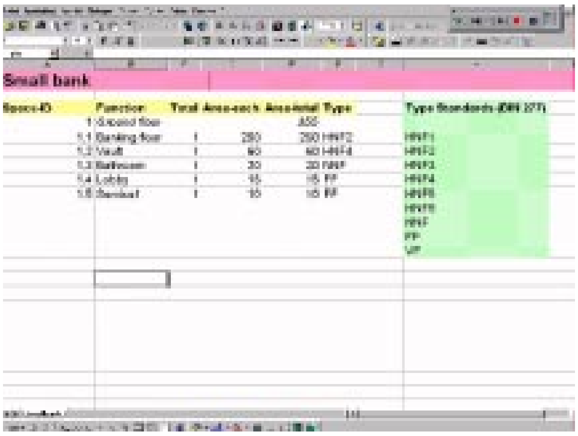

Figure 1 – Building program for a small bank, defined in a spreadsheet

Alberti reads the list of spaces and the content of the spreadsheet, displays them (Figure 1), and then generates the corresponding "bubbles" (Figure 2).

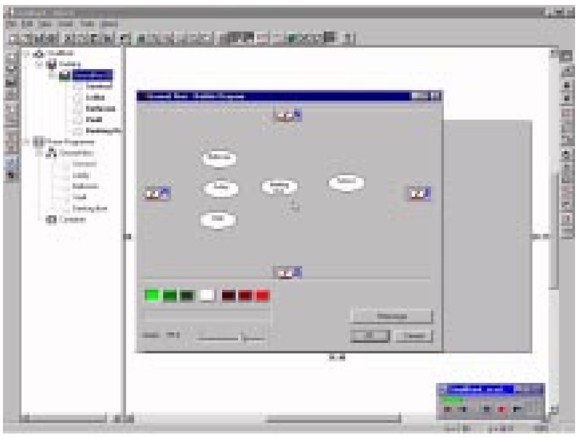

Figure 2 – Building program for as small bank, shown as a "bubble diagram"

The designer defines weighed relationships among the spaces in the diagram by connecting the bubbles with each other. Note that the designer can choose from seven grades of relationship importance, and that orientation can also have a relationship to a space (Figure 3).

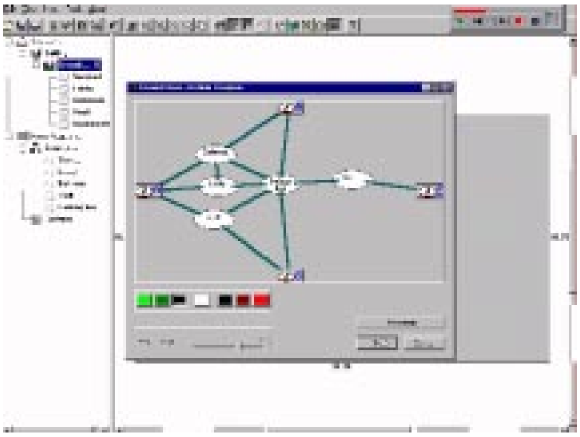

Figure 3 – Weighed relationships among spaces

When instructed, Alberti draws rectangular spaces from the information developed so far: The area of each space is proportionate to the space area as defined in the building program (Figure 4). The program resolves the previously defined (and sometimes conflicting) weighed relationships and positions the rectangles accordingly.

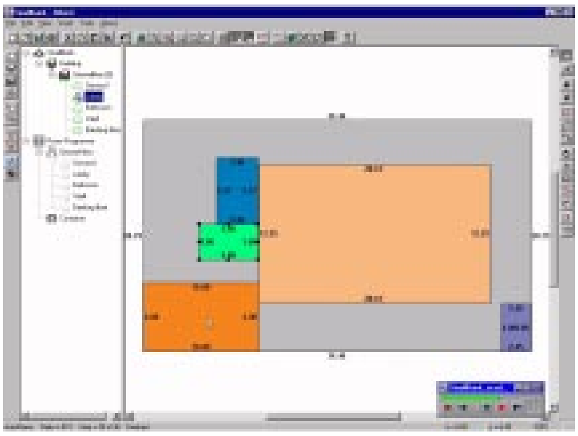

Figure 4 – Rectangular representation of the "bubble diagram" in which proximity to a space represents the weighed relationship

The designer can now engage in rearranging the rectangles, changing their shapes and moving them around until an architecturally desirable arrangement (layout) is found (Figure 5). Regardless of how much the shape of individual rectangles may change, each rectangle preserves its predefined area. The reached arrangement represents a "floor plan" that is equivalent to a very preliminary schematic design solution, as it contains all the properties expected of a design solution at that stage of development.

At this point Alberti saves all generated information in IFC-compatible format in an *.ifc file.

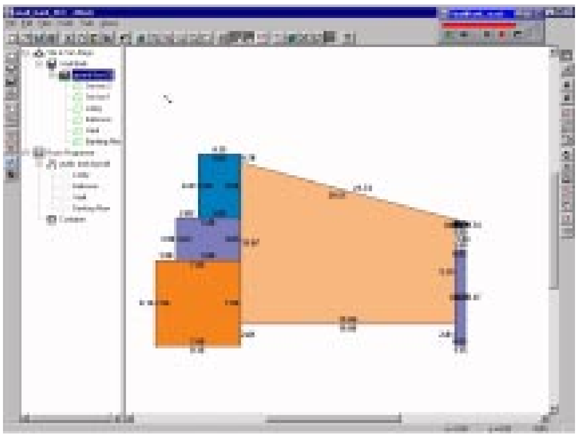

Figure 5 – Schematic floor plan of the small bank

REFINING THE DESIGN

The schematic floor plan is refined and added to with the help of Allplan FT. The designer imports the *.ifc file (generated by Alberti) into this sophisticated CAD application which "knows" that the edges of Alberti's rectangles represent walls and displays them as such in 3-D (Figure 6).

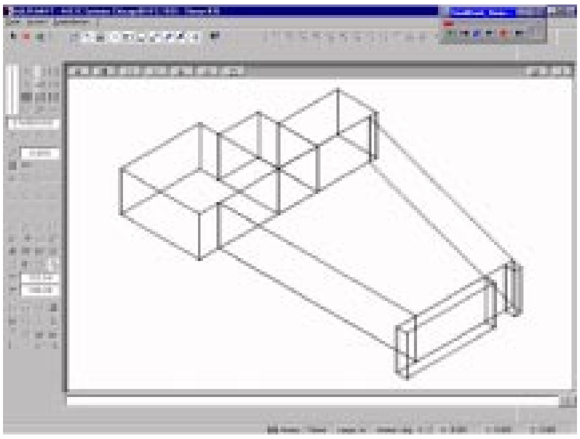

Figure 6 – 3-D view of the preliminary design

The designer now continues architectural design: The better definition of walls and the addition of columns (Figure 7).

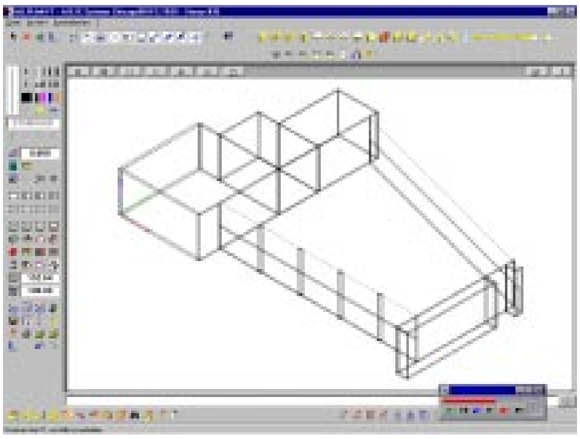

Figure 7 – Definition of walls and columns

Subsequently, the designer selects the appropriate windows from the program database (Figure 8), and makes the preliminary definition of all glazed surfaces.

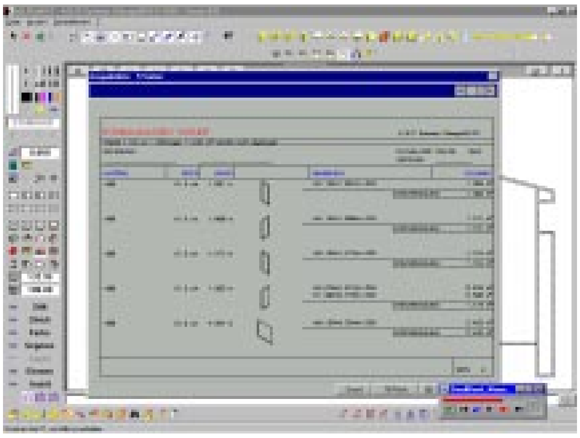

Figure 8 – Selection of windows from Allplan FT's database

Eventually, the designer defines doors and all other openings that are needed in this building (Figure 9).

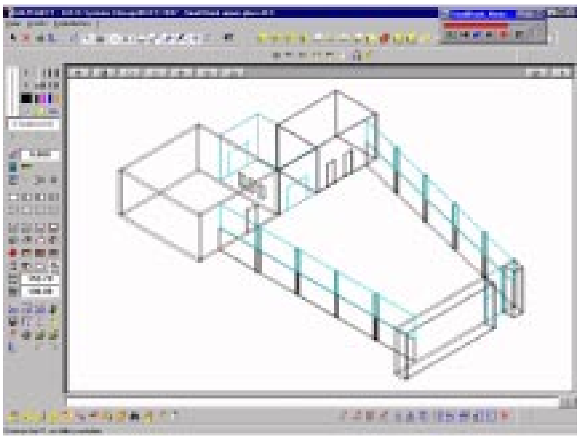

Figure 9 – Architectural refinement: The addition of all openings in walls

In doing this, the designer performs exactly the same functions as in any regular design project when aided by a sophisticated CAD application. The fact that the generated information and the reached design decisions are recorded in an IFC-compatible format is completely transparent. The only difference is that the CAD application must be IFC-compatible and that all files are saved in the *.ifc format. The CAD application itself has exactly the same characteristics and offers the same functionality to the designer regardless of whether it is being used in conjunction with an IFC project model or in another environment.

To show interoperability with another sophisticated CAD application the designer again saves the current work in an *.ifc file and imports it into Architectural Desktop. The designer now continues architectural refinement of the building design by making some adjustments to doors that were defined before and by adding the still undefined roof.

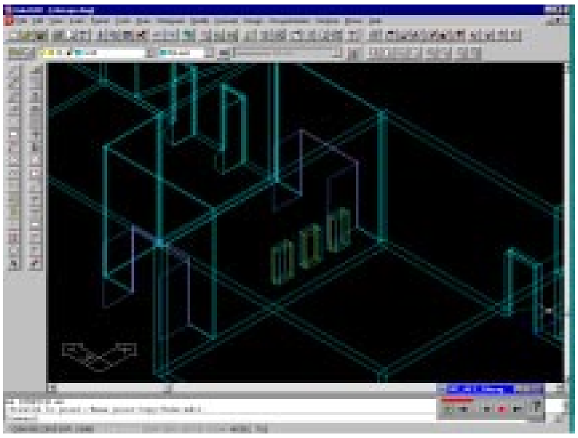

Figure 10 – Change of location and swing of a previously defined door

The designer first repositions the door to the vault and then changes the swing of the door (Figure 10). This is done in the object-oriented application by simply moving the door along the wall and reversing its swing.

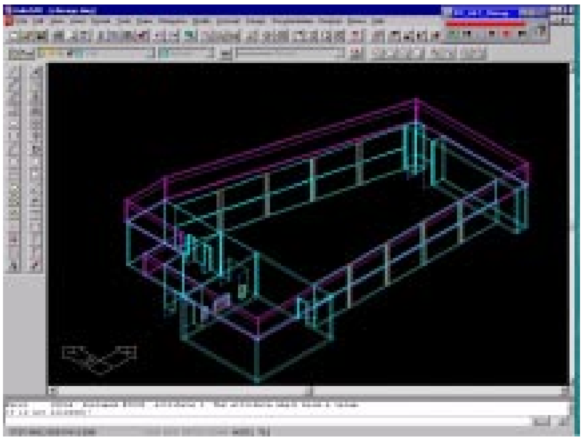

Figure 11 – The completed schematic design includes the building roof

When the roof is completed (Figure 11), the designer has developed sufficient information to start the quantitative analysis of thermal performance of the designed building. Again, the work is saved and passed to the next IFC-compliant application.

EARLY QUANTITATIVE ANALYSIS OF ENERGY PERFORMANCE

SMOG/RIUSKA is an energy-application pair that consists of two separate programs. SMOG is a CAD application that runs "on top" of AutoCAD R14 and creates the definition of building geometry for the simulation. RIUSKA assigns thermal zones, selects HVAC systems, defines internal loads, and runs the simulation's DOE-2.1E engine. RIUSKA's post-processor formats and displays the output from the simulation.

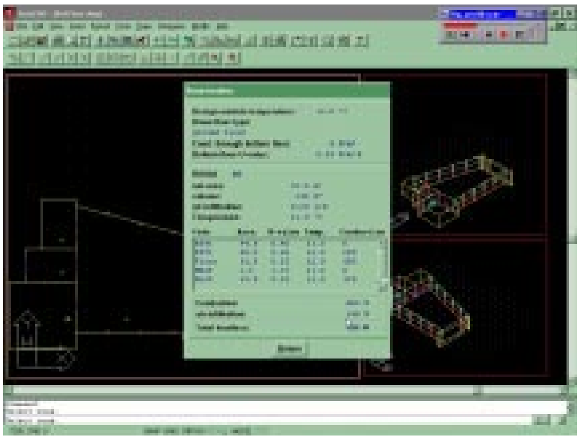

Figure 12 – Space peak heat loss and gain calculated and displayed by SMOG

This sequence starts when SMOG imports the *.ifc file, regenerates the building CAD description, assigns names to spaces, and calculates and displays static peak heat loss and gain (Figure 12).

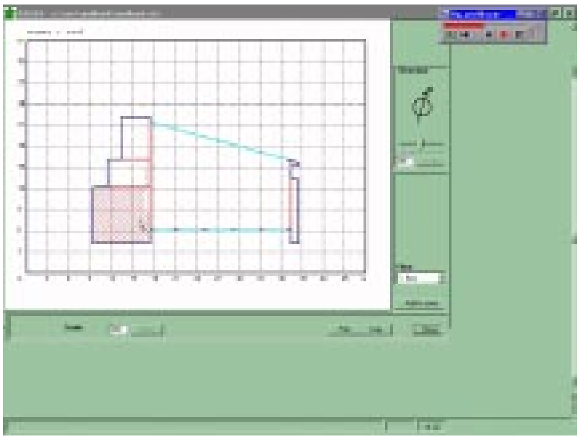

Figure 13 – Assignment of thermal zones

The designer then starts RIUSKA, gives spaces names for the simulation and assigns spaces to thermal zones (Figure 13).

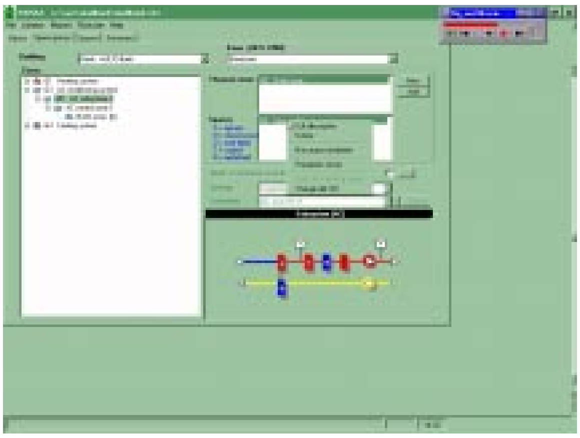

Figure 14 - Selection of HVAC systems

Subsequently, the designer selects heating and cooling systems (Figure 14) and equipment, and defines internal loads for the simulation, all with the help of a convenient graphic user interface.

After choosing the appropriate annual weather file, all required input is set for a whole-building annual simulation of the building. With the simulation run completed, the designer uses RIUSKA's post-processor to select specific output reports and results from a large set of options, and displays them graphically (Figure 15).

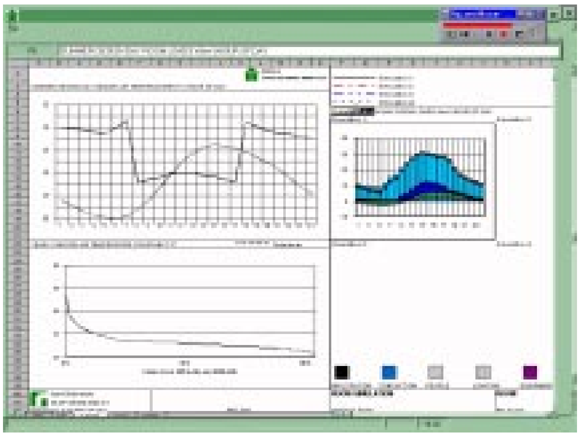

Figure 15 – Selected output from the simulation: Summer peak-day loads and temperatures

The designer again saves the work in an *.ifc file, including the information that was calculated and used in the energy simulation (such as the U-value for each exterior surface). This information is needed by ComCheck, an application that checks the designed building's compliance with prescriptive building energy code.

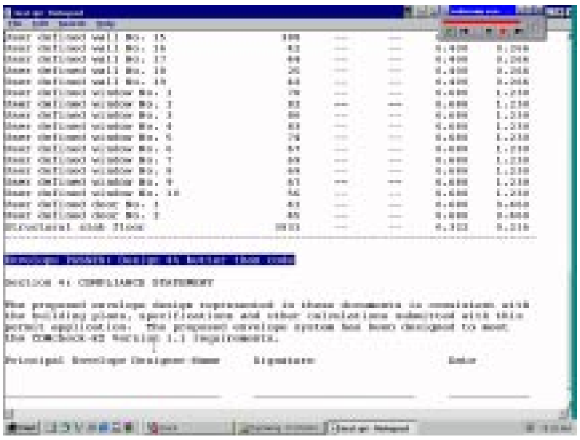

Figure 16 – Energy code compliance report from ComCheck

ComCheck extracts simple building geometry and thermal performance information from the imported *.ifc file. The program displays this information in tabular form and the designer, if needed, has an opportunity to repeatedly modify the building's components and specifications until the code is met. Eventually, ComCheck prepares a formal compliance report that is required by energy code enforcement agencies (Figure 16).

THE EFFECT OF THE DEMONSTRATED SOFTWARE INTEROPERABILITY

In a real-life environment the single "designer" in the demonstration could in fact be several professionals, each with particular skills needed for the tasks performed in schematic design. The fact that these different individuals may not work in the same location does not affect the demonstrated benefits from interoperability, as long as they all use IFC-compatible software and have unimpeded access to the project model.

Interoperable software, when used in support of early design, has a *major* effect on resources needed to accomplish the task. The following are estimates based on educated guesses and the current prevailing professional and consulting rates on the West Coast of the U.S.

The cost of doing the same work in the current, conventional way is high:

- Minimum $7,000 in direct cost (estimate), distributed among architects and consultants;
- A total of one man-week of work to complete all necessary work;
- Three-eight weeks to complete the process (estimate).

In comparison, the cost of doing the work with the support of IFC-compliant software is much lower:

- $800-1,500 in direct cost (estimate), with all work possibly done in one (the architect's) office;
- A total of one man-*day* of work to complete all necessary work;
- One-three *days* to complete the process (estimate).

The wide range of estimates to complete the process is due to the possible necessity to reach agreement among several parties, some of whom may be peripheral to the process (such as the chief designer). These parties may not all be readily available when agreement is needed.

The demonstration also shows that software support of *early* design decisions is possible and can be *cost-effective* with the use of existing, commercially available IFC-compatible applications. It is now specifically possible to:

- Generate useful information about the energy performance of the building design and design alternatives during the schematic design phase;
- Attain *dramatic* reduction in time, effort and cost to accomplish the task;
- *Cost-effectively* create IFC project models of buildings and automatically import the needed building geometry into other IFC-compliant applications, such as structural, HVAC, lighting, facilities management and costing applications;
- *Cost-effectively* create a 3-D model of the building that can later be used in other applications, such as rendering and visualization.

A GLIMPSE INTO THE FUTURE

IFC-compatible software applications that are emerging on the market are the pioneers that constitute only the beginning. Several software developers are now implementing IFC 2.0 and other are preparing for the implementation of release 3.0. The following is only part of the work that will directly impact the world of building simulation and which has already started or is planned to start soon:

- Interactive acquisition of building geometry for EnergyPlus;
- The definition of the EnergyPlus "view" of the building;
- Additions to what is being defined in the current IAI projects to support performance metrics and HVAC control systems;
- The completion of the HVAC extension model in the IFC object model;
- Development of IFC schemata that will allow the definition of air-flow in buildings.

Eventually, the IFC object model will fully support the definition of energy performance of buildings. Building simulation software of the future will be fully interoperable. This is one of the reasons why EnergyPlus and similar simulation software of the early 21st century will be *truly* "new generation."

CONCLUSION

Commercially available, IFC-compatible software can dramatically reduce the cost and time needed for energy analysis *during* early building design. Its use can be equally effective during later stages of design. The use of such applications does not require any special skills beyond those needed to operate the professional CAD and/or simulation software and the expected professional competence.

The availability of interoperable software for the building industry will increase as the development of the IFC object model progresses. This demonstration shows what can already be accomplished in the design of energy-efficient commercial buildings with only five software applications. Future interoperable software will be able to support virtually any process in the industry. This will fundamentally change how decisions regarding building are made, and will irreversibly change the ways in which we all do our work. The building industry is finally on the way to fully benefit from the use of information technology.

REFERENCES

Bazjanac, V., 1982. “This is Where and What,” *Progressive Architecture*, April, 1982: 122-126.

Bazjanac, V., 1998. “Industry Foundation Classes: Bringing Software Interoperability to the Building Industry,” *The Construction Specifier*, June, 1998: 47-54.

Bazjanac, V., & D.B. Crawley. 1997. “The Implementation of Industry Foundation Classes in Simulation Tools for the Building Industry”, Proceedings of *Building Simulation ‘97*, September, 1997, Prague, Check Republic: I-203-210.

International Alliance for Interoperability. 1999. *Introduction to the IAI and Its IFCs*, IAI, April, 1999.

ISO. 1994. *ISO-10303-Part 21*, ISO-10303-1:1994.