

# DECISION MAKING THROUGH USE OF INTEROPERABLE SIMULATION SOFTWARE

Konstantinos Papamichael, John La Porta & Hannah Chauvet  
Advanced Building Systems Group  
Building Technologies Program  
Environmental Energy Technologies Division  
Lawrence Berkeley National Laboratory  
1 Cyclotron Road, MS 90-3111  
Berkeley, CA 94720

## ABSTRACT

Many building simulation computer programs, originally developed on mainframe computers for research purposes, can now run on the powerful workstation and personal computers that are available to most architectural and engineering firms. Major efforts have been underway during the last decade to compile these programs on personal computers and make them available to a wider range of building professionals. However, even with the addition of user-friendly front- and back-ends, their use is still limited to a small number of specialized consultants. Considering the tremendous benefits of informed decisions that these programs can support, it is critical to address and resolve the issues that are associated with their limited acceptance.

In this paper, we report on our research and development efforts to better understand decision-making and develop computer tools that will facilitate the use of simulation software during the building design process. We present a brief analysis of decision-making and then describe how we try to address it in building design through the development of the Building Design Advisor (BDA). Moreover, we elaborate on the major issues that we have encountered, discuss lessons learned, and offer recommendations for short- and long-term developments in this area.

## INTRODUCTION

As the cost of computing power continuously decreases, more and more simulation software is being ported from the mainframe computers where it was originally developed for use by researchers, to the powerful and relatively inexpensive desktop computers that are now available in most building design offices. This phenomenon started with CAD software and is now expanded to analytical simulation tools. In most cases, these porting efforts include development of "front-" and "back-ends," which refer to graphical, user-friendly interfaces for the preparation of the required input and the review of the resulting output.

However, even with the use of friendly, graphical interfaces, such programs have seen limited acceptance by the wider building design communities. This is because the programs are still hard to use and they do not really fit within the building design process. The preparation of the input is time-consuming and it often requires significant understanding of the underlying modeling principles and implementation details. This is especially true for sophisticated simulation tools, which require very detailed descriptions of the building and its context. Moreover, the use of multiple simulation tools requires the preparation of multiple input descriptions, because different simulation tools assume different building models. To realize widespread use of such computer-based simulations in building design, along with the associated benefits of improved designs, it is important to understand how they fit within the building design process to meet the needs of building decision-makers.

## BACKGROUND

Lawrence Berkeley National Laboratory has a long tradition in the development of simulation tools. Sophisticated computer programs, such as the DOE-2 building energy simulation program [Winkelmann et al 1993], the SUPERLITE daylighting analysis program [Modest 1982], the RADIANCE day/lighting and rendering program [Ward 1990] and the COMIS airflow and indoor air quality program [Feustel 1992], have been under development for many years. Through use of such tools we came up with several useful strategies and technologies to improve building energy efficiency in cost-effective ways, without sacrificing comfort. However the performance of most varies, depending on the context of their application. In order to transfer such strategies and technologies to the building industry, we need to provide means for decision-makers to properly evaluate their performance on a project-by-project basis.

Responding to this technology transfer challenge, we initiated research efforts in 1986 to explore the use of information technologies for the

development of building design tools that can integrate the predictive power of simulation tools in decision making during the building design process [Selkowitz et al 1986]. Through collaborative projects with various academic and research institutions, we explored the use of various software technologies, focusing on artificial intelligence techniques and the multimedia.

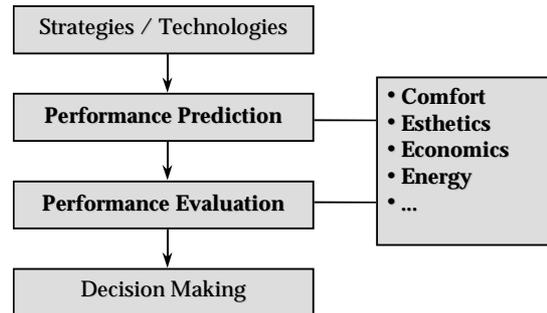
By 1989 we had developed a significant understanding of the possibilities and limitations of such methods and techniques. Most important we had realized the need to better understand the decision making process. Through collaborative efforts with the Department of Architecture at the University of California at Berkeley, we focused on the development of a comprehensive design theory. By 1991 we had developed a significant understanding of the decision-making process in building design [Papamichael 1991; Papamichael and Protzen 1993]. Based on this new knowledge, we came up with several new ideas and a demonstration prototype of a building design tool, which attracted the interest of California utilities for use in Demand Side Management programs. Finally, in 1994, we initiated the development of the Building Design Advisor, a computer program that facilitates decision making through integrated use of multiple simulation tools and databases.

## DECISION MAKING

A major prerequisite in decision making is the ability to predict performance, which, in building design, is only possible through simulation. Up to a few years ago, simulation methods were limited to manual procedures, such as drafting, drawing, building of physical scale models, and performing manual calculations. Research and development efforts during the last two decades have produced a large variety of computer-based simulations that offer significant advantages when compared to manual methods. Almost all of the architectural and engineering firms currently enjoy the benefits of Computer-Aided Drafting (CAD) software, while a significant number of applications on structural, lighting, energy, economic, etc. analyses are now used regularly on large projects that can afford the higher associated costs.

Performance prediction is necessary but not sufficient for decision making. The true essence in decision-making is evaluation, that is the assignment of “goodness” or “appropriateness” to the predicted performance. Since “good” and “bad” can only make sense when there are at least two of a kind, performance evaluation, requires *comparison* of alternative courses of action. Most

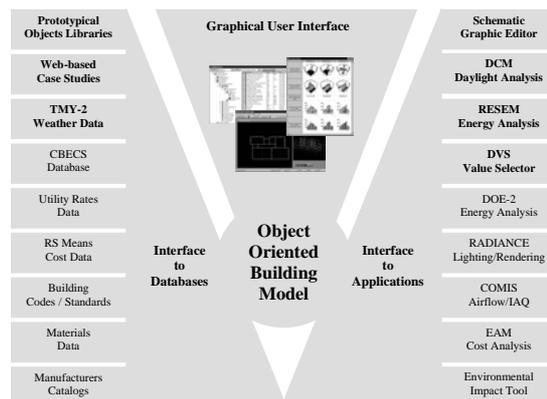
important, it involves all performance aspects in a qualitative type of judgment (Figure 1) [Papamichael & Protzen 1993].



**Figure 1.** Decision-making requires performance prediction as well as performance evaluation with respect to multiple performance considerations.

## THE BUILDING DESIGN ADVISOR

The Building Design Advisor (BDA) is a Windows™ computer program that addresses the needs of building decision-makers from the initial, schematic phases of building design through the detailed specification of building components and systems, following the decision-making theory described above. The BDA is built around an object-oriented representation of the building and its context, which is mapped onto the corresponding representations of multiple tools and databases. It then acts as a data manager and process controller, automatically preparing input to simulation tools and integrating their output in ways that support multi-criterion decision making (Figure 2).



**Figure 2.** Schematic diagram of the BDA architecture. Elements in **bold** font type indicate current development efforts. Elements in plain font indicate plans for future extensions.

One of the original objectives in the development of the BDA was to support links to third party simulation tools without the need to recompile the BDA. We wanted to be able to expand the model

dynamically, by adding objects, attributes and relations as needed by new simulation tools. Moreover, we wanted to lay a foundation that would allow us to link the BDA to tools that address stages of a building's life beyond design, such as construction, commissioning and monitoring tools. Finally, we were challenged to support the use of sophisticated simulation tools from the early, schematic phases of building design, when decisions on building details have not yet been made. To satisfy these objectives, we developed the BDA using three databases, the Schema Database, The Prototypes Database and the Project Database.

### THE SCHEMA DATABASE

The Schema Database describes the content, or structure, of items that BDA can store. These items are Building Objects (e.g. space, wall), Relations used to link objects together (e.g., has, faces), Properties that describe the objects (e.g., height, U-value), Units used to measure properties (e.g., ft., cm., °C, °F), and Simulation Tools that compute values of performance properties (e.g. DOE-2, RADIANCE, etc.). Each object parameter is linked to the simulation tools that use it as input or output along with the associated type of units.

To facilitate the development of the Schema Database we developed a Graphical User Interface that allows developers to define new Simulation Tools, Building Objects, Properties, Units, and Relations. The Schema Database also has reporting utilities that allow developers to check the consistency and semantics of the Schema (e.g. parameter definitions and relations to simulation tools).

### THE PROTOTYPES DATABASE

The Prototypes Database is used to store Libraries of predefined building objects, or Prototypes, e.g., glazings, wall constructions, etc. Each Prototype is created with its own list of parameters as defined in the Schema Database, and each parameter is assigned a Value from some Source or Data Reference. The Prototypes Database is the main source of building components and systems available to the user for the description of the building.

Like the Schema Database, the Prototypes Database has its own Graphical User Interface that allows developers to enter new Prototypes and modify existing ones. Moreover, it too has reporting utilities that allow listing of all Instances for each Object Type, so that all of the

Prototypical Database contents be listed and printed.

### THE PROJECT DATABASE

The Project Database is used to store the Building Objects that are created at run-time for a specific project that the BDA user is working on. Staying with our "generic" approach, we did not define classes for different building objects. Rather, we defined classes for Run-time Building Object, Run-time Parameters, and Run-time Values, along with five derived classes to handle integer, real, string, real array, image, and multi-media data types.

In the BDA run-time system, not only Building Object Type Instances are C++ objects, but Parameters and Values as well, so that Parameters can have more than one Value, each from a separate Source or Simulation Tool. The reason for this "expensive" representation is the desire to use the BDA environment for the implementation of a Building Lifecycle Information Support System (BLISS), which will expand beyond design to address the data needs of building construction, commissioning, operation, etc.

To satisfy the need for performance evaluation, the BDA supports the maintenance of multiple design alternatives within a "project" database. A new alternative design solution is generated at any point as a copy of any of the existing solutions. The BDA user interface supports the concurrent review and manipulation of any number of alternative design solutions. Moreover, it supports their side-by-side comparison with respect to multiple performance considerations.

### THE GRAPHICAL USER INTERFACE

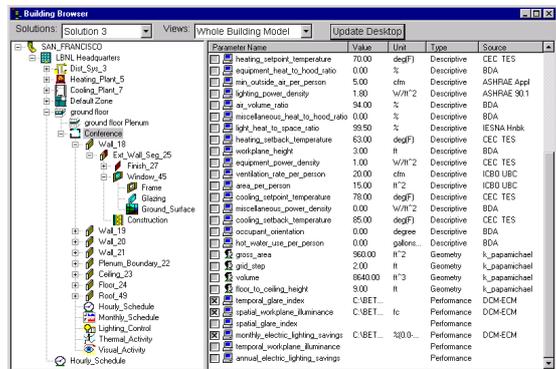
The BDA has a Graphical User Interface (GUI), composed of two main elements: the Building Browser and the Decision Desktop. Both GUI elements are views into the Object-based building model. Both were designed around user interface schemes that are familiar to Windows™ users.

From the user's point of view, the BDA represents buildings using a hierarchical structure of objects (e.g., site, building, zone, space, wall, window, etc.) that are characterized by lists of parameters.<sup>1</sup> In the same way that the Windows™ Explorer supports browsing through directories and their

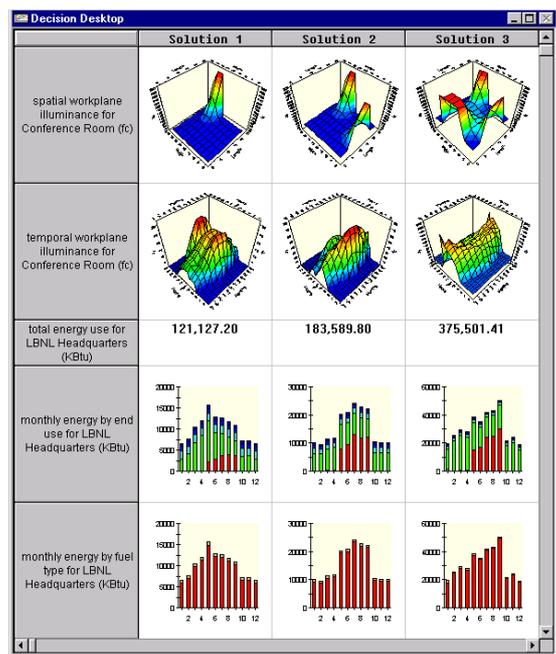
---

<sup>1</sup> In fact the relations among building components and systems maintained by the BDA model result in a network schema.

files, the Building Browser supports browsing through building objects and their parameters (Figure 3). Moreover, it supports editing of parameter values and it allows the user to select which parameters they would like to compute and/or view in more detail in the Decision Desktop



for decision-making. To facilitate browsing through large numbers of objects and parameters, the Browser also supports the definition of customized views, sorting by any parameter attribute, etc.



**Figure 3.** The Building Browser allows the user to navigate through the hierarchy of objects, edit their specifications, and select the parameters to be displayed in the Decision Desktop.

**Figure 4.** The Decision Desktop allows the user to compare multiple alternative design solutions with respect to multiple design considerations.

The Decision Desktop is a spreadsheet-like GUI element, whose rows correspond to building objects and parameters, and columns correspond

to alternative design solutions (Figure 4). Every parameter selected by the user in the Building Browser is automatically entered as a row into the Decision Desktop. In this way, BDA supports the side-by-side comparison of multiple solutions with respect to multiple performance considerations. The Desktop's cells can hold any data type, from numbers and text to graphics, audio and video. Users can re-order parameters and solutions at any point, sort solutions by any of the displayed parameters, etc.

## THE CASE STUDIES DATABASE

To further support the performance evaluation process, the BDA is linked to a World-Wide-Web-based Case Studies Database (CSD), so that decision-makers can compare the simulated performance of proposed designs to that of existing buildings. The case studies database also serves as a source for information on real-world applications of strategies and technologies. The World-Wide-Web offers unique potential for fast, worldwide growth, and, through its multi-media capabilities, an excellent opportunity for comprehensive coverage of descriptive as well as performance characteristics.

We have implemented a schema through which any one can "submit" a case study to the CSD by completing a form with basic information about the building and providing a link to the particular Web Site of the case study. We then maintain a master list of all entries with the basic information, which can be searched either through "text" searches or through database queries on specific descriptive and performance fields.

## ADDRESSING SCHEMATIC DESIGN

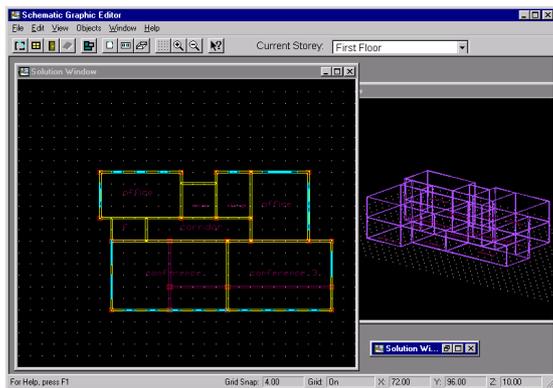
One of the main challenges that we had to address in the BDA development efforts was the need to use sophisticated simulations from the early, schematic phases of building design, without "distracting" decision makers for the required detailed information about the building and its context. Moreover, we needed a graphic editor that would allow building designers to quickly and efficiently specify and visualize the main geometric characteristics of building components and systems. These requirements resulted in the development of a Schematic Graphic Editor (SGE) and a Default Value Selector (DVS).

## THE SCHEMATIC GRAPHIC EDITOR

The Schematic Graphic Editor (SGE) was developed on top of a third party library of CAD functions. It was developed as a stand-alone

application because we consider it as the equivalent of a simulation tool, and also because we expect that it will be replaced with more sophisticated CAD applications in the future.

The SGE allows the user to draw building “objects” (e.g., spaces, windows, doors, etc.) as opposed to “lines” that represent building objects in one’s mind, like in most conventional CAD packages (Figure 5). The SGE continuously communicates with the main BDA application, passing on information on the object types and geometry specified by the user.



**Figure 5.** The Schematic Graphic Editor allows the user to draw building objects as opposed to lines.

### THE DEFAULT VALUE SELECTOR

The Default Value Selector (DVS) is a mechanism for the automatic assignment of default values for all parameters that have not been addressed by the user and are required as input by the simulation tools linked to the BDA. The DVS is an integral part of the BDA and works in synch with the Schematic Graphic Editor (SGE).

As the SGE sends information to the BDA about each object that the user draws on the screen, the DVS selects default values from the Prototypes Database, based on building location and type, and in some cases space type and boundary type. The rules for the selection of the default values follows building codes, standards, and recommended practice, taken from a number of sources, such as the ASHRAE Handbook of Fundamentals [ASHRAE 1993], the Handbook of the Illuminating Engineering Society [IESNA 1993], etc. The user can change the default values at any point through the BDA user interface.

### THE DATA ASSIGNMENT SCENARIO

During the creation of a new space in SGE, the user is asked to select a Space Prototype from the

ones available in the Prototypes Database, such as “Lobby,” “Conference Room,” etc. When the user creates a Space object, the Schema Database is queried for a list of Parameters that describe a space object. Once these parameters are created, the Default Value Selector is used to assign values based on building type and location, and occasionally on space type and boundary type.

### MAJOR ISSUES

There are several major issues that we have been addressing during the development of the BDA. These can be classified into two categories: technical and administrative. So far, the most important technical issues have been the differentiation of the SGE and the BDA models, and the modeling of conceptual objects. The most important administrative issues have been the size and composition of the software development team, time requirements for exploration of alternative implementation methods, and the required involvement of a diverse set of professionals. Herein we focus mostly on the technical issues and briefly touch up on the administrative ones.

### THE CAD AND PHYSICAL MODELS

The greatest difficulty in using a CAD system to describe a building for simulation tools is that the CAD system and the simulation tools model the world in totally different ways. The primary goal of a CAD system is to allow the user to specify and manipulate geometry. This is accomplished by representing the world with various graphic objects (entities or symbols), such as lines and polygons. These entities can be easily created and manipulated by the user because as an object, a CAD polygon knows how to display itself, show grip handles at its vertices, and respond to mouse clicks and drag events.

The second model is that of the physical world. In the A/E/C industry, this is a model of building objects such as spaces, walls, windows, etc. The Physical model is rich with non-geometric attributes, but does not have a notion of displaying itself on a computer screen. This is the model required by simulation engines that reason about various domain parameters in energy, comfort, structures, etc.

A simple example of the disparity between the two models can be seen in a wall object. In the Physical model, a wall object contains a long list of non-geometric objects such as surface reflectance, materials, U-Values, and a set of relationships to other objects such as spaces, doors

and windows. The wall would contain only that geometry necessary to describe itself in the real world - probably a list of vertices. By contrast, in the CAD model, the Wall object consists of a polygon defining the wall, layer information, lines styles, color, pen thickness, and rich set of methods which allow it to display itself and be modified through mouse-based interactions with the user.

### THE DUAL MODEL APPROACH

In BDA, the completely disparate needs of the two models resulted in two separate applications to create and maintain them. The CAD model is maintained by the Schematic Graphic Editor and the Physical model is maintained by the BDA in a rich, object-oriented database system.

Because SGE is a separate application built on top of a third party library of CAD functions that does not have an interface to our database management system, all the objects drawn in SGE are saved to a file that is independent of the BDA Project database. As a result, we have to make sure that the two separate representations are synchronized during the “save” and “load” operations. This synchronization problem keeps us from using the Project database to its full advantage. If both SGE and BDA operated on the same Building Objects, then changes could be saved as they occur and only those parts of the Project database that needed to be displayed would be loaded, truly utilizing all of the advantages of a Data Base Management System.

The only advantage of our current approach is that the Physical model in BDA can be kept free of the large amounts of CAD information that is extraneous to the needs of the simulation tools.

### THE SINGLE, INTEGRATED MODEL

The most viable long-term solution that we see is the merging of the two models into one, so that a single environment exists with both the CAD functionality required by the user, and the database functionality required by the simulation tools. In this approach, the wall object will know everything about being a wall in the physical world, as well as how to display itself on the screen and respond to mouse clicks and drags. Unfortunately, such an environment does not yet exist. However, the industry is moving closer to it with the efforts of the International Alliance for Interoperability (IAI). The IAI is developing a data model that will encompass both the graphic needs of CAD systems, and the data needs of analysis tools. The other goal of the IAI is that of

standardization. If a standardized model existed, then the conversions between different CAD and analysis programs would be eliminated. No conversion would be required since all programs would simply create IAI Wall objects, IAI Window objects, etc. However, a new generation of simulation tools would have to be written to take full advantage of this approach.

### REAL VS CONCEPTUAL OBJECTS

Another major challenge in our development efforts has been the modeling of conceptual objects, such as plenums, schedules, activities, etc., which do not really exist as real, physical objects. The most common and most problematic conceptual object is the Space one. The space has been a focal point not only in the required functionality of the Schematic Graphic Editor (e.g., users want to be able to “move spaces around...”) but in the modeling of the simulation tools as well. Most daylight calculations are performed on a space-by-space basis, as are many of the thermal and air quality calculations. This is intrinsically problematic in modeling because the object that we consider as most important does not exist in the physical world.

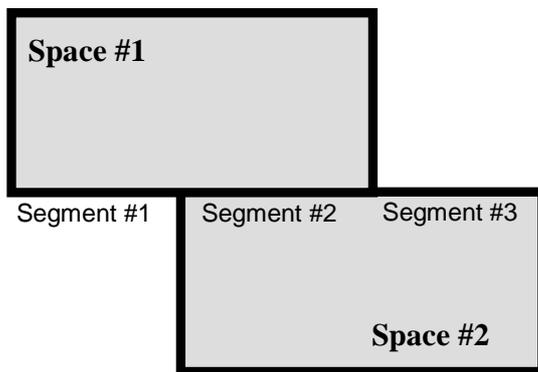
The space is an abstraction that permits us to reason about a given volume that is defined by a combination of physical and imaginary boundaries. In the simplest case, a space is defined on all sides by physical boundaries (e.g., walls, floor and ceiling). However, spaces can also be defined by a small change in elevation, or a change in the floor material, or by completely imaginary boundaries that we use to mentally “close” a room, but which do not exist in the physical world.

### WALLS AND WALL SEGMENTS

The approach that we have taken in BDA is to allow the user to define each space by drawing a polygon in SGE, explicitly closing it. Then, after the space has been defined, the user may edit specific space boundaries and designate their construction to NULL. This provides for an exact definition of the space, while allowing for non-physical boundaries. One problem that arises from this approach is that walls shared by two spaces are defined twice, since each space is explicitly described. To solve this problem we introduced the notion of the Wall Segment object.

While Wall objects are still used to define the perimeter of each space, each Wall is composed of one or more Wall-Segments. When the SGE detects overlapping Walls, these are automatically

segmented into the proper number of Wall Segments, so that there is no overlap (Figure 6). The Wall Segment then is used to define the construction and other physical attributes required by the simulation tools. Through this approach we model the conceptual boundaries of the space using Wall objects and the physical boundaries of the space using Wall-Segments. The automatic generation and maintenance of the Wall-Segment objects has been one of the most difficult, however necessary part of the SGE functionality. It allows the user to freely move whole spaces at any time during the design process, which is most important during the early, schematic phases of building design.



**Figure 6.** Wall segmentation. Segment #2 is common to the boundaries of both Space #1 and Space #2.

### ADMINISTRATIVE ISSUES

While there is a general agreement on the conceptual model for interoperable software, there is no clear understanding on advantages and disadvantages on alternative options for its implementation. Unlike their development, the implementation of concepts depends greatly on the available technologies, which change continuously over time. The support for most implementation efforts, including the one presented in this paper, is barely enough to keep up with the progress in software and hardware technologies.

Seeing the development of a tool like the BDA as that of the design of a building, there are many ways to move ahead and decision-making on the best way to follow can only come through comparison of alternative courses of action. Moreover, just like for buildings, the development of such tools requires the involvement of many different professionals from the buildings as well as the information technologies disciplines. To properly move ahead in a timely fashion, we need support for a much larger and diverse team of professionals.

### CURRENT STATUS

We are currently nearing completion of BDA 1.0, which, in addition to the Schematic Graphic Editor and the Case Studies Database, is linked to a daylight analysis tool that uses the DOE-2 daylighting algorithms [Winkelmann 1983], and a thermal and energy analysis tool that uses the RESEM algorithms [Carroll et al 1989]. Moreover, in collaboration with the Center for Integrated Facility Engineering, at Stanford University, we are nearing completion of the links to an economic analysis module that takes into consideration first, construction, operation and maintenance costs.

### SHORT-TERM PLANS

Our plans for the future (BDA 2.0) include the development of links to more sophisticated simulation tools, such as the DOE-2 building energy simulation, the RADIANCE day/lighting and rendering, and the COMIS airflow and indoor air quality computer programs. Moreover, we plan to adopt the IAI building model and modify the BDA data structures and operation based on the performance of the 1.0 version and the comments that we get from its users. Most important we plan to seek collaboration with appropriate academic, research and industrial institutions so that we expedite progress and make it more efficient.

### THE FUTURE

The BDA development efforts were initiated to produce short-term results capitalizing on the availability of simulation tools like DOE-2, RADIANCE and COMIS. These programs have been developed over long periods of time and use old software technologies that are already obsolete, considering current methods and approaches. Eventually, they will be replaced by even more sophisticated and technologically innovative approaches in the years to come. These approaches will certainly be influenced by the development of tools like the BDA, which will facilitate their use through appropriate process control and data management. However, the opportunities and possibilities for new simulation tools should also affect the development efforts of tools like the BDA.

Unlike the currently available monolithic simulation programs, we expect that the future simulation algorithms will be modular with very fine granularity. To explore how such programs could work together, we have developed the BDA so that the output of a simulation program can be automatically directed as an input to another. In this way, we envision the user requesting a

computation that may result in the activation of several independent processes, in a chain-like reaction. In the current version of the BDA, if the user has not specified the size of the HVAC system when an energy computation is requested, then the BDA automatically activates an HVAC-autosizing module that provides the required input to the energy analysis module.

Finally, we see the manufacturers of building components and systems as major and critical players that will greatly influence the realization of the overall vision. Standardization of data requirements for performance simulation, as well as means to determine them, is one of the most important issues that need to be addressed.

## CONCLUSIONS

The development of a software environment that will support the integrated use of multiple simulation tools and databases is very complicated with several alternative courses of action that need to be explored. Most current development efforts do not have the required support to effectively explore and compare approaches in time to produce results that follow the general progress in information technologies. As a result, despite their potential benefits and promise, the development of such tools is slow and inefficient.

Unfortunately, the building industry is too fragmented to form a significant body of power that will demand fast development and eventually activate large teams through well-coordinated efforts. The development of tools like the BDA that seek the provision of mechanisms to lead to better buildings is more of a "public" need rather than a business opportunity. As such, it becomes a service that is most suitable for governmental organizations, which should lead efforts to explore alternatives and eventually pull the building industry together to realize the significant potential benefits.

## ACKNOWLEDGEMENTS

This work was funded, in part, by the California Institute for Energy Efficiency (CIEE), a research unit of the University of California. Publication of research results does not imply CIEE endorsement of or agreement with these findings, nor that of any CIEE sponsor. This work was also supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technology, State and Community Programs, Office of Building Systems of the U.S. Department of Energy, under Contract No. DE-AC03-76SF00098.

## REFERENCES

ASHRAE: *ASHRAE Fundamentals Handbook*, American Society of Heating, Refrigerating and Air-conditioning Engineers, 1993.

W.L. Carroll, B.E. Birdsall, R.J. Hitchcock, and R.C. Kammerud: "RESEM: An evaluation tool for energy retrofits in institutional buildings," Proceedings of the International Building Performance Simulation Association, 1989, pp. 107-112.

H.E. Feustel: "Annex 23 multizone airflow modeling – an international effort," Proceedings of the International Symposium on Air Flow in Multizone Structures, Budapest, Hungary, 1992.

IESNA: *IESNA Handbook*, Illuminating Engineering Society of North America, 1993.

M. Modest: "A general model for the calculation of daylighting in interior spaces," *Energy and Buildings*, Vol. 5, pp. 66-79, 1982.

K. Papamichael, J. La Porta, H. Chauvet, D. Collins, T. Trzcinski, J. Thorpe and S. Selkowitz: "The Building Design Advisor," *Proceedings of the ACADIA 1996 Conference*, University of Arizona, Tucson, AZ, October 31- November 3, 1996.

K. Papamichael: "Design process and knowledge; possibilities and limitations of computer-aided design." *Ph.D. Dissertation*, Department of Architecture, University of California, Berkeley, CA, August 1991.

K. Papamichael and J.P. Protzen: "The Limits of Intelligence in Design," *Proceedings of the Focus Symposium on Computer-Assisted Building Design Systems, of the Fourth International Symposium on System Research, Informatics and Cybernetics*, Baden-Baden, Germany, August 3-4, 1993.

S. Selkowitz, K. Papamichael and M. Wilde: "A concept for an advanced computer-based building envelope design tool." *Proceedings I of the International Daylighting Conference*, Long Beach, California, November 4-7, 1986.

G. J. Ward: "Visualization." *Lighting Design and Application*, Vol. 20, No. 6, pp. 4-20, 1992.

F.C. Winkelmann: "Daylighting calculation in DOE-2," LBNL report No. LBL-11353, III.2.9, Lawrence Berkeley National Laboratory, 1983.

F.C. Winkelmann, B.E. Birdsall, W.F. Buhl, K.L. Ellington, A.E. Erdem, J.J. Hirsch and S.D. Gates, "DOE-2 Supplement: Version 2.1E" Lawrence Berkeley Laboratory report no. LBL-34947, 1993.