Title
Integration of Simulation into Building Design: The Need for a Joint Approach

Authors
Augenbroe, G.
Winkelmann, F.C.

Publication Date
1989-12-07

LBL-28197

Lawrence Berkeley Laboratory
UNIVERSITY OF CALIFORNIA

APPLIED SCIENCE DIVISION

To be presented at the **Design Tools for Passive Solar and Building Energy Conservation,** 1990 ASME International Solar Energy Conference, Miami, FL, April 1–4, 1990, and to be published in the Proceedings

Integration of Simulation into Building Design: The Need for a Joint Approach

G. Augenbroe and F. Winkelmann

September 1989

APPLIED SCIENCE DIVISION

Prepared for the U.S. Department of Energy under Contract Number DE-AC03-76SF00098.

DISCLAIMER

Submitted to **Design Tools for Passive Solar and Building Energy Conservation**
1990 ASME International Solar Energy Conference,
Miami, FL — April 1-4, 1990

LBL-28197

INTEGRATION OF SIMULATION INTO BUILDING DESIGN: THE NEED FOR A JOINT APPROACH

Godfried Augenbroe
Building Physics Group
Delft University of Technology
Delft, The Netherlands

and

Frederick Winkelmann
Simulation Research Group
Lawrence Berkeley Laboratory
University of California
Berkeley, CA 94720, USA

September 1989

ABSTRACT

We describe the need for a joint effort between design researchers and simulation tool developers in formulating procedures and standards for integrating simulation into the building design process. We review and discuss current efforts in the US and Europe in the development of next-generation simulation tools and design integration techniques. In particular, we describe initiatives in object-oriented simulation environments (including the US Simulation Problem Analysis Kernel, the UK Energy Kernel System, the Swedish Ida system, and the French ZOOM program) and consider the relationship of these environments to recent R&D incentives in design integration methodology provided by the Commission of the European Community (the COMBINE program). Topics discussed include the role of simulation in building design, deficiencies of current energy performance evaluation tools, characteristics of intelligent building design systems, transfer of data and knowledge between simulation and design, and the STEP standard for the exchange of product model data.

This work was supported in part by the Assistant Secretary for Conservation and Renewable Energy, Office of Buildings and Community Systems, Building Systems Division of the U. S. Department of Energy under Contract No. DE-AC03-76SF00098

1. INTRODUCTION

Recent advances in simulation, computer-aided design, intelligent systems, and information technology raise important expectations for future integrated intelligent building-design systems (IIBDS's). In this paper we emphasize the critical area of simulation and its integration into IIBDS's. Reflecting the background of the authors, we concentrate on energy-related performance evaluation, which is taken as representative of the kind of simulation that can provide needed information in the building design process.

A conceptual framework is presented that shows the necessity for a joint approach among design and simulation researchers in developing IIBDS's. This framework also shows the challenges that will be faced in establishing design links between architecture and engineering, the building design professions that, in both Europe and the US, have traditionally acted as separate, non-integrated disciplines.

The concepts and goals of several ongoing research projects will be discussed. It will be shown that these projects can have a major impact on the design systems of the future, provided that their potential in a joint approach is recognized. To set the tone for the following sections, we give here a short assessment of what can be expected from IIBDS's.

Intuitively, it is clear that an IIBDS should have two major ingredients:

- A set of design support tools under complete control of the designer.
- A system in which these tools are embedded; this system should be able to provide intelligent assistance as to how and when to use a particular tool.

We make the following observations:

- A great variety of design support tools is currently available, each "tuned" to a specific design domain or goal (for example, presentation, specification, analysis, construction, etc.). These tools usually perform evaluations (for example, by calling specialized simulation programs) to support design decisions.
- In no way do we want to imply that the IIBDS's of the future will do "automatic" design. On the contrary, the designer will retain control over the creative process, with the IIBDS providing the information necessary to make decisions.
- The notion of a single person, a "superdesigner", at the controls of the system is by no means implied, nor is it realistic; an IIBDS would normally be used by several team members, each with individual expertise.
- We must acknowledge the fact that presently-available design and simulation tools, virtually all of which were intended for stand-alone use, cannot easily be integrated into IIBDS's. We will report below on a new generation of simulation environments that can solve this problem. It is also important to note that current CAD tools are aimed at drawing and display and thus provide very limited design support.

We will consider the integration problem from the point of view of the two basically different approaches that designers use. In the *top-down* approach, which is methodology oriented, the questions asked are when and how to do what, based on what information. In the *bottom-up* approach, which is performance oriented, the question is how a particular aspect or component of a building will perform. To be successful, an integration scheme must account for both approaches and provide an interface between them.

2. INTELLIGENT BUILDING DESIGN SYSTEMS

In order to provide intelligent assistance to the designer, any IIBDS must be based on a *process model*, i.e., on a general description of the design process. This process model should closely correspond to the working methods and "design scenarios" used by experienced designers. Any system that deviates from this by imposing a rigid and unnatural way of working will find very little acceptance.

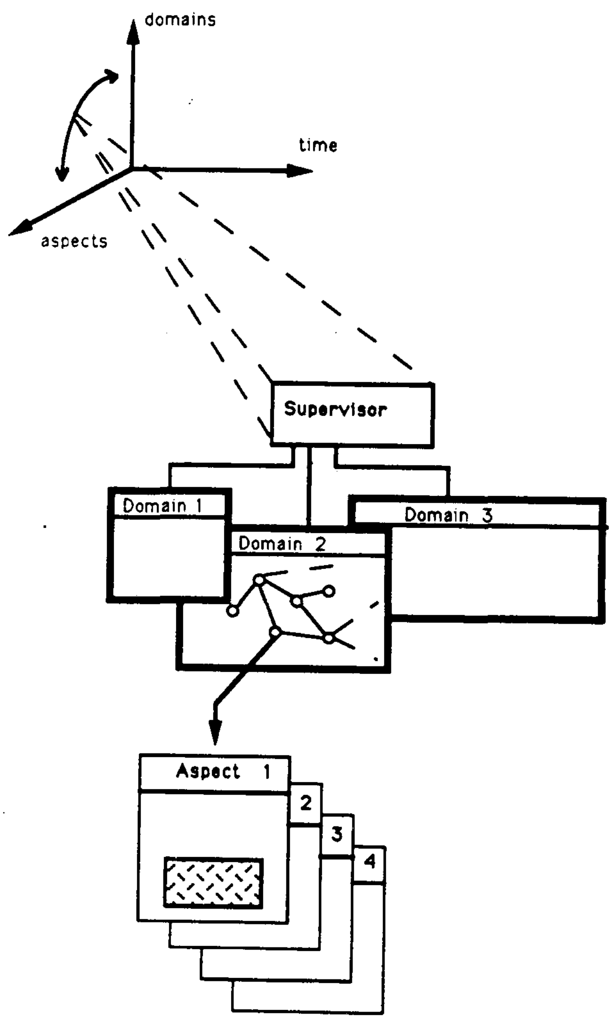

Fig. 1. Schematic of an Integrated Intelligent Building Design System

Although much work is being done on general design theories, a clear cut and widely accepted theory for the very complex process of building design [1,2] is a long way off. However, since our main focus is on interface issues and not on the development of a specific IIBDS, we can proceed without a design theory. In order to introduce some general ideas, we show in Fig. 1 a schematic model of an IIBDS.

In this figure we introduce the following abstractions, shown as orthogonal axes in process definition space:

- *time*: the design object goes through distinct phases — concept, briefing, preliminary design, etc., through to construction, operation and maintenance. As the design progresses, the description of the object becomes increasingly more concrete and detailed.
- *domains*: in building design it is common for several specialist teams to be involved at the same time, each performing what could be called a *design subprocess*. This notion suggests an essentially top-down approach to design. Examples of design domains are envelope design, interior design, lighting, HVAC design, construction design, and choice of materials [3].
- *aspects*: this axis represents the many different aspects (criteria and goals) on which we base our judgment of the performance of a design, e.g. energy-related, comfort, HVAC-related, environmental, costing, construction, functional, aesthetic, and psycho-sociological. In this paper we are primarily interested in the first three of these aspects.

This approach to the design process is basically top down. It must be kept in mind, however, that on different hierarchical levels we are dealing with multi-aspect and multi-domain activities. Generally, tasks are carried out concurrently, so that there has to be a supervising mechanism within each domain as well as across domains that resolves conflicting aspect- and domain-related suggestions and alternatives. An IIBDS should help the designer to make decisions, whereas an internal supervisor should provide all the necessary actions to maintain the coherence of the object description and support the negotiation process.

2.1 Availability of tools

At present there are very few integrated tools available, with some exceptions in limited domains, e.g. HVAC design [4]. It is common practice to hire specialist consultants to whom the design context and design object are communicated in some form or other. The form of communication usually leaves a lot to be desired. This is one of the reasons why these consultants are engaged mostly in the later design stages when the design context is limited and confined. Lack of integration inhibits their involvement in earlier stages.

Consultants usually handle only a small part of a design domain. They use specialized building performance evaluation (BPE) tools that are specific to their domain and generally operated in a stand-alone mode. Without integrated tools, supervision of concurrent processes in different domains is very difficult. As a result, supervision and negotiation are usually inefficient, with critical decisions made prematurely or based on insufficient information. Future design systems should be able to avoid this problem by offering:

- Easy communication of the description of the design object in machine-readable form.
- Links (interfaces) between an object description and domain-specific application tools.
- Guidance in using these tools.
- Support in supervising the design process (on different levels). An interesting idea in this respect is to regard the supervisor on the highest level (across domains) as merely a tool that enhances communication in meetings among design teams [5].

The level of assistance offered by a building design system can be classified as follows:

- *Little*: a limited and disjoint set of tools is provided, some of which may be very advanced; but a lack of integration significantly reduces their usefulness for design.
- *Intermediate*: some integration of design and BPE tools has been achieved; easy data communication among tools and some intelligent support are provided for one domain.
- *Extensive*: tools are completely integrated; easy data communication across all domains and supervisory support of the total design process are provided.

2.2 Identification of R&D requirements

In considering the R&D that is required for IIBDS's, it is useful to distinguish two different areas of integration:

- *Data Integration*: R&D in this area will lead to a standard for describing design objects and methods for making object descriptions available through a neutral format to different design domains, and within each domain, to different design aspects.
- *Process Integration*: this involves definition of the design context for any aspect-related task, such as performance evaluation, and handling the flow of information and decisions between these tasks, between design domains, and between designers.

To achieve both data and process integration requires a joint approach that is initially limited in scope, with future progress based on incremental improvements. First efforts could be limited to a part of the "time" axis (early design, for example), or part of the "domains" axis (HVAC design, for example), or part of the "aspects" axis (energy-related, for example). We feel strongly that R&D should acknowledge that the key issue is the *multicriterion* nature of design, so that any restriction to a set of criteria specific to a particular building trade or discipline should be rejected. Also, limiting the domain is acceptable only if the domain can be clearly identified with a design specialist (HVAC engineer, for example). Thus, we feel it would be best to initially restrict R&D to part of the time axis. The focus should, therefore, be on early design since the decisions made at this stage have the biggest impact on the final product. A good example of a program that is following this approach, i.e., providing multicriterion design assistance while focusing on early design, is ICADS (Intelligent Computer-Aided Design System) [6], an exploratory system under development in the US.

We note at this point that some recent design integration initiatives, such as that of Brambley et al. [7], fail to clearly distinguish these issues. Moreover, initiatives on bottom-up aspects (energy-related, for example) fail to acknowledge the fact that "energy" is not a design domain, so that there is no such thing as "energy design". Rather, energy-related *aspects* are present in all design domains and, therefore, must be dealt with in all phases of design.

Finally, we note that the failure of early, so-called integrated systems to penetrate the market stems mainly from their lack of an adequate process model, causing them to force an unnatural way of designing upon their users.

3. THE ROLE OF SIMULATION IN DESIGN

During the design process, decisions are often made based on an *evaluation* of the design as it exists at a particular point in time. The evaluation may involve many different aspects and may serve several purposes, such as checking code compliance, verifying goal fulfillment, choosing among alternatives, and satisfying budget constraints. At a lower level, the evaluation may involve extensive computation, such as calculating thermal loads or energy use. Based on such "hard" evaluation, judgment, and experience, the designer will be able to make adequate decisions and proceed with the design.

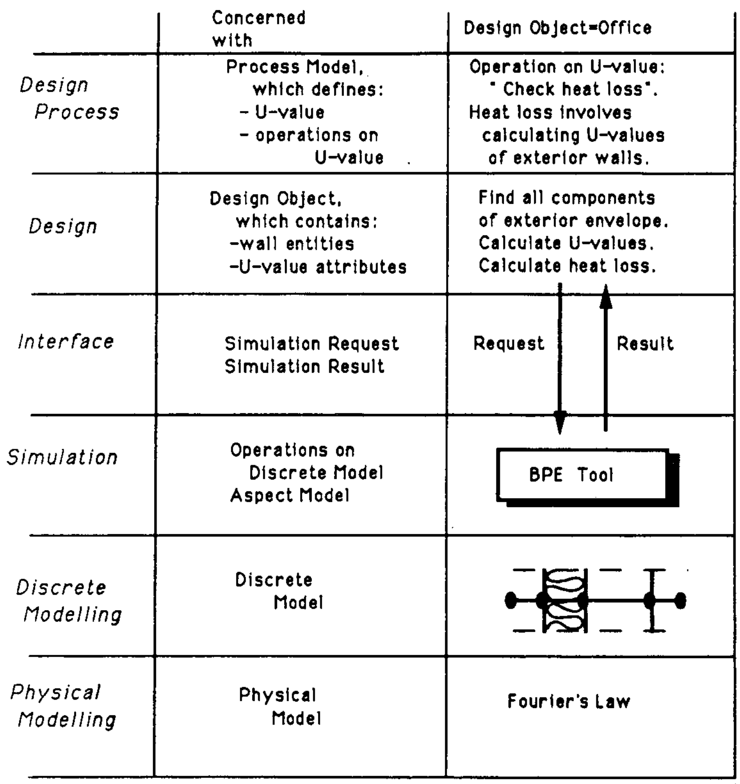

Fig. 2. Relation between simulation and design

Considering energy use evaluation in more detail, it is likely that a dynamic simulation of the building would be required to assess thermal performance as a function of the time-varying exterior weather conditions. In this case, the design system obviously needs to provide access to an appropriate simulation tool. *Integrated* use of this tool requires that physical knowledge from the field of heat transfer has to "migrate" into meaningful design information. This is illustrated in Fig. 2, which shows a simple example from the envelope design domain of determining the heat loss through exterior walls.

The diagram shows the top-down migration of general design knowledge and the bottom-up migration of physical knowledge. The interface layer handles the "client-supplier" relation by providing the translation of information in the two directions. It is interesting to note that this interface operates in present design practice mainly as a person-to-person communication via exchange of object descriptions (architectural designer gives drawings to engineering design consultant), formulation of a design-oriented request (which is cast by a simulation expert into a simulation input), and description of simulation output (to be translated into the design context by either the architect or the engineering consultant). This type of person-to-person interface is generally cumbersome, time-consuming, and inefficient, and thus performs poorly in everyday practice.

The general requirements of the interface are as follows:

- For *data transfer*: the interface has to map design-oriented data to BPE-oriented input. We discuss this further in Section 5.1.
- For *knowledge transfer*: the interface has to translate design requests into simulation instructions and translate simulation output into meaningful design rules.

For reasons of conceptual clarity, the distinction between knowledge (rules) and simulation (instructions) is maintained at this point. However, in actual implementations of design systems this distinction will be vague since the simulation will be an integral part of the reasoning rules. The actual simulation tasks will be carried out by running existing "external" software. This is explored further in Section 5.2. Later we will also address the possibility of replacing simulation altogether by rules (Section 5.4).

Further research is needed. For the top layers in Fig. 2, we need better models for the design process. These models will have to provide the purpose, context, and data for specific BPE requests. For the bottom layers, research on the integration of BPE tools and a methodological approach to validation in integrated environments are required. Also, additional research in needed to produce better simulation of physical process interactions (such as the coupling of the building envelope and the HVAC system and the coupling of interzone air flows and thermal loads).

4. ENERGY–RELATED BPE TOOLS

4.1 State of the art

Over the last 15 years, hundreds of energy-related BPE computer programs have been written for such applications as thermal comfort analysis, energy use calculation, HVAC equipment sizing, and lighting analysis. The spectrum of modeling approaches in these programs is quite broad. At the bottom range in terms of complexity are simplified methods that use fairly rough information about a building, such as the overall thermal conductance of the envelope and the number of degree days, and give a correspondingly rough indication of the performance of the building, such as annual heating load. In mid-range are programs that perform a quasi-steady-state hourly thermal calculation under actual weather conditions using transfer function or finite difference techniques. At the upper end are very complex, detailed programs exemplified by finite element methods for the solution of the Navier-Stokes equations for natural convection and component-based programs that calculate the minute-by-minute dynamics of HVAC systems by iteratively solving large sets of coupled differential and algebraic equations.

4.2 Deficiencies of current tools

Despite the range and power of the current-generation BPE tools, their use in building design practice has been very limited. For example, a 1987 survey by the American Institute of Architects [8] showed that only 10% of architectural firms in the US use BPE software. A similar situation exists in Europe. A number of reasons for this low level of use can be identified:

- The programs are hard to learn.
- The input, particularly geometric information, is difficult and time consuming to prepare.
- Each program has its own particular input and output format so that using more than one program on a project is particularly frustrating.
- Most programs require detailed input data, which makes them hard to use for early design (which is when energy-related design decisions are most important).
- Program output is hard to interpret and is often too sparse or too voluminous.

Other deficiencies common to most BPE tools that affect their reliability and extendibility are the following:

- The programs are non-modular ("monolithic"), with calculation methods closely intertwined with data structures; this makes them difficult to enhance, even for the original developers.
- Their parts are not easily reusable; a routine from one program can rarely be used in another program without extensive rewriting.
- There are no standards for testing and validation.
- It is usually impossible to determine the accuracy of a program for a particular design application; as a result programs are often misused.

There are also deficiencies of a more general nature that prohibit the straightforward integration of these simulation tools into design systems:

- The simulation language lacks expressiveness, which prevents an adequate translation of design-oriented requests into input to the simulation tool. Until recently there have been no attempts to develop flexible, modular, externally configurable simulation environments based on expressive simulation languages (see Section 6.2). Because current tools use predefined solution paths from numerical problem statement to numerical result, they can't directly handle most design requests, which are generally "inverse" (What should I do to get the desired answer?), "interrogative" (Why is this not what I expected?), or "incremental" (Do as before but slightly differently).

- There is a lack of expressiveness in describing the object being simulated. For example, most tools allow only a limited set of geometries and topological structures, which makes it difficult to map real design objects to simulated objects.
- There is a lack of explicit knowledge on how to use the tool. BPE tools require expert knowledge to translate the design request into proper input. Unfortunately, only part of this knowledge is explicitly available; the rest depends heavily on experience or creativity, or is hidden inside the tool in the form of the particular mathematical models and algorithms used in the simulation.

4.3 New developments

Near- and long-term efforts are under way to address these deficiencies. In the near term, BPE tools are being linked to computer-aided design and drafting (CAD) systems to simplify input of geometric data; interactive front-ends are being attached to programs to speed learning and data input; increased use is being made of graphics to assist in results interpretation; and BPE tools are being linked to expert systems as a first step towards incorporating domain-specific knowledge in the simulation process. However, the majority of these developments are rather mono-disciplinary in nature (i.e., bottom-up), so that they fall short of the desired design system discussed in Section 2. Although the products resulting from these efforts will increase the efficiency of specialized consultancies, a dramatic change in the present low level of use of these tools in architectural design cannot be expected.

There is, however, a long-term effort — the development of "object-oriented simulation environments" — that addresses the expressiveness deficiencies of current tools. These environments, which are discussed in more detail in Section 6.2, will produce the next generation of "user-friendly" BPE models and will facilitate the integration of these models into intelligent design systems.

5. INTERFACE REQUIREMENTS

5.1 Data Transfer

A building project deals with generating, updating and communicating an enormous amount of data. Formally, we denote the complete set of data about the design by the *design object description.* This description includes the topology and structure of the object, along with information that is relevant to particular tasks or participants in the design process. Such information spans data about costs, manufacturing, function, strength, color, tolerances, etc. Traditionally this description is stored and displayed in analog, segmented, and unstructured media, causing numerous problems associated with the ambiguity, incompleteness, and inconsistency of the information. A consensus is emerging in the computer industry that the key to integration will be the definition of complete data models for each product type that will satisfy all of the above information needs. A major effort in this direction, the formulation of the STEP standard, is described in the next section.

This complete data model is generally called a ***product model.*** Although we are mainly concerned with design in this paper, it must be realized that integration based on product models reaches beyond the design stage, spanning the entire life cycle of the product. Present product modelling efforts reflect this broader Computer Integrated Manufacturing (CIM) scope.

5.2 The STEP standard

Since 1983, subcommittee TC184/SC4 of the International Standards Organization (ISO) has been working on a standard for the exchange of product model data (ISO/STEP). Other efforts in the same area, e.g. PDES (in the US) and CAD*I (in Europe), have produced substantial input to the STEP-efforts, but will not be discussed here. At present, the first version of STEP is about to become an ISO draft proposal [9]. STEP's main target is the exchange of multiple representations of the design object between computers. These representations are critical in integrated environments since each domain, aspect, and particular simulation tool requires a different representation of the same object. We will refer to these different "views" of the object as *aspect models.* The key to providing a standard that permits the exchange of data is the definition of a central and complete product model, serving as a reference from which all aspect models can be derived [10,11]. Figure 3 shows how this is accomplished.

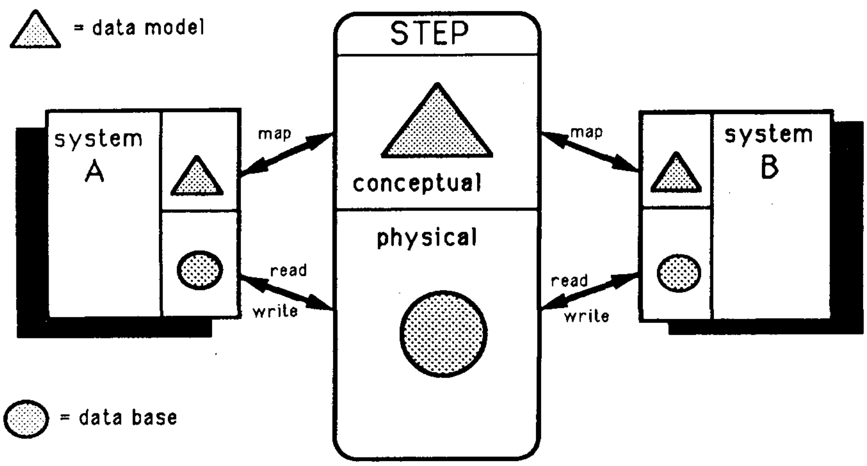

Fig. 3. Data exchange between two systems using a product model standard.

The data model is specified in two layers, a *conceptual layer* and a *physical layer.* The conceptual layer is the exchange reference; it serves as the basis for implementing the physical layer (the neutral format for the storage of data), making possible the actual exchange of the product model data. Interface standardization based on this neutral format is as yet undefined.

In Fig. 3, systems A and B might be two different IIBDS's. If both systems were (independently) developed with the awareness of the emerging STEP standard, there is a fair chance that the exchange of object representations between the systems would be possible without much loss of information content. However, it is to be expected that the standard will be so huge that a complete, one-to-one mapping between systems will be unattainable in practice. The standard will however enable the specification of the adherence to a particular subset of the standard (to be specified for system A and system B by their respective developers). Obviously both systems would have to supply a STEP translator to make the actual data exchange work. In this way the two systems (addressing different design domains, for example) could be easily integrated.

The present situation shows a sharp contrast with this picture. Current systems use their own conceptual data model, confined to the limited scope for which the system was designed. As a result, there are few "overlapping" entities in the three different conceptual models belonging to A, STEP, and B, which makes a mapping from A to B virtually impossible.

Special attention should be given to the "internal" integration within each of the systems A and B (supposing both systems to be themselves integrated, i.e., consisting of separate CAD and BPE software components). Depending on the development philosophy, the internal integration could be either based on the same external product model or be completely hidden or in some other way hard-wired inside the system. For proprietary reasons, it is unlikely that the private market will be interested in "opening up" their products. However, for publicly-funded R&D initiatives it is important to ensure interchangeability of components among systems and to have a standardized approach to integration within systems. (Note that in the latter case, Fig. 3 can be interpreted as the exchange of information between components within a single system.)

If we look closer at the components in a system, application-dependent aspect models will be found that are too specific to be part of the official standard. For this reason the STEP standard will present a layer concept that leaves room for these application-dependent models at the bottom layer, as shown in Fig. 4 [12].

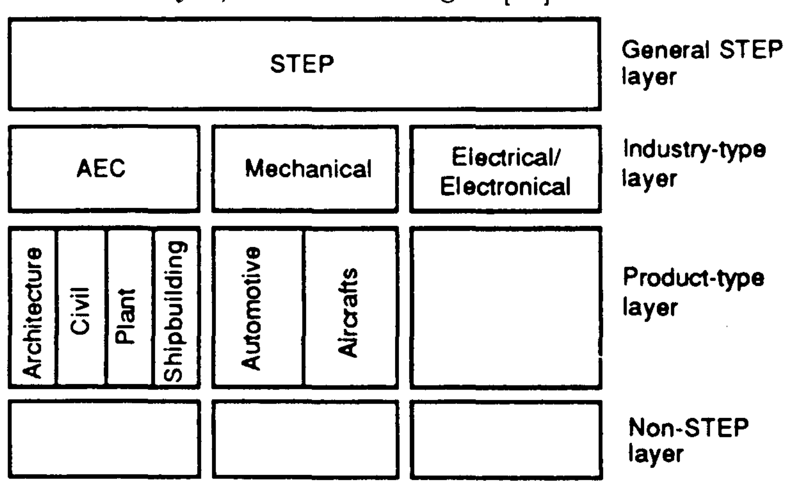

Fig. 4. STEP abstraction levels.

The present STEP draft proposal contains a general architecture, engineering, and construction (AEC) reference model (GARM). Future research will supply more specific models for architectural products, i.e., buildings. It is important to note that additional standards (application, national, enterprise) might be added at the bottom layer. Typically, these additional standards would evolve from efforts to develop integrated design systems from the energy and HVAC viewpoint. Future STEP enhancements will provide room for this kind of additional input [12].

We conclude this section with the following remarks.

- The STEP effort is long term but provides exciting challenges.
- STEP is an exchange standard for "external" representations; it does not apply to the "internal" representations. How we store and manage data internally in a system is dictated by functions the system is expected to perform.
- For architectural products, the draft GARM is an important proposal that contains useful abstraction mechanisms and powerful concepts and entities able to support design activities [13,14].
- The working methods (involving tools for data modelling, integration, and specification) used to produce STEP provide a useful toolkit for the data integration problem at hand.

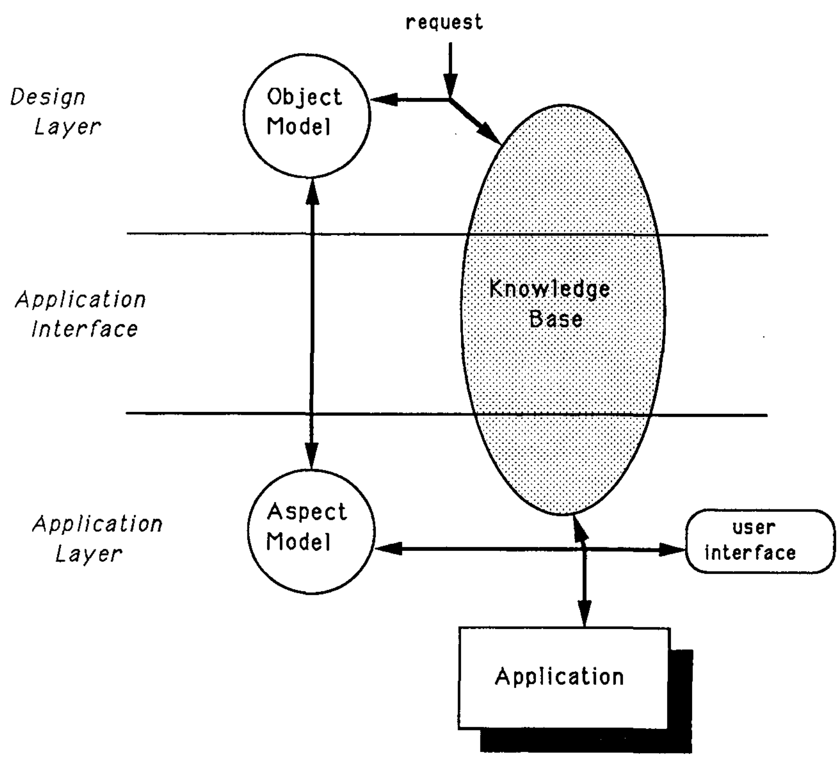

Fig. 5. Functions of the application interface supported by the knowledge base.

5.3 Knowledge transfer

For a design system to be both integrated and intelligent it must offer more than just data exchange capabilities between its components. Although we won't attempt to specify a system software architecture for an IIBDS, it will be necessary to specify some general functionality in order to define the kind of intelligence that should be present at each level within the system. Since we are especially interested in the interface between design and application (see Fig. 5), we will take a closer look at the type of knowledge that is required to support the functions of that interface [15].

Assume that at a certain point in the design session a request for information by the designer (issued through the user interface of the IIBDS) is interpreted as a request that a particular application be invoked. The knowledge base in the design layer should then be able to carry out the following tasks:

(1) Check the validity of the request and choose the appropriate application.

(2) Inspect the object model to determine whether it contains the information required by the application.

If no errors are detected the request is sent to the application interface, which should then:

(3) Translate the object model into the aspect model for the application.

(4) Invoke the application's user interface, which prompts the user for additional input and selective actions when appropriate.

Through combined support from the application interface and the IIBDS interface, the following takes place:

(5) Configure the application tool to carry out the desired simulation (through generation of input data or simulation instructions).

(6) Run the application (interactively, under user control)

(7) Add the resulting output data as "extensions" to the aspect model.

If the user exits the application layer, the following tasks will be carried out by the application interface:

(8) Inspect the extensions to the aspect model; translate them and then add them to the object model.

Knowledge rules in the design layer must be able to interpret the new data in the object model and provide the designer with context sensitive responses to his original request. We note that the above tasks involve knowledge rules for design process (tasks 1,2,8) as well as for modelling and simulation (tasks 2,3,4,5).

We make the following observations:

- Implementation of tasks 1-8 requires powerful software environments in which knowledge rules and data bases can be integrated. It must be recognized that task 3, for example, is not an algorithmic process; it involves knowledge rules "inside" the object and aspect models.

- BPE tool developers will be required to express the modelling and applicability knowledge of their tools. They will have to adopt standardized formats for this purpose [16,17].
- BPE tool developers will be required to define aspect models for their tools.
- Tasks 5,6,7 will be greatly enhanced by future object-oriented simulation environments (Section 6.2).

5.4 An alternate route

Until recently it was felt that intelligent design systems should be based exclusively on simple design rules or other types of knowledge bases, such as those obtained from detailed parametric studies using simulation programs. Direct use of simulation in the design process would thus be avoided. The obvious benefits resulting from this approach would be (1) an application layer would not be needed since all knowledge would be available in the design layer; and (2) there would be no time-consuming simulation calculations. However, based on the experience of other design areas, we offer the following reasons why relying only on design rules for buildings will not work:

- Every building project is unique and seldom routine or simple; it is thus unlikely that preconceived, generic design rules will always be applicable to the case in hand.
- The use of simulation tools makes the vast amount of internal modeling knowledge embodied in them directly available to the designer without the need to extract and reformulate this knowledge. As explained above, the application interface requires only the external knowledge about the use and applicability of the tool.
- The increasing availability of inexpensive, fast computers means that compute-intensive simulation can be done during design with acceptable response times.

6. RECENT R&D INITIATIVES

6.1 The COMBINE project

Following an initiative taken at the Commission of the European Communities (CEC) workshop on the future of building energy modelling [18], it was decided that the CEC would fund work in the area of integrated design systems. The funding would be allocated within the scope of the JOULE program, which addresses energy-related research. COMBINE (Computer Models for the Building Industry in Europe) was chosen as the (very broad) name for this topic. Prior to the call for proposals, a definition study was carried out [19]. The actual research, to last about two years, will start in 1990. At the present time (mid 1989), a detailed research plan for COMBINE is being formulated. The main objective of COMBINE is to take a first step towards full integration of energy-related BPE tools into intelligent design systems. Although the development of a full-blown IIBDS is far beyond the scope of this project, the research will formulate the underlying concepts that will enable the BPE discipline to play an active role in IIBDS development and will provide an incentive to the user community and to CAD software vendors to seek the participation of the BPE discipline in future IIBDS developments.

The COMBINE project is expected to result in:

- A tested and validated approach to the interface issues regarding the coupling of BPE tools to current and future building design systems.
- A common data model for several BPE tools and several CAD tools. This model will be conceptually close to the emerging AEC-STEP standard discussed in Section 5.2.
- Two (or more) prototype integrated design systems that will appeal to both designers and CAD vendors. The emphasis of the prototypes will probably be different. For example, one may be based on existing state-of-the-art tools from different design domains. The emphasis in this case would be on ease of integration, and the system's intelligence would mainly be aimed at this functionality. Another might be a small-scale intelligent design system with the emphasis on providing design assistance.
- Implementation of a common data model in a suitable software environment.
- A first step in creating common goals and working procedures among BPE research groups, and between BPE groups and design research groups.

6.2 Object-oriented simulation environments

As we have already pointed out, the monolithic, non-modular nature of current-generation BPE tools makes them extremely difficult to adapt to the future needs of designers. However, such adaptation is crucial in order to provide users with up-to-date models that can simulate advanced building components and HVAC technologies, and that can take advantage of the improved solution techniques and user interfaces that will make the programs more robust and easier to use. To overcome these difficulties, model developers have been investigating new methods of structuring simulation programs. Out of this has emerged the idea of *object-oriented simulation environments* in which models of arbitrary complexity can be built by linking together calculation objects.

Four such environments that are under development in different countries are:

- in the US: the Simulation Problem Analysis Kernel (SPANK) [20]
- in Sweden: Ida (formerly MODSIM) [21]
- in the UK: the Energy Kernel System (EKS) [22]
- in France: ZOOM [23]

In SPANK and Ida the calculation objects are differential and algebraic equations that describe physical processes. In the EKS, objects are algorithmic procedures extracted from existing programs. In ZOOM, objects are "cells" (spatial domains or physical components) and "transfers" (quantities that can be exchanged between cells).

Although the structures of these systems are quite different, they share a number of common features:

- A processor links calculation objects together to form simulation models.

- According to the standard object-oriented programming paradigm, the methods and data associated with a calculation object are encapsulated; i.e., they are internal to the object and cannot, in general, be altered by other objects.
- Classes of objects can be defined, then instantiated to create particular instances of an object for use in a simulation model.
- Small objects can be assembled in a hierarchical fashion into larger objects (macro-objects or submodels).
- Objects and macro-objects can be stored in a library.

Such simulation environments provide several important advantages relative to traditional methods of program development:

- Depending on the objects selected and how they are linked together, a broad spectrum of models can be assembled, ranging from simplified methods appropriate to early design, to detailed methods appropriate to final design.
- Objects can easily be added to a model, and the internal calculation of an object can be modified without "knock-on" effects in the rest of the model. These features make models easy to upgrade and extend.
- Objects can be reused at a later time for building other models.
- Objects can be shared among different simulation environments if they are expressed in a standard form, then translated for use in a particular environment. To accomplish this, a "neutral model format" for calculation objects has been proposed [17].

Because of their modularity and flexibility, these simulation environments have the potential to facilitate the integration of simulation into intelligent design systems. Models appropriate to the time, domains, and aspects coordinates of design can be created with a simulation environment and then incorporated in an IIBDS. Alternatively, the simulation environment itself could be imbedded in the IIBDS, so that application models tailored to the design questions as they emerge could be generated "in real time", executed to provide answers, and then saved for later use or released at the end of the design session. Whichever model-creation approach is taken, the potential of object-oriented simulation environments will only be realized if the developers of these environments and and the developers of design systems begin to work together now to formulate the mutual specifications and protocols that will allow this integration to proceed naturally and efficiently.

7. CONCLUSIONS

Based on the awareness that only a small fraction of the buildings that are designed today undergo an energy performance evaluation, we have argued that the next generation of design-support software should offer the designer easy access to these evaluation tools in integrated design systems. We have suggested that the successful development of such design assistants requires a "top-down" effort by the design community to define real-world design process models, and a "bottom-up" effort by simulation researchers to refine and validate their calculation models and to develop flexible simulation environments that will facilitate integration of these models into design systems. We have stressed that a joint approach, combining these two efforts, will be necessary.

R&D requirements from the view of the energy-related BPE community have been defined, with an emphasis on the link between the design layer and the performance evaluation layer in integrated design systems. A recent joint initiative in Europe promises to fulfill these requirements. Strong links with ongoing efforts to develop new object-oriented simulation environments will have to be initiated. We feel that organizations like the International Building Performance Simulation Association (IBPSA) in North America, the Building Environmental Performance Analysis Club in the UK, or a future European organization have major roles to play in establishing these links.

8. REFERENCES

1. Gero, J.S., 1987, "Expert Systems in Computer Aided Design", Proceedings, IFIP WG 5.2 Working Conference, Sydney.

2. Gero, J.S., 1985, "Knowledge Engineering in Computer Aided Design", Proceedings, IFIP WG 5.2 Working Conference, Budapest.

3. Dupagne, A., 1987, "Applying a Solid Modelling Language to Built Form Representation and Evaluation", Proceedings, Workshop on Future Energy Modelling, Ispra, pp. 59-78.

4. Doheny, J.G. and Monaghan, P.F., 1987, "IDABES: An Expert System for the Preliminary Stages of Building Energy Systems", Artificial Intelligence in Engineering, 1987, Vol 2, no 2,pp 54-64.

5. Dupagne, A., 1989, Private Communication.

6. Pohl, J. and Chapman, A., 1988, "An Expert Design Generator", Architectural Science Review, Vol. 31, pp. 75-86.

7. Brambley, M.R., Crawley, D.B., Hostetler, D.D., Stratton, R.C., Addison, M.S., Deringer, J.J., Hall, J.D., Selkowitz, S.E., 1988, "Advanced Energy Design and Operation Technologies Research: Recommendations for a U.S. Department of Energy Multiyear Program Plan, Pacific Northwest Laboratory Report No. PNL-6255.

8. "AIA Firm Study Profiles U.S. Design Market", Building Design and Construction, November 1987, pp. 19-20.

9. "ISO - TC184 / SC4: External Representation of Product Definition Data", ISO Status Report, 1989.

10. Turner, J.A., 1988, "A Systems Approach to the Conceptual Modelling of Buildings", Proceedings, CIB Seminar on Conceptual Modelling of Buildings, Lund.

11. Reed, K.A., 1988, "Product Modelling of Buildings for Data Exchange Standards; from IGES to PDES/STEP and Beyond". Proceedings, CIB Seminar on Conceptual Modelling of Buildings, Lund.

12. Tolman, F. and Gielingh, W., 1989, "STEP Planning Model", ISO-TC184/SC4/WG1-document.

13. Gielingh, W., 1988, "General AEC Reference Model (GARM)", Proceedings, CIB Seminar on Conceptual Modelling of Buildings, Lund.

14. Willems, P., 1988, "A Meta-Topology for Product Modelling", Proceedings, CIB Seminar on Conceptual Modelling of Buildings, Lund.

15. Rogier, J. and Tolman, F., 1989, "De Formulering van een Ontwerpparadigma voor bouwkundige Ontwerpers", Proceedings, Cape '89, Amsterdam.

16. Laret, L. and Dubois, A.M., 1988, "Modelisation et Simulation de Fonctionnement Thermique des Batiments et des Equipements Energetique Associes", CSTB Report MGL/88-1031/NB.

17. Sahlin, P. and Sowell, E.F., 1989, "A Neutral Format for Building Simulation Models", Proceedings, Building Simulation '89, International Building Performance Simulation Association, pp. 147-154.

18. Hattem, D. van, 1987, ed., Proceedings, Workshop on Future Building Energy Modelling, Commission of the European Communities, Ispra.

19. Augenbroe, G. and Laret, L., 1988, "COMBINE, Pilot Study Report", Commission of the European Communities, Brussels

20. Sowell, E.F., Buhl, W.F., Erdem, A.E., and Winkelmann, F.C., 1986, "A Prototype Object-based System for HVAC Simulation", Proceedings, Second International Conference on System Simulation in Buildings, Liege.

21. Sahlin, P., 1988, "MODSIM, a Program for Dynamical Simulation of Continuous Systems", Proceedings, The 30th Annual Meeting of the Scandinavian Simulation Society.

22. Clarke, J.A., 1988, "The Energy Kernel System", Energy and Buildings, Vol. 10, pp. 259-266.

23. Bonin, J.L., Grandpeix, J.Y., Joly, J.L., Lahellec, A., Platel, V., and Rigal, M., 1989, "Multimodel Simulation: the TEF Approach", Proceedings, European Simulation Conference, Rome.

LAWRENCE BERKELEY LABORATORY
TECHNICAL INFORMATION DEPARTMENT
1 CYCLOTRON ROAD
BERKELEY, CALIFORNIA 94720