Building Energy Simulation:
The State-of-the-Art

J A Clarke BSc PhD
Energy Simulation Research Unit
Department of Architecture
University of Strathclyde
Glasgow G4 0NG
Scotland, UK

Abstract

Most contemporary computer aided building design (CABD) systems aim to offer draughting *and* performance appraisal through an advanced graphics interface. At the present time Vendors are expending some effort on the latter, and some are even contemplating a simulation based approach to building energy appraisal. The ability to preview environmental and energy behaviour at the design stage, set against the building's other cost and performance attributes, is therefore an emerging possibility.

This paper describes the evolution of building energy models: From the early handbook methods to today's comprehensive thermodynamic simulators. As an exemplar, the form and content of one advanced model – the ESP system (Clarke 1985) developed at the University of Strathclyde – is described in terms of its theory, accuracy, user interface and relevance in a design context.

Finally, the paper outlines the deficiences of the present generation and describes the research now underway in Europe and North America to bring about a next generation energy model architecture.

Introduction

In the field of building design, the development of CABD has been underway for twenty years or so (Mitchell 1977). During this time the economic factor has ensured that the draughting function has received much attention by system Vendors and is now becoming well established in the marketplace. Software to aid in the design process, on the other hand, is much less developed. The complexity of design, stemming from its multi-variate nature, makes it difficult to devise a computer based approach which would perform well and be generally accepted in practice. One aspect of the design activity has received much attention however: The appraisal of building performance throughout the design process (Maver 1982). Here powerful, computer-based models are created to represent the range of cost, performance and visual impact issues of importance in design. From life-cycle cost estimates at the outline proposal stage; through realistic visualisations of the design prior to construction; to comprehensive evaluations of building and plant energy performance. At the present time, a demand is is steadily growing for systems which possess both the appraisal *and* draughting functions. And in response, Vendors have appended appraisal programs to their proprietary draughting packages. The resulting CABD system is then, increasingly, offered on an inexpensive micro or on personal workstations utilising 32 bit chip technology and state-of-the-art bit-mapped displays.

The energy appraisal system described in this paper – known as ESP – is one of a number of systems developed at the ABACUS CABD Unit at the University of Strathclyde. Together, these systems allow the prediction of a building's cost, performance and visual attributes at the schematic and detailed stages of the building design process.

Energy Modelling

Traditionally, building and HVAC system designers have relied on a myriad of manual calculation methods as the basis of component (building and plant) selection at the design stage. These methods are based on numerous empirical simplifications and, in many cases, are confined to the steady state calculation domain. In view of the complexity of a building's energy sub-system, the usefulness of such an approach is now being seriously questioned by many designers and researchers. Figure 1 highlights this complexity and is reproduced here as a concise statement of the modelling challenge.

The building, plant and control elements can be viewed as physical regions of vastly different time constants, with complex spatial and temporal interactions. For example, time varying boundary conditions cause complex transient effects; control actions are highly temporal and essentially non-linear; heat and mass transfers are inextricable; and the interactions between regions of different time constants pose fundamental numerical difficulties. Superimposed on this are the time dependent, perhaps non-linear flow-paths themselves. The problem is a truly integrated and dynamic one, requiring modelling techniques which respect the laws of thermodynamics and causality.

Model Evolution

The evolution of building energy models is summarised in the following table.

1st Generation	Handbook orientated Analytical in formulation As simplified as possible Piecemeal in approach		Indicative Application limited Easy to use Difficult to interpret
2nd Generation	Dynamics important Still analytical Still piecemeal Suitable for low order	Feedback loop	increasing integrity

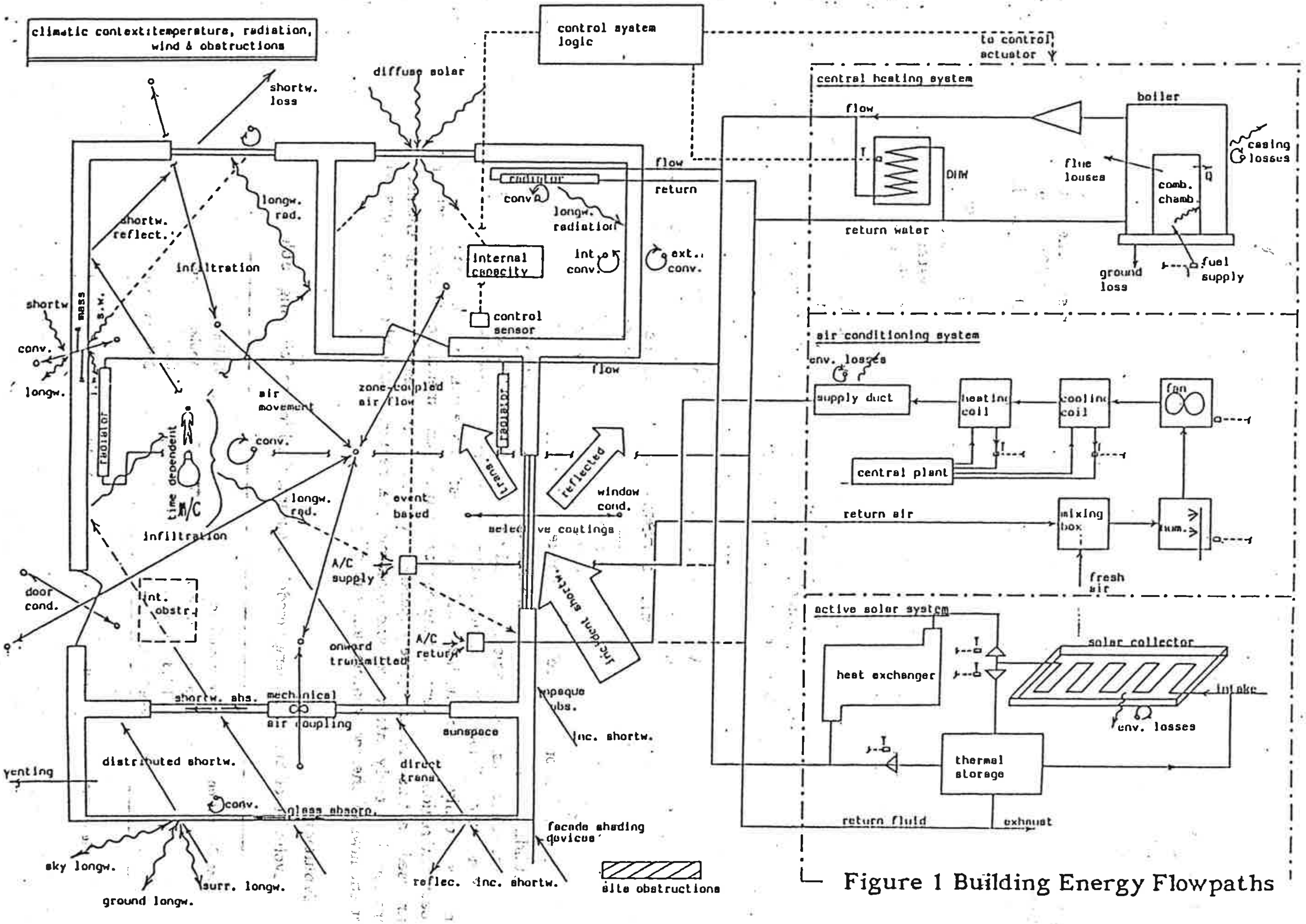


Figure 1 Building Energy Flowpaths

	problems with time invariance	vis–a–vis the real world
3rd Generation	Field problem approach Requiring numerical methods Integrated view of energy sub–system Suitable for high order problems with time variation Heat & mass transfer considered Better user interface and partial CABD integration	leading to
		see next table

1st Generation

Traditionally, building designers have relied on a range of calculation techniques to quantify building performance at the design stage. The professional handbooks abound with such techniques, regarded by many as the ultimate tools for design appraisal. This 1st Generation 'handbook' approach – including computerised manual methods – is piecemeal, in that no coupling is evident between the various discrete calculations; in fact, the implication is that the designer is the coupling mechanism. For example, a steady state U–value calculation may be used to quantify envelope heat loss, a handbook table may then be consulted to determine an allowance for zone solar gain, and a degree–day formalism may be relied on to achieve an estimate of long term energy requirement. The individual calculations are analytical, embodying many simplifying assumptions to permit their formulation in the first instance. The approach does not attempt to faithfully represent the actual energy and mass flow–paths which occur in real buildings. Instead, the intention is to provide the designer with an indication of performance. Such methods are easy to apply but difficult to interpret since the designer is required to appreciate the application limits of each calculation type, taking into account the complex sub–system interactions by relying on her/his experience. This implies the need for expert knowledge on the part of the user. Also, since many of the real world flow–paths and interactions are degraded or ignored, the integrity and overall accuracy of the handbook method is low; for example, the commonly assumed steady–state, uni–directional wall heat flow rarely, if ever, occurs in the real world. And so, pursuant to the oil embargo of '73, the modelling challenge of the past decade has been to raise model integrity by explicitly representing the active flow–paths and the observable interactions.

2nd Generation

In the mid–70's a 2nd Generation approach emerged. Now the temporal aspect of energy flow began to receive attention, particularly in the case of long time constant elements such as multi–layered wall, roof and floor constructions. The underlying calculation methodology was still piecemeal and analytical in nature; for example, time (Stephenson and Mitalas 1967) or frequency (Mackey and Wright 1944) domain response functions may be used to obtain the envelope's dynamic response to climatic stimuli, weighting factors or energy balance techniques may then be used to achieve zone response by invoking the superposition theorem, and the cooling and heating loads to result may be used as the basis for separate, steady–state design calculations addressing the building's systems and central plant. The approach is wholly appropriate if the system for simulation is linear and can be described mathematically by parameters which are time invariant. Modelling integrity remains relatively low because of the inherent decoupling and the simplifying assumptions applied to the flow–paths. For example, the nature of any control link between the building and its plant, if included, will be treated in a rudimentary manner. And air movement and radiative heat transfers are usually time invariant prescriptions. While offering an improvement on 1st Generation models, a number of new problems now emerge. The need for large mainframe processing and the creation of extensive input data sets poses an insurmountable barrier to most designers. And the complete absence of appropriate user interface techniques serve only to widen the credibility gap between the model developers and the design end-users. As a result many research groups now operate 2nd Generation models in parametric mode, creating a feedback loop which attempts to improve the quality of the paradigms and simplified methods on offer to the profession.

3rd Generation

In very recent times, 3rd Generation models have begun to emerge which attempt to resolve these difficulties by theoretical extensions and improvements at the machine interface.. Typically the building system is considered as a field problem in which the only true independent variables are the space dimensions and time. All other quantities (flux exchanges and variables of state) are entirely dependent and fully coupled across space and time. No single process can be solved independently and so simultaneous processing methods are required. Finite volume (or element) discretisation is preceded by the generation of conservation equations for each volume and for each property to be conserved – such as energy, mass or momentum. The overall equation set is typically a mix of partial and ordinary differential equations and algebraic expressions. The equations may well be non-linear, sometimes complex. Whole-system equation-sets are *stiff* and, if viewed in matrix equation form, topologically sparse. Advanced numerical methods are therefore required to achieve efficient and accurate time-step integration. 3rd Generation models are therefore suitable for complex transient problems, exhibiting weak and strong couplings between components. Combined heat and mass transfer has been addressed for the first time, and there has been a distinct shift towards more appropriate user interfaces utilizing graphical I/O and the new processor/display technologies (Workstations, *Unix*^{TM}, etc.). Modelling integrity has been raised substantially so that the systems have become more predictive vis-α-vis reality, more generally applicable and, because of the improved interface, easier to use. However, an important problem confronting 3rd Generation programs is that numerical representations of HVAC equipment requires knowledge of heat capacities, fluid heat transfer coefficients and component geometry. This data is generally unavailable or extremely difficult to obtain. A major reason that some 2nd Generation models are in wide use is that they simulate a variety of building equipment in terms of quasi-steady-state performance curves, readily obtainable from manufacturers' data. Also, and because of the lack of any complementary technology transfer mechanism, the credibility gap remains so that the most productive use of the technology is still to generate performance knowledge or simplified methods for containment in a 1st Generation frame. Before describing the 4th Generation developments now underway in Europe and North America, the paper continues with a description of one 3rd Generation model.

ESP: A Third Generation Exemplar ?

As shown in Figure 2, ESP is a multi-module system which permits an appraisal of the energy performance of existing or proposed building designs, incorporating traditional and/or advanced energy features.

Using the system, a designer is able to conduct a high integrity, first principle performance appraisal, modelling all aspects of the energy subsystem simultaneously and in the transient domain. The contention is that design performance assessment requires modelling methods which have knowledge of the complex interactions found in the real world. In other words, a piecemeal approach, in which a particular region is considered in isolation, is totally inappropriate and often misleading. At an early design stage, a model such as ESP is particularly powerful since it can be used to quantify the performance impact of the site, the building geometry and construction: all factors which have considerable impact on operational performance and costs. Then, at the more detailed design stage, the model allows the designer to focus on specific issues such as control and comfort. The following table lists the input data required by ESP to enable such an analysis.

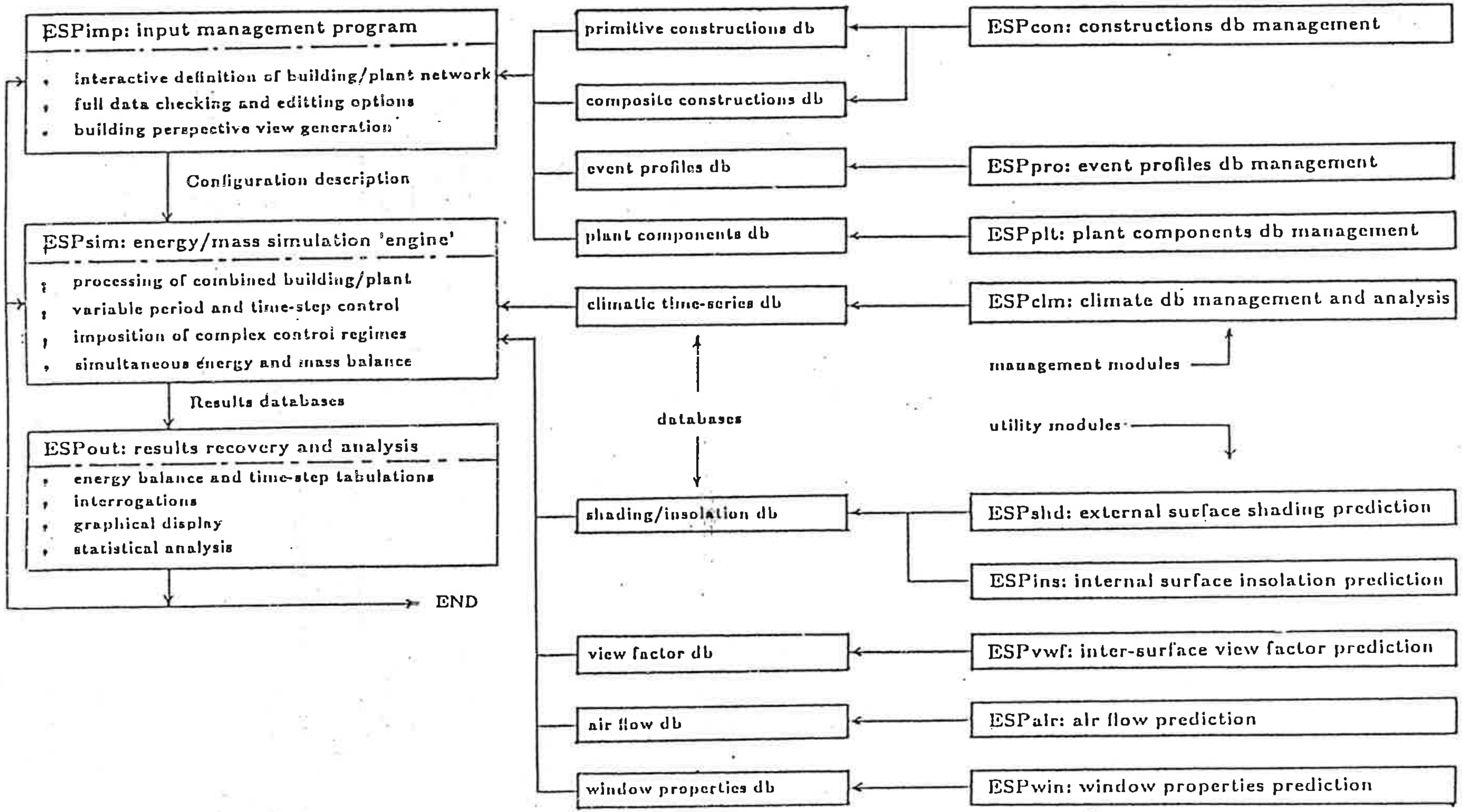


Figure 2 Program Modules of the ESP System

Category	Item
General	Site details Climatic time series
Building	Geometry Construction Operation Leakage distribution
Plant	Components Connections
Controls	Functions Loops
Simulation	Periods Constraints

Obviously much of this data may be difficult to obtain, especially at the earlier stages of the design process. This is not necessarily a justification for a simplification of the laws of thermodynamics so that a model can be developed which operates with a reduced data set. An alternative possibility is to retain the best representation of reality – a high integrity model – and, instead, to generate the complete data set from whatever information the designer can offer at any stage. This approach lays emphasis on *intelligent* design tools as opposed to *simplified* design tools. In the former case a knowledge base is established to supplement the information supplied by the designer (see later).

Theoretical Basis of ESP

ESP uses a numerical method (Clarke 1985) to solve the various equation types (algebraic, ordinary differential and partial differential) which are used to represent the heat and mass balances within buildings. For each real–world energy flow–path (as shown in the previous figure), ESP has a corresponding mathematical model. Within a simulation, this numerical method ensures that all modelled flow–paths evolve simultaneously to fully preserve the important spatial and temporal relationships. The system is non–building type specific and can handle any plant system as long as the necessary component models are installed in the plant components' database. Components, if missing, can be added by a user so that they become available for selection to define a multi–component plant network.

In addition to the usual energy analysis features (causal breakdowns, plant sizing, comfort assessment, condensation checks, and the like), ESP is equipped to handle the spectral analysis of glazing systems, the time varying shading caused by site obstructions, solar ray tracing, pressure and buoyancy induced air movement, and complex, distributed control systems. A conceptual explanation of the model's calculation technique would go something like:

ESP accepts a building/plant description in terms of 3–D geometry, construction, usage and control. The continuous system is then made discrete by division into many small, finite volumes of space – perhaps as many as 10,000 for a medium sized building. These finite volumes represent the various regions of the building and plant within and between which energy and mass can flow. Throughout a simulation, ESP relies on *state–space* equations to track the energy and mass balance for all finite volumes as they evolve under the influence of the system boundary conditions (climate & control) and the constraints imposed by the inter– and intra–volume links (resistance & capacitance). This technique ensures that all regions of the building are correctly connected across space and time and so any excitation at some point in space or time will have the correct causal effect. Volume time constants are tracked dynamically and state–space equations adjusted between pure implicit and mixed implicit/explicit to ensure solution stability. The following table lists the finite volume, state–space formulations currently active within ESP.

building–side processes	transient conduction buoyancy driven surface convection pressure & buoyancy driven air flow facade & site induced shading shortwave radiation solar mapping to internal geometry internal and external long–wave radiation casual inputs
plant–side components	boiler radiator hot water cylinder pipe pump cooling coil heating coil humidifier mixing box fan duct solar collector heat exchanger thermal store
control–side elements	sensors actuators control laws
components under development	heat wheel heat pump heat pump chiller cooling tower

And the system's modular structure allows the addition of new models or the replacement of existing ones.

Validity of ESP

Many model users do not yet appreciate the complexities of large scale energy model validation. Indeed it is probably true that absolute declarations of validity will never be possible. This is because the data used to describe the problem is itself subject to great uncertainty. And because the combinatorial links between input assumptions, logic representations, mathematical techniques and output interpretations is very large. It is now widely recognised that confidence levels can only be improved by developing verification methodologies and by repeatedly undertaking studies which aim to verify discrete aspects of a model under different operating conditions. In this respect ESP has a good track record. It was/is a participant model in the International Energy Agency's Annex 1, 4 and 10 projects (Fabers 1980, Clarke and McLean 1983, McLean and Clarke 1985, Tang 1984) concerned with inter-model and empirical model verification in both building and systems simulation modes. ESP has also been tested by several organisations external to ABACUS and found to agree well with known analytical solutions (Gough 1984), monitored data sets (Williams 1984) and inter-model comparisons (Clarke 1987a). The program has been declared the European Reference Model for Passive Solar Architectural Design (Archard and Gicquel 1987) and has been examined extensively in a UK collaborative research project to develop a model verification methodology (SERC/BRE 1987). ESP is also being critically examined in a major EEC project (Gicquel and Cools 1986) in which several National Consortiums are using identical solar test cells to critically test ESP and develop test methodologies for passive solar design elements.

ESP's User Interface

ESP is a highly graphical, interactive program, driven by menu command selection. Traditionally, this type of program has required a mainframe environment with an attached graphics terminal. In recent years computing hardware has changed dramatically, providing advanced processor/graphics combinations at low cost. This technology is enabling the introduction of advanced interfaces and the incorporation of a degree of artificial intelligence within models such as ESP. The following table summarises the possibilities.

An appropriate computing environment.	32 bit micro high capacity hard disk floppy/tape unit large RAM ethernet/RS232 bit-mapped, high resolution screen window manager programmable keyboard mouse Unix operating system eg SUN3, APOLLO DN320, Whitechapel MG-200
An advanced graphics I/O capability.	icons perspective displays menuing
And the incorporation of a degree of intelligence within the application program with respect to problem specification and performance appraisal.	dynamic defaults form fill-out performance methodologies knowledge bases

ESP is now operational on a range of low cost workstations (5,000 and up at 1987 UK prices) with a greatly improved interface offering pop-up menus, form fill-out data entry, dynamic defaults and the like (Clarke and McLean 1986). Recently, developments have commenced which seek to transform ESP into an intelligent knowledge based system (*ikbs*). Two approaches to the design of a building energy *ikbs* are being pursued: One, a short term and pragmatic approach, is currently operational; the other, a more fundamental approach, is in the early development stage.

A Pragmatic Approach to an Energy ikbs

In essence this approach involves the use of the Unix Command Interpreter or Shell (Bourne 1982) as a pseudo expert system shell. Scripts are designed to coordinate the operation of ESP and other objects (programs) against the rules and relations of particular performance assessment methodologies (Clarke 1986). Script rules, although hard-wired, can be replaced by the designer at Script invocation. Within a Script, ESP and other relevant objects act together to perform set operations as a function of the in-built rules. The computational path to be followed at any stage in the Script will depend on the performance data to emerge at run-time and on the embodied rules. Each Script can be viewed as a design assistant: The performance assessment and program operation knowledge is known to the assistant; the designer is free to focus attention on design decision making. The Scripts are developed to operate on personal workstations offering a bit-mapped display controlled by a window manager.

To demonstrate the possibilities and the technique, one ESP Script is described here. Its purpose is to undertake a comfort analysis with the following mission:

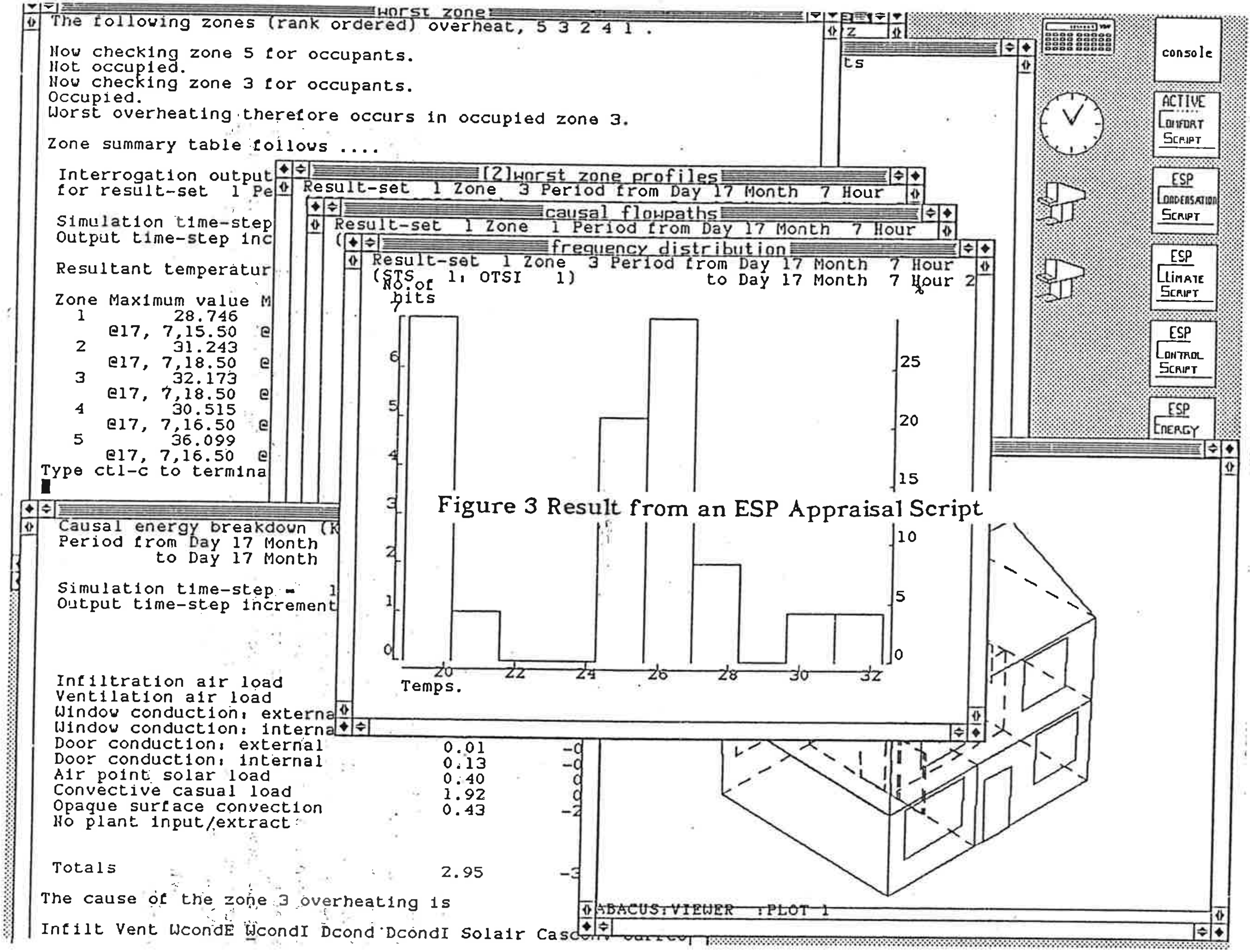
- To determine an appropriate simulation boundary condition by selecting a climatic collection with a severity rating matched to the building's geographical location and function.
- To initiate and control the simulation processing over a period of time determined as a function of severity criteria.
- To seek out building zones which are uncomfortable according to user specified (or default) comfort criteria.
- To present statistics on the prevailing comfort levels.
- To determine the cause of the problem.
- To initiate a sensitivity study, focusing on the causal energy flow paths, and to so rank order the options for design intervention.
- And to provide a comprehensive report on comfort performance, including problem causes and potential cures.

Firstly, the Script uses the ESP climate module **clm** (see previous figure), and a rule on the relationship between climatic severity and building type, to determine a meaningful climatic context. Then **sim** is used to conduct a detailed simulation against this climatic influence. Secondly, comfort statistics are recovered, zone-by-zone, and rank ordered in terms of function-related criteria. This is done using the Unix **awk** process. Performance data (frequency of occurence, temperature profiles, etc) on the worst zone offenders are then output. Thirdly, the cause of any discomfort is investigated by analysing energy balance information, always following the worst causal flow-path at each branch point. Lastly, a sensitivity analysis is commissioned to determine the most productive design intervention strategy. In effect, a Script is the controller for a particular automated appraisal, possessing the knowledge on how to concurrently process ESP programs and Unix tools. It is also an electronic representation of application knowledge such as appropriate comfort indices (SET, PPM and PMV) and their subjective mappings.

When invoked, this Script, after a computational effort which will depend on the complexity of the building and the length of the required simulation, will produce a screen image of the form of figure 3.

The following table lists the Scripts now available to drive ABACUS programs.

Heating and cooling plant sizing.
Climatic severity assessment.
Plant control strategy appraisal.
Condensation analysis.
Comfort investigation.



```
The following zones (rank ordered) overheat, 5 3 2 4 1 .

Now checking zone 5 for occupants.
Not occupied.
Now checking zone 3 for occupants.
Occupied.
Worst overheating therefore occurs in occupied zone 3.

Zone summary table follows ....

 Interrogation output
 for result-set  1 Pe

 Simulation time-step
 Output time-step inc

 Resultant temperatur

 Zone Maximum value M
   1          28.746
      @17, 7,15.50  @
   2          31.243
      @17, 7,18.50  @
   3          32.173
      @17, 7,18.50  @
   4          30.515
      @17, 7,16.50  @
   5          36.099
      @17, 7,16.50  @
Type ctl-c to termina
```

```
Result-set  1 Zone  3 Period from Day 17 Month  7 Hour
```

```
Result-set  1 Zone  1 Period from Day 17 Month  7 Hour
(
```

```
Result-set  1 Zone  3 Period from Day 17 Month  7 Hour
(STS   1, OTSI   1)                to Day 17 Month  7 Hour 2
```

```
 Causal energy breakdown (K
 Period from Day 17 Month
          to Day 17 Month

 Simulation time-step -   1
 Output time-step increment



 Infiltration air load
 Ventilation air load
 Window conduction: externa
 Window conduction: interna
 Door conduction: external         0.01      -0
 Door conduction: internal         0.13      -0
 Air point solar load              0.40       0
 Convective casual load            1.92       0
 Opaque surface convection         0.43      -2
 No plant input/extract


 Totals                            2.95      -3

The cause of the zone 3 overheating is

Infilt Vent WcondE WcondI Dcond DcondI Solair Casc
```

Figure 3 Result from an ESP Appraisal Script

Energy requirements and causal breakdown.
Solar utilisation.
Air flow analysis.
Perspective view generation.
Cost–in–use
Regulations compliance

A number of additional Scripts are also under development which aid in the building description process. Here the aim is to match Scripts to different building types and levels of knowledge. For example, one Script might accept a simple description of a passive solar building and generate the full data set as required by ESP. Another Script might accept a detailed description of an air handling unit, adding its own typical office description to allow a meaningful plant appraisal.

A Detailed Approach to an Energy ikbs

Although the Script technique outlined previously offers an intelligent interface to energy simulation, it suffers from three fundamental limitations. Firstly, it is constrained to the performance appraisal aspect of the problem; it does not, for example, address the problematic issues surrounding data preparation in the face of uncertainty. Secondly, each Script is considered as an independent design assistant. This implies that the user must be able to coordinate Script selection and to act as the overall integrator. And thirdly Scripts do not allow 'Why do you ask ?' type responses. They have no real understanding of the system they address; they are merely clever prescriptions.

It would obviously be attractive to design an ikbs which could act as an expert consultant, recognising the user's plan, commissioning simulations and reporting back on overall performance. This is the goal of the next version of the ESP system. Such a development is now the subject of a SERC collaborative project (MacRandal 1986). What is envisaged is the system architecture shown in figure 4.

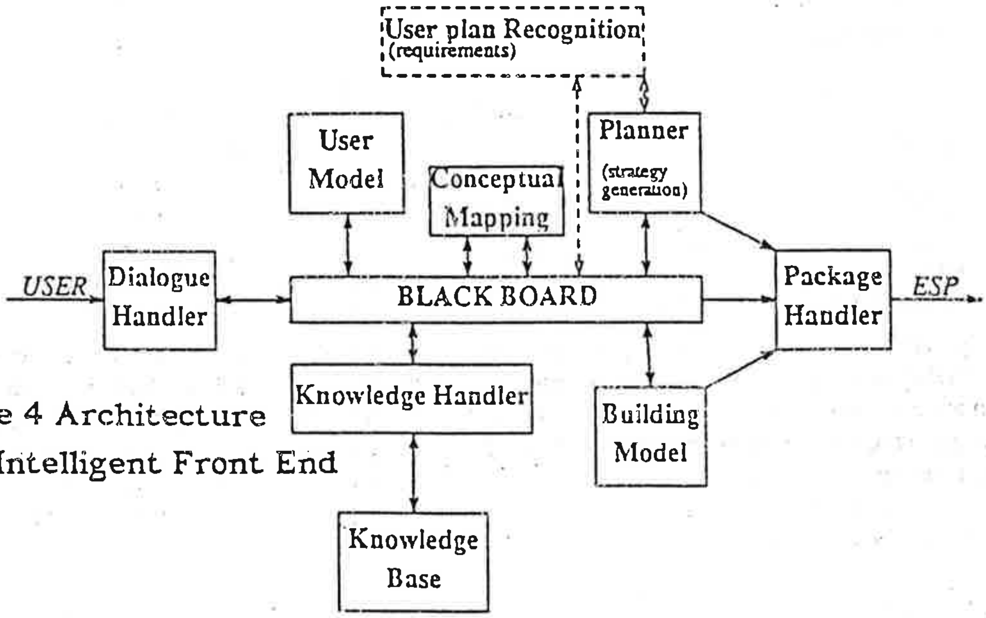


Figure 4 Architecture of an Intelligent Front End

A central *blackboard* exists to manage information traffic. Each module of the system can examine this center for relevant information, posting results where appropriate.

The *knowledge base*, implemented in Prolog, holds both application knowledge (concerning building physics) and modelling knowledge (defining simulation strategies).

The *dialogue handler* is the user communication mechanism. This controls the consultation session, allowing a user to volunteer information, to redirect the systems line of inquiry, or to make 'Why do you want to know ?' type responses to the systems prompts.

The ikbs will possess more than one *user model.* For each user type - architect, engineer, energy modeller, student, etc. - at least two categories, naive and proficient, are envisaged.

The *user plan recognition* module exists to interpret the user's objectives and to generate a list of performance assessments which will meet these.

The *planner* provides the complementary function, namely the generation of a list of simulation requests which match the required performance assessments.

The *building model* assembles the full data set which describes the building problem. These data can come directly from a user, if available, or from the knowledge base in the form of dynamic defaults which may depend on the user dialogue.

And, lastly, there is the ESP *handler* which extracts the data from the building model and planner to permit the simulations.

Deficiences of the Current Generation

At the present time then, the extant systems for building energy simulation are a mix of 2nd and 3rd Generation. These are the systems which, with continuing refinement over the coming years, will find their way into the CABD systems and seek to replace the traditional, 1st Generation techniques. Indeed a growing number of practitioners, educationalists and researchers are currently struggling to find ways to cost-effectively apply these systems in practice. But what of 1990 and beyond? While this new technology offers sophisticated modelling capabilities, there are many deficiences which will restrict future refinements to satisfy the needs of an increasingly more demanding user base. For example, any existing system will suffer to some extent from one or more of the following.

- The software structure is often extremely inflexible and unyielding. The program will have been conceived in a now outdated machine environment - probably batch oriented, possibly with card input. This means that the structure is monolithic, imposing extreme management and updating difficulties.
- To date, it has not been possible to formulate a common approach to the simulation problem. Funding policy and the ever present competition within and between the private and public sector has served only to stifle inter-organisation collaboration. The result is that each system has customised I/O procedures and unique internal structure. Physical models are different in each code, as are linking protocols, network representations, theoretical emphasis and so on. This situation is obviously divisive and generates little confidence in the developments taking place.
- Often system authors are energy specialists, but coding amateurs. The software structure is inelegant, with the application knowledge inextricably bound to the source code. Contemporary software engineering favours the separation of the domain specific knowledge from the procedural software. This is a fundamental prerequisite of intelligent knowledge based systems which, if disregarded, can become a serious barrier to software evolution. For example, with existing systems it is often difficult to upgrade heat transfer algorithms since this may require the detailed knowledge of data structures, internal memory and the side effects of one change on the rest of the software package. More elaborate modifications, such as substituting one numerical integration scheme for another, are at present intractable problems.
- Many of the currently available systems are of unacceptably low integrity. In the pursuit of a simplified design tool, many real world flow-paths may have been oversimplified or omitted, and the issue of causality has not been addressed. An unsuspecting user is then left to struggle with the inadequacies of a model which has unacceptably degraded the laws of thermodynamics in the name of model reduction.

The whole issue of model validity and end–use appropriateness is only now becoming better appreciated.

- And finally, there exists no clear statement on long term objectives and task sharing developments. As a consequence many existing systems are not well tailored since each author organisation has been forced to address every element of the problem: I/O, heat transfer theory, database design and management, solution techniques, software structure, validation, documentation, etc. It is clear that no single organisation will possess the necessary expertise in all areas. Each system is then promoted in a manner which implicitly undermines the development effort expended on its contemporaries. This is clearly an intolerable situation and one which serves only to fragment the development community.

The Future

If the challenge is to overcome these problems is accepted, then the time is right to devise a plan of action which will allow effective community wide collaboration and focus our energies toward next generation possibilities for building energy modelling. By observing developments in computer science generally, and past model evolution in particular, it is possible to anticipate the future. Completing the previous table:

4th Generation	Full CABD integration More advanced numerical methods Intelligent knowledge based Object–orientated software architecture	↑ Feedback loop ?	Predictive Generalised Easy to use

It is evident that there is no single evolutionary path leading to 4th Generation models and beyond. Instead, the promise of the future is in its pluralism. Highly focused models living beside generalised ones; first principle approaches interlinked with the empirical; and an explosion of approaches in pursuit of intelligent knowledge based systems. From this viewpoint stems an axiom for the future: We must be able to construct models of arbitrary complexity from the same corpus of procedural primitives. This is the only way to foster state–of–the–art and facilitate the technology transfer process.

At the present time there are several software engineering trends which are influencing the way in which future models will be constructed. There is the move to an object–orientated programming environment to provide better modularity, the emergence of better software tools and methodologies for program proving, the general availability and consolidation of means for providing more effective user interfaces, and the current shift away from functional only programs to programs which embody a degree of human–like intelligence.

These are the themes which have come together in the *Energy Kernel System* (*eks*) project now being pursued, collaboratively, by research groups in Europe and North America.

The Energy Kernel System

The *eks* (Clarke 1987b) is a software/hardware environment which allows the construction of a program from pre–existing objects which represent accredited techniques in a particular application field. It is a program development environment, offering a high degree of flexibility and encouraging, through the ease with which new techniques can be integrated with the old, state–of–the–art developments. The specific goals of the *eks* are:

- To permit a free evolution of alternative modelling methodologies.
- To enable and encourage collaboration between model developers, and between developers and end–users.

- To promote state–of–the–art developments through ease of integration.
- To separate data structures, calculation procedures and application knowledge.
- And to remove the burden of machine portability and other hardware/software problems from the model builder.

The *eks* accepts a specification of any modelling system in terms of the objects which comprise it and the inter–object links. A multi–object program is then assembled and output as source or executable code for some target machine. While the *eks* is targeted for low cost, personal workstations, it will generate programs which can run within other machine environments. One of the objects is merely the controller for the others. It defines the run–time environment and the features it will offer. Consider figure 5 which shows the elements of the *eks*.

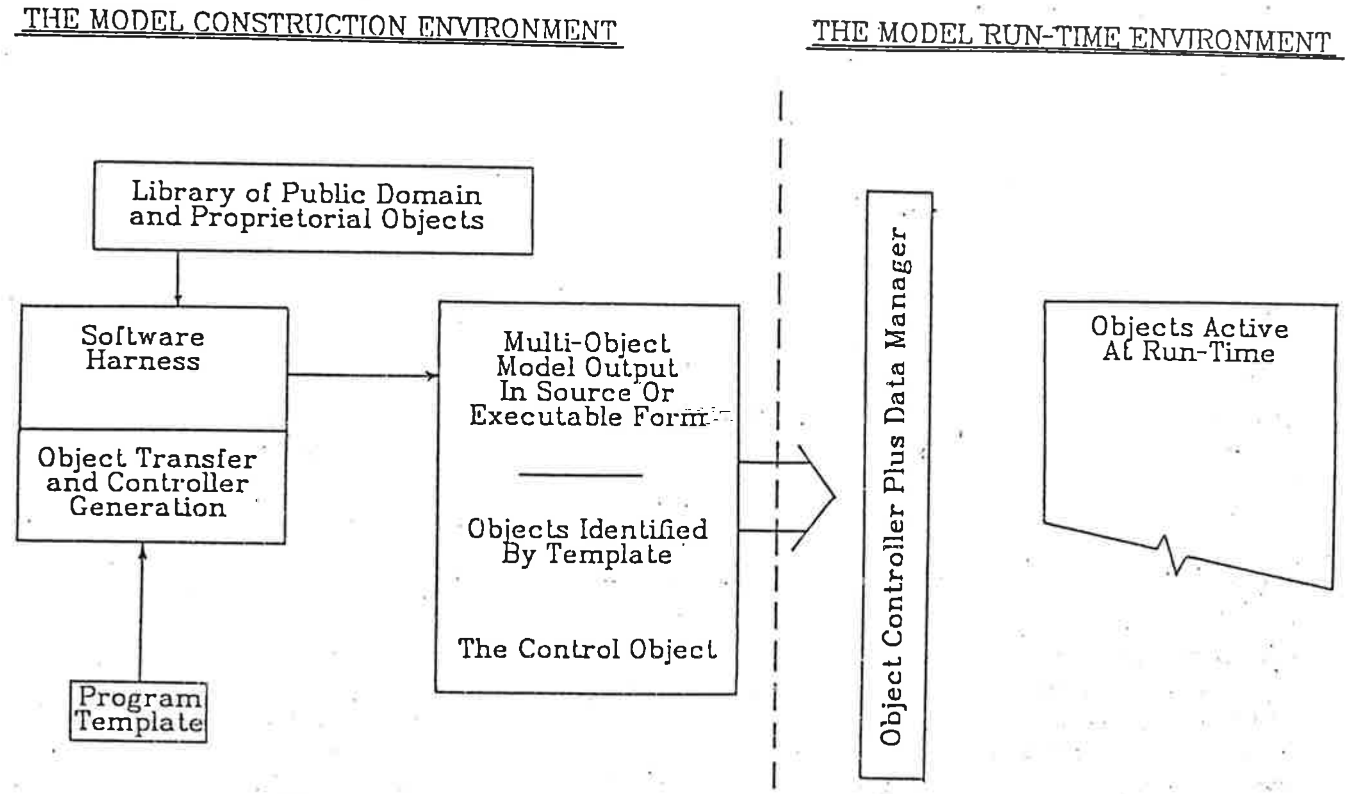


Figure 5 The Energy Kernel System

- A *template* holds the application knowledge. It is constructed to define the objects and inter–connections which will comprise a program. A proposed program is decomposed into primitive operations, each one represented by an object. An object is therefore a dedicated function program which exists to perform a single, usually small, task. It has no knowledge of the 'outside world' or of the data requirements of other objects. It is the template's job to represent both the objects and the data stream connections which are required. This is done via a command syntax which permits the definition of multiple inter–object data links and supports object clustering to form macro objects.

  Although model developers can develop completely new templates from scratch, it is also possible to modify existing ones. In this way templates will evolve so that future systems can more easily build on their previous generations. Indeed it is likely that the principal mechanism for software transmission in future will be by template: The received template being passed to the *eks* at the remote site. Since each local Kernel can be given machine specific information (see the harness description), problems of machine compatibility can be largely overcome.

- The template is passed to the *harness* which then constructs the program, outputing it in the form of a multi-object model, expressed as source or executable code. This is done by transferring the selected objects from the *object's library*, and automatically generating the control object which will coordinate data traffic at program run time.

  The output from the harness is therefore a program targeted for a specific machine. While the *eks* itself is designed to operate only on low cost workstations utilising the Unix operating system, the programs it generates can be targeted for any machine as long as the harness is designed to generate an appropriate object controller.

- *Objects* exists to hold the working primitives which are manipulated by the Kernel. These are organised by standard file classification as defined by the Unix operating system: That is, placed in a directory structure and classified by type. All objects have a corresponding manual entry so that documentation is on-line. And the source code is available for all public domain objects.

  Objects would likely be written in c, or c^{++} because of the latter's inheritance feature, superior type checking, the ability to operator overload and the fact that compilation produces standard c code. When objects are created their input and output data is arranged into *structures*, each one containing a data type. Objects are then connected by couplings structures as required. Any structure can simply be discarded if not required. This means that well designed objects can be used in a number of different applications. If a particular data structure is not exactly as required, then a filter object will be needed to combine the elements of the structure, discard some elements or rearrange their order. Special purpose filters will be installed in the Kernel for this purpose.

Many useful objects already exist within contemporary modelling systems. And much additional software continues to emerge as existing codes are extended or new developments pursued. The first task is to extract this knowledge from existing models, re-express it as self-contained, logically independent entities, and to place these within the *eks*. Focusing on existing programs has the merit that the entire modelling methodology is defined and cast in a functional form understood by all contemporary modellers. Commencing with the development of entirely new objects, targeted for a new program architecture, would incur the danger that the methodology would be debated rather than the role of the *eks* in building it. As an example, the following table lists the range of objects which could be readily freed from a system such as ESP:

Object type	Examples
Geometry	Areas; volumes; angles; sun position; shading; insolation; specular, diffuse and off-specular view factors; perspectives.
Properties	Psychrometrics; time constants; dimensional groupings.
Climate	Severity assessment; prediction; curve fitting; statistics; interpolation and extrapolation; manipulation.

Object type	Examples
Heat Transfer	Longwave radiation; short–wave radiation; surface convection; sky temperature; comfort; air flow; moisture; window spectral behaviour; control volume conservation equations.
Numerics	Sparse matrix methods; super–matrix construction and partitioning; scaling and pivoting; finite difference transformations; Newton–Raphson; Trapezoidal integration; Taylor series expansions; time–step control.
General	Heat transfer models; control laws; matrix topology control.
Data and I/O	File management; error trapping; graphics; statistics; menuing; terminal (colour) drivers, database formatting.
mmi & ikbs	Intelligent menuing; icons; performance assessment rules.

By acting in this way, there is no attempt to prescribe or limit new models to the architecture of the present generation. Instead, it is the intention that the *eks* can reproduce existing models so that these can serve as a starting point for model evolution for those who do not wish to follow an entirely new course.

- An *object dictionary* holds object definition and a description of data I/O. The former is a statement drawn up by an object's author to cover capability, theory and limitations. The latter is the definition of the object's data structures, defined by several type classifications (constant or variable; integer, float, complex or character; physical, numerical or filter; temperature or enthalpy; and so on).

It is important to note that the Kernel has no knowledge of the application; it can make no assumptions about the form or capability of any model. Instead, it allows the construction of *any* modelling approach, subject only to the availability of suitable objects. For this reason, objects are not constrained to conform to any particular format. They are self–contained entities which exist only to operate on data. There is no direct communication between them. There is no direct communication between objects. Data structure connection is only symbolic; all objects receiving data from, and returning it to, the controller at run–time.

An eks Generated Program at Run–time

The *eks* is a model builder. It is an efficient way to develop and maintain CABD, and, in particular, energy programs. At run–time, a program will not appear any different from a program developed by conventional methods. A user will simply interact with the interface (whether good or bad) as defined by the program's template. Within the machine things will be radically different however. When the program is invoked the first task is to load the control object. This in turn loads the remaining objects comprising the entire program and establishes the data space which corresponds to the given problem. The controller then schedules these objects based on a *data flow architecture*. In this scheme, a data network topology is continually scanned. When an object's input data structures are ready, the data is passed to the object. This action activates the object which performs its function, passing back its output data structures before suspending itself.

Advantages of the eks Approach

The modular approach offered by the *eks* provides a flexible framework for future model development, testing and management. For example, no single organisation need be expert in all areas in order to formulate a whole-building model. On the other hand, anyone who wishes can undertake the development of either general or special purpose whole- or part-building models using the software primitives and the harness. New capabilities, in the form of additional software primitives, can be added as they are developed and proven. Modelling methods - detailed and general, micro and macro - can be mixed within one computational environment, and can even be changed at run-time. Models can be constructed which represent the entire building system; from form and fabric, to the systems, plant and control action. Alternatively, a decoupling can be introduced at any interface to facilitate theoretical examination or model reduction. Mixed one, two, and three dimensional schemes will be possible for the case of transient conduction modelling since more than one primitive can exist for any function. And any external database system could be used to supply the application data - a new 'get' object is merely added.

Other research activities will continue as before. The hope is that the *eks* will act to bring about resource sharing by making available the results of one group's labours to another group in the form of a new object. Individual groups will continue to evolve mathematical models of the components and sub-systems found in buildings; to establish and test solution techniques suitable for the integration of non-linear differential equations; to develop advanced man-machine interaction techniques and intelligent knowledge-based methods for user communication; and to formulate methodological templates which define a program's structure in terms of the many internal connections among its parts. These are the activities which the *eks* seeks to order and make available.

Technology Transfer

In the UK, an energy modelling research and application club is being set up (Clarke 1987, Bloomfield 1987b) to address the issues of model use in practice. With a membership drawn from public and private sector organisations and academia, the Club mission would be to critically examine the relevance, reliability, applicability and *modus operandi* of current, and in time, future, modelling systems. In North America a similar organisation - the International Building Performance Simulation Association (IBPSA) - has been incorporated to pursue similar objectives. The hope is that the international dimension of IBPSA will grow so that the possibilities for simulation in the building design process can be nurtured.

Conclusions

A number of detailed energy simulation models have now emerged and the first attempts are being made to incorporate these into existing CABD packages. With advances in machine technology, the performance/cost ratio of these systems is rising rapidly and the technology is becoming more available.

ESP is an advanced energy model developed in the University sector. In addition to its native, interactive graphics mode, the system can now be operated in expert mode. A number of Scripts have been developed to control ESP's program modules against rules which relate to particular performance assessment methodologies. Each script is then equivalent to a design assistant, endowed with knowledge of ESP and the application domain. The shortcoming of this approach is that the internal representation of knowledge is primitive and inflexible. To remedy this, a more ambitious ikbs system is needed. Such a system is now under development.

To address some of the problems at a fundamental level, funding has now been obtained in the UK (SERC, DoE, DEn)) and the USA (DEn). This will allow the development of a Kernel System for the construction of advanced building energy simulation models. This development should facilitate the goal of a next generation model architecture, possessing high integrity and easy to modify as new theories, interface techniques, validation procedures and performance assessment methodologies emerge.

This, and the developments in knowledge engineering, will finally allow the creation of a truly useful computer–aided approach to building design.

References

Archard P and Gicquel R (eds) 1987 'Basic Principles and Concepts for Passive Solar Architecture' *European Passive Solar Handbook* Ecole des Mines, Sophia Antipolis.

Bourne 1982 *The Unix System* Addison–Wesley.

Clarke and McLean 1983 'Results from the Analysis of the Collins Building Second Floor and Plenum by the ESP System' *Appendix C8, IEA Annex 4 Final Report* Building Research Establishment, Garston.

Clarke 1985 *Energy Simulation in Building Design* Adam Hilger Ltd Bristol.

Clarke and McLean 1986 'ESP User's Manual: System 5, Release 3' *ABACUS Manual Series* University of Strathclyde.

Clarke 1986 'An Intelligent Approach to Building Energy Simulation' *Proc. CICA/BSRIA Conf. on Expert Systems* London.

Clarke 1987a 'Zone Solar Weighting Factors: Comparison of ESP Predictions with Three North American Models' *ABACUS Technical Report* University of Strathclyde.

Clarke 1987b 'The Future of Building Energy Modelling in the UK: The Energy Kernel System' *Report to the UK SERC* ABACUS, University of Strathclyde.

Bloomfield 1987 'A Proposal to Establish a UK Building Energy Modellers Club' Document in Preparation, Building Research Establishment, Garston.

Fabers 1980 *IEA Annex 1 Final Report* Oscar Fabers & Partners, St Albans.

Gicquel R and Cools C 1986 'Programme PASSYS: Status Report' Ecole des Mines, Sophia Antipolis, France.

Gough 1984 'ESP: Comparison with Monitored Direct Gain Tests Cells' *IEA Task 8 Report* Building Research Establishment, Garston.

Mackey C O and Wright L T 1944 'Periodic heat flow – homogeneous walls or roofs' *ASHVE Trans.* 50 V293.

MacRandal D 1986 'The Application of Intelligent Front Ends in Building Design' *Informatics Group Report Rutherford Appleton Laboratory, Didcot.*

Maver 1982 'The Impact of Computer Based Models on Design Decision–Making' *Rebuild* Wiley, pp95–141.

McLean and Clarke 1985 'Results of the Collins Building VAV Air Conditioning Simulation' *IEA Annex 10 Report* ABACUS, University of Strathclyde.

Mitchell W J 1977 *Computer Aided Design* Petrocelli/Charter, New York.

SERC/BRE 1987 'Various Publications on Model Validation' Building Research Establishment, Garston.

Stephenson D G and Mitalas G P 1967 'Room thermal response factors' *ASHVE Trans.* 73 V2019.

Tang D C 1984 Central Heating System Simulation *IEA Annex 10 Report* ABACUS, University of Strathclyde.

Williamson T 1984 'The Goodness of Fit Between ESP Predictions and Monitored Data From Two Australian Tests Houses' *ABACUS Technical Report* University of Strathclyde.