Interoperability-based Optimisation of Architectural Design

A thesis submitted by

Monjur Mourshed
BArch, Bangladesh University of Engineering and Technology, 1999

to the Department of Civil and Environmental Engineering, National University of Ireland, Cork in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

in Civil and Environmental Engineering
at the National University of Ireland, Cork, Ireland.

July 2006

Dedicated to my parents for tireless support and encouragement

Contents

Acknowledgements

I am ever so grateful to the persons below:

Dr. Denis Kelliher, my supervisor - I am greatly indebted to him for providing guidance throughout the development of this research. I thank him for his patience and encouragement that carried me on through difficult times, and most importantly for introducing me to the world of optimisation. I would also like to thank him for his advice on both my research and my academic career.

Dr. Marcus Keane, Co-Director of IRUSE - I am ever so grateful to him for his tireless support and guidance throughout and beyond this thesis. I thank him for his valuable and timely advice on building simulation and HVAC systems which have been extensively used in this thesis.

Prof. Philip O' Kane, Head of department - I am grateful to him for his encouragement and support.

Prof. Ger Kiely, **Dr. Eamon McKeogh** and **Dr. Brian Ó Gallachóir**, I would like to thank them for their encouragement and advice.

Fahmida Khandokar, my wife and a fellow graduate student at the department - I cannot thank her enough for the unconditional support she rendered during this period.

Wilfred Matipa, **Gearoid Lydon**, **Dr. Dominic O' Sullivan**, **James O' Donnell** and **Dr. Elmer Morrissey, -** I would like to thank my fellow graduate students at IRUSE for their genuine friendship and help.

Dr. Tasleem Shakur, Edgehill College, UK - I would like to thank him for his support and advice on various matters - academic and career.

Prof. Fuad Mallick, **Dr. Zainab Ali**, **Dr. Khandokar Shabbir Ahmed**, **Prof. Nizamuddin Ahmed** and **SM Najmul Imam**, Former colleagues, mentors and tutors - I would like to thank them for their continued support and encouragement.

Maria Murray-Carmichael, **Deidre Davenport**, **Ron Smith, Julie Cross, Andrei Chamchine** and **Don Fairhurst-** Friends and colleagues at the University of Central Lancashire, I would like to thank them for their encouragement and advice.

Finally, I would like to thank all the staff and graduate students in the Department of Civil and Environmental Engineering, NUIC.

Abstract

As a major contributor to the planetary greenhouse effect, construction industry needs to adopt sustainability at the core of its activities - to reverse or slow down the impacts of climate change. Increased collaboration among stakeholders along with analysis/performance based decision making is the way forward for enhanced sustainability. Emphasis is placed on the process of shared creation through multi-disciplinary collaboration, enabled by the implementation of IT (Information Technology) that acts as a platform to augment our ability to communicate. Developments in the Construction IT have been product oriented and aimed at solving particular domain problems usually with a narrow focus - further reducing the accessibility and interoperability of information over the lifecycle stages. Advances in the semantics based interoperable data standards, such as IFC (Industry Foundation Classes) offer significant advantage in removing such barriers to successful vertical and horizontal integration of software tools and process. The use of building simulation in architectural design requires specialist knowledge and a rich set of information about the proposed building which are not available to the design team at early stages. Standards based mapping of information for input processing of the simulation engines can act as an alternative to simplified tools supporting the exploratory nature of design. Detailed based input processing also restricts the use of simulation to occasional validation of solutions - even during detailed design stages. For a directed exploration of the solution space, numerical optimisation methods can be applied to enhance simulation assisted design. Successful application of optimisation methods pivots on the ability of the analysis and decision making components of the software to communicate with each other without the loss of data semantics.

To realise this potential, a process-oriented integrated framework based on the interoperability of information and software tools have been developed and implemented in this thesis. For horizontal integration of domain specific tools through intra-software messaging, ardML - an XML (eXtensible Markup Language) based schema has been developed which attempts to connect non-interoperable software tools. Multi-disciplinary environmental design of buildings has been chosen as the domain of discourse. The framework currently employs industry standard zonal building simulation as an analysis tool and gradient-based mathematical optimisation methods for informed decision making. Interoperability among tools, processes and information has been achieved through the implementation of IFC based data model. The modular nature of the object-oriented framework allows incorporation of existing and future tools. The applicability of the framework has been investigated in the early stages of architectural design, in particular the selection of form and orientation - considering the environmental aspects. The implementation of the framework at an ambiguous and exploratory stage of design reinforces its applicability in a wider industry context.

Abbreviations

ADF	**A**verage **D**aylight **F**actor
ADF Calc	**A**verage **D**aylight **F**actor **Calc**ulator
AEC	**A**rchitecture, **E**ngineering and **C**onstruction
AEC/FM	**A**rchitecture, **E**ngineering, **C**onstruction and **F**acilities **M**anagement
API	**A**pplication **P**rogramming **I**nterface
ArDOT	**Ar**chitectural **D**esign **O**ptimisation **T**ool
BIM	**B**uilding **I**nformation **M**odelling
BPM	**B**uilding **P**roduct **M**odel
CAD	**C**omputer **A**ided **D**esign
CAx	**C**omputer **A**ided **x**=Design/Engineering/Manufacturing
CFD	**C**omputational **F**luid **D**ynamics
DHTML	**D**ynamic **HTML**
DGN	Microstation **D**esi**gn** file
DOE	**D**esign **O**f **E**xperiments/**D**epartment **o**f **E**nergy, US (Context dependent)
DTD	**D**ocument **T**ype **D**efinition
DWG	AutoCAD **D**ra**w**in**g** format
DXF	**D**rawing e**x**change **F**ormat
EPW	**E**nergy**P**lus **W**eather File
FEA	**F**inite **E**lement **A**nalysis
GUI	**G**raphical **U**ser **I**nterface
GUID	**G**lobally **U**nique **Id**entifier
HTTP	**H**yper**t**ext **T**ransfer **P**rotocol
IAI	**I**nternational **A**lliance for **I**nteroperability
IDF	EnergyPlus **I**nput **D**ata **F**ile
IFC	**I**ndustry **F**oundation **C**lasses
IGES	**I**nitial **G**raphic **E**xchange **S**pecification
ISO	**I**nternational **O**rganisation for **S**tandardisation
IT	**I**nformation **T**echnology
JAXB	**J**ava **A**rchitecture for **X**ML **B**inding
MCAD	**M**echanical **CAD**
MIS	**M**anagement **I**nformation **S**ystems
NIAM	**N**atural Language **I**nformation **A**nalysis **M**ethod

PDES	**P**roduct **D**ata **E**xchange **S**tandard
PDM	**P**roduct **D**ata **M**anagement
PMD	**P**roduct **M**odel **D**atabase
RELAX	**RE**gular **LA**nguage description for **X**ML
RELAX NG	Schema language for XML based on RELAX and TREX
RSA	**R**esponse **S**urface **A**pproximation
SDAI	**S**tandard **D**ata **A**ccess **I**nterface
SET	**S**tandard d'**E**change et de **T**ransfert
SGML	**S**tandard **G**eneralised **M**arkup **L**anguage
SME	**S**mall and **M**edium **E**nterprises
SPF	**S**TEP **P**hysical **F**ile
STEP	**ST**andard for **E**xchange of **P**roduct model data
TMY	**T**ypical **M**eteorological **Y**ear
TREX	**T**ree **R**egular **E**xpressions for **X**ML
TRY	**T**est **R**eference **Y**ear
UI	**U**ser **I**nterface
VDA-FS	**V**erband **D**er **A**utomobilindustrie **F**lachen **S**chnittstelle
XML	e**x**tensible **M**arkup **L**anguage
XSL	E**x**tensible **S**tylesheet **L**anguage
XSLT	**XSL** **T**ransformations

List of figures

List of tables

Chapter 1
Introduction and research objectives

Chapter 1 begins with a discussion on the fragmented nature of the Architecture, Engineering and Construction *(AEC) industry. The role of* Information Technology *(IT) in enhancing collaboration among stakeholders involved in AEC projects is described. Issues related to interoperability in architectural design process are examined and a brief introduction to the interoperability standards is given. Architectural design as an optimisation activity is introduced as well as the general concepts of optimisation. Integration of building simulation in design is discussed with reference to performance based design. To ensure sustainability in AEC projects, interoperability-based integrated framework has been proposed; incorporating performance analysis and optimisation to guide design exploration. The motivation and objectives for this research are described and the chapter concludes with an outline of the thesis.*

Background

Since the development of the first graphic display system by Sutherland (Sutherland 1963) that uses drawing as a novel communication medium, sophisticated computer systems ranging from *Computer Aided Design* (CAD) to Simulation and *Management Information System* (MIS) have been developed for the AEC industry over the years. Despite the opportunities offered by these techniques and tools their potential has not been realised fully. Fragmentation in the industry (Latham 1994; Egan 1998; Austin *et al.* 2002) and the lack of integration among software tools and processes (Austin *et al.* 1994; Fischer *et al.* 1998) are considered to be the primary reasons. Fragmentation is inevitable to some extent due to the fact that the AEC industry is an increasingly complex network of *Small and Medium Enterprises* (SME) with a diverse workforce(Egan 1998). A number of approaches and strategies have been implemented to overcome industry fragmentation, but without much success. According to a recent report by the *Commonwealth Scientific and Industrial Research Organisation* (CSIRO) (Tucker *et al.* 2001), "*most of these approaches have tried to focus on elements linked to time, quality and/or cost, and research has noted that 85% of commonly associated problems are process related, and not product related*". The term *process* refers to the ways buildings are conceived; a significant part of this *process* is the aspects related to collaboration among stakeholders.

Integration among the fragmented processes through the use of IT is generally concerned with the development of information systems, which is a multifaceted process, embodying elements of rational design together with varying degrees of organisational negotiation. It is often observed that the methods and tools used to focus development efforts often encourage developers or tools to over-emphasise the rational design element and to overlook the organisational context and stakeholder perceptions against which the development takes place (King 1997). This results in the *lack of usability*, deterring widespread adoption by a fragmented industry like AEC. Recent conferences on *IT in Construction* have raised concerns over the lack of appreciation of the process aspect of IT implementation.

Advances in building simulation (see Augenbroe 2002; Mahdavi 2004; Malkawi 2004) and engineering analysis, in particular in *Finite Element Method* (FEM) and *Finite Difference Method* (FDM) (Kelliher 1999) implies that a significant part of architectural design problems can now be formulated mathematically. Simulation tools have applications beyond environmental performance or life cycle assessment; e.g. pedestrian movements (Jian *et al.* 2005), layout and circulations (Radford and Gero 1988; Liggett 2000), etc. Increased processing power of computers now allows architects/designers to take advantage of simulations in decision making, which can be taken further by deploying mathematical optimisation techniques (Mourshed *et al.* 2003a). Mathematical optimisation, a specialised area of operations research and engineering design has not been deployed widely in the domain of architectural design. Reasons include, but are not limited to data integration and difficulties in problem formulation - although the benefits of deploying optimisation techniques in the design process have been observed (Gero and Kazakov 1998; Al-Homoud 2000; Wright *et al.* 2002; Mourshed *et al.* 2003b; Wetter and Polak 2005).

This thesis looks at architectural design holistically on the basis that computation is an integral part of the design process where data integration plays the central role. The term *computation* refers to the *building simulation/analysis* and *optimisation* techniques for informed decision making in design. Data integration is concerned with the interoperability among software tools and stakeholders in a collaborative environment.

Interoperability issues in architectural design process

Architectural design in its broadest terms is an iterative process where cognitive models of design intent and content play a vital role. Researchers and practitioners have long tried to introduce automation into the design process, not only to improve the quality and consistency of design but also to eliminate guesswork from the parts of the process that can be mathematically formulated. The challenge has been described by Clayton *et al.* (Clayton *et al.* 1996) as the introduction of automation in a way that makes design faster and more effective; stimulates rather than hinders designers' creativity and freedom in reaching a solution. The fundamental issue in automation in design is the need for a common language or representation of the domain information that is understood by multi-disciplinary stakeholders in the project. Referred to as data interoperability, this representation is theoretically intended to represent information in all lifecycle stages of a building.

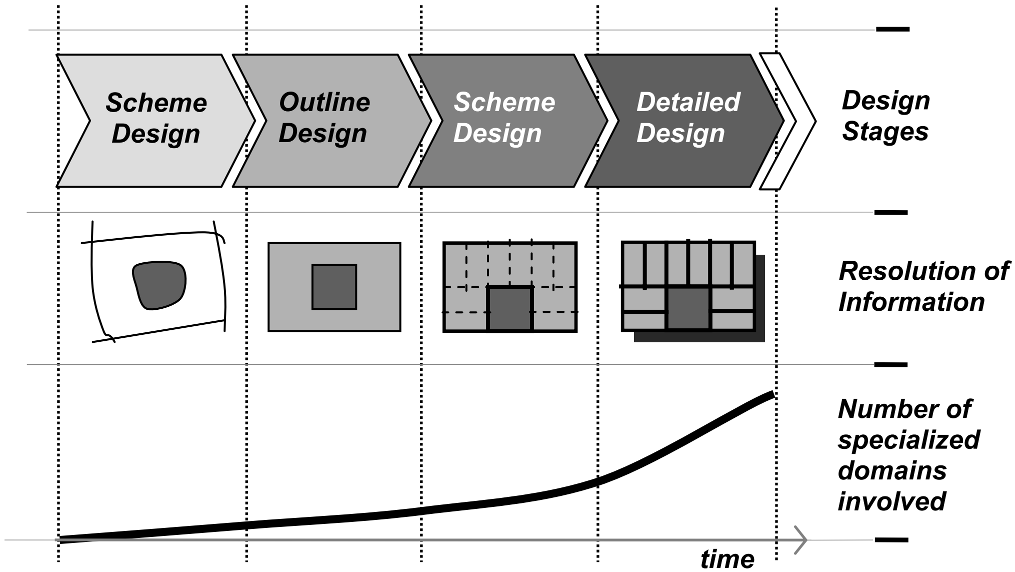


Figure 1.1: Resolution of information at different design stages.

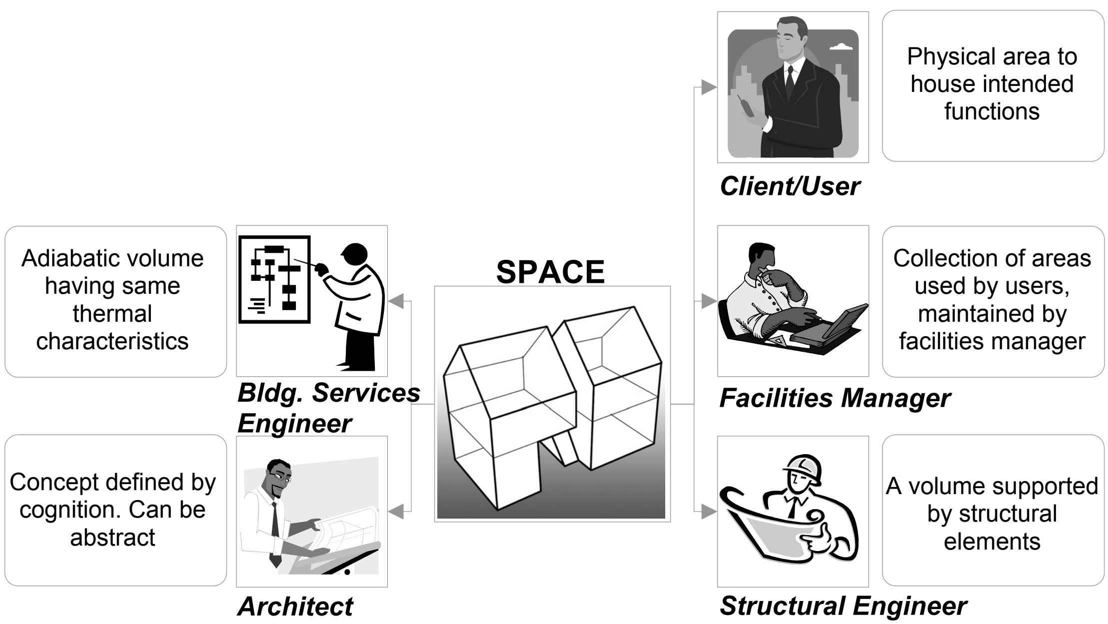


Figure 1.2: Varying representations of an AEC concept (e.g. space) by stakeholders in an AEC project.

Achieving interoperability in the multi-disciplinary, distributed and fragmented AEC industry depends on flexibility and strength of storing, extracting and sharing of information. Building information varies from stage to stage in its resolution; i.e. how detailed a particular building element is at that stage. For example, during feasibility and site planning a building is referred to as a volume and/or floor area; form and envelope of the building may not be defined at this stage. As the design progresses, the conceptual mass is articulated with elements providing the building with enclosure and structure (see Figure 1.1). To enable digital collaboration, information models need to capture varying *resolution* of information during lifecycle stages.

Architectural design is exploratory; graphic representations of the building elements provide links with the symbolic representations used to explore the solution space cognitively. Associating design information with geometric entities has long been a chosen method for information modelling in the AEC industry. Because of the multi-disciplinary nature of stakeholders in a building project, the views and representations taken by the team participants also vary. Varying representations of an AEC concept by concerned professionals are illustrated in Figure 1.2 where space is understood and represented differently by individual groups of professionals. Information models in the AEC industry also need to cater for such varying *representations* for effective multi-disciplinary collaboration.

Data exchange

Efforts at establishing neutral data exchange standards include *Drawing Exchange Format* (DXF), *Initial Graphic Exchange Specification* (IGES), *Standard for Exchange of Product Data* (STEP), and *Industry Foundation Classes* (IFC). The extension of the capabilities of EXPRESS based information modelling is underway, the recent example being ifcXML, part 28 representation of the IFC schema.

Understanding the limitations of data exchange methodologies in existing CAD/CAM systems, STEP has been developed by *International Organisation for Standardisation* (ISO) (ISO 1994a). STEP, also known as ISO 10303 is an international product data standard to provide a complete, unambiguous, computer-interpretable definition of the physical and functional characteristics of a product throughout its life cycle. STEP is targeted at the exchange of data describing a product between *Computer Aided Design/Engineering/Manufacturing* (CAx) and *Product Data Management* (PDM) systems; and storage of data in repositories throughout the lifecycle (Moeller 2000). Fundamental to the STEP approach are the layers segregating the logical structure of the information from the format in which it is carried. STEP development is based around various part models known as *Application Protocols* (AP), with the expectation that the APs would later be re-established as larger domain specific models. APs are represented using EXPRESS (ISO 1994b), a data modelling language part of ISO 10303 specification. EXPRESS is able to represent knowledge embedded in a product model in an implementation independent manner.

IFC is an information standard for the AEC/FM industry developed by *International Alliance for Interoperability* (IAI) with a vision to improve communication, productivity, delivery time, cost and quality throughout the whole building life cycle by providing a universal basis for process improvement and information sharing in the industry (IAI 2005a). IFC is based on the idea of digital representation of objects that occur in a constructed facility. IFC specifications represent a data structure supporting a digital project/product model for use in data exchange and sharing across applications and domains. EXPRESS has been adopted as the data definition language for IFC model. Both STEP and IFC data can be encoded to *STEP Physical File* (SPF) according to Part 21 (ISO 2002) specification of ISO-STEP. Data can also be shared and accessed using *Standard Data Access Interface* (SDAI) (ISO 1998b) repository. Thus, the IFC data model corresponds with the STEP standard and consequently contributes to the evolution that permits the exchange of building data between different programs. Further discussions on existing and evolving interoperability standards are carried out in Chapter 3.

Architectural design as an optimisation activity

Architectural design can be referred to as an optimisation activity as it aims to improve the design so as to achieve the *best* way of satisfying the design requirements within the available means. The selection of the *best* or the *optimum* design solution from a range of available means is known as *design optimisation*. Traditionally, optimisation has been a recognised activity in engineering design, mainly due to the fact that most engineering problems can be described mathematically. A considerable part of architectural design is technological, i.e. design objectives and boundaries can be formulised as mathematical problems.

Based on the comparison between engineering design and structural optimisation described by Kelliher (Kelliher 1999), a comparison between architectural design and formal optimisation is drawn up in Figure 1.3. Architectural design depends on individual's training, experience and cultural influences rather than formal analyses of the systems involved. Rules of thumb usually complement designer's cognitive ability in reaching a solution. This combination of cognition and educated guesses does not necessarily lead to the intended optimum as the solution space is complex involving a large number of variables. However, advances in building simulation have now made it possible to eliminate guesswork from parts of the design process, in particular decision making related to energy efficiency. Proposed alternatives can now be evaluated to ascertain building performance based on predefined criteria. To understand the effect of interdependent design variables in the overall performance, effective visualisation of outputs is needed.

Optimisation comes as a logical choice for simulation-assisted *performance based designs* which aims to *find the optimum* solution satisfying *predefined objective*(*s*) with/without user intervention. Objectives can range from single to multiple; e.g. reduction in operation/capital cost or maximisation of daylighting or a combination of both. A close look at Figure 1.3 reveals that although both the conventional and optimal design process are conceptually similar, the systematic improvement of the design in optimisation is rather directed by the evaluation of the imposed criteria, which are usually estimated from the current design responses. Optimisation is essentially about automating the design process to find the best *outcome* (design), hence referred to as *synthesis* than design (Kelliher 1999).

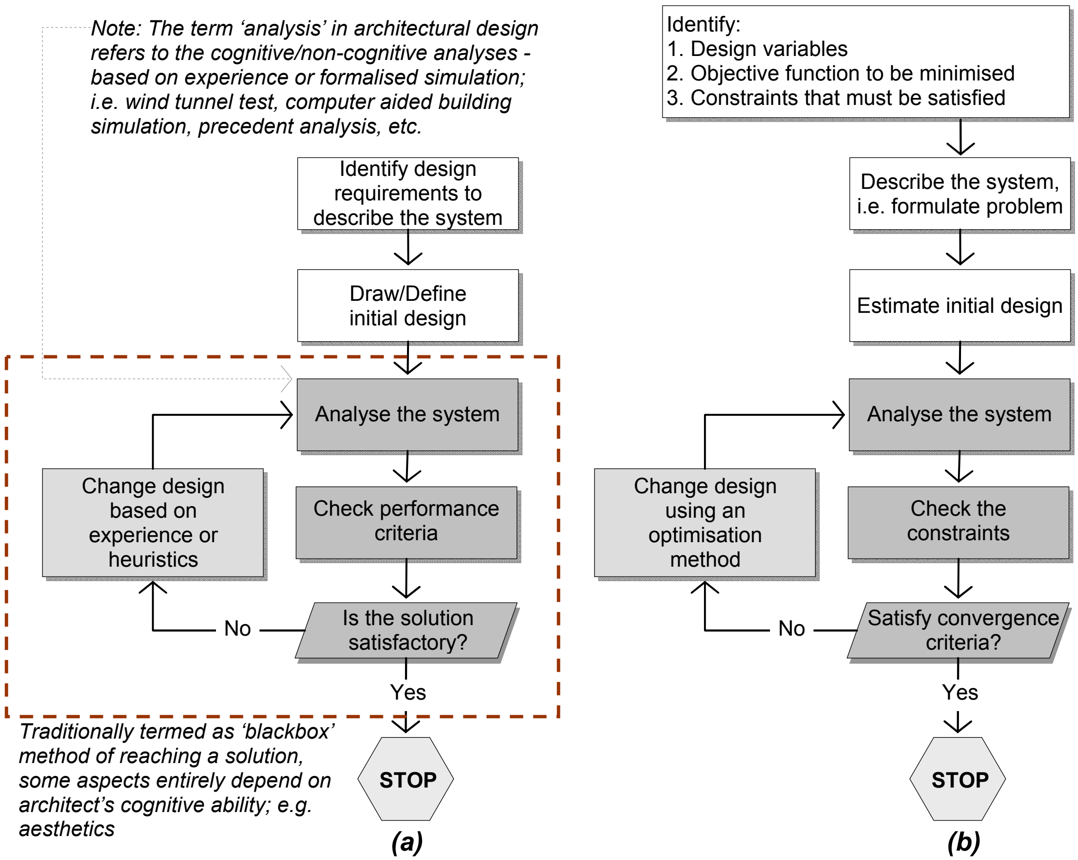


Figure 1.3: Flow chart depicting (a) the conventional architectural design process and (b) the optimal design process.

General concepts of optimisation

All optimisation problems have the basic mathematical formulation, which is expressed as:

Find the vector x such that it

Minimises (or maximises): $f(\mathrm{x})$ (1.1)

Subject to: $g_j(\mathrm{x}) \leq 0 \qquad j = 1,2,\ldots,m$ (1.2)

$h_k(\mathrm{x}) = 0 \qquad j = 1,2,\ldots,m$ (1.3)

$s_i^{lower} \leq s_i \leq s_i^{upper} \qquad i = 1,2,\ldots,n$ (1.4) where: x is design vector $(x_1, x_2, ..., x_n)^T$, n being the number of variables,

$g_j(\mathrm{x})$, $h_k(\mathrm{x})$ are the behaviour constraints, and

s_i^{lower}, s_i^{upper} are the lower and upper bounds of design variable s_i.

All the design variables: $x_1, x_2, ..., x_n$, are assembled into the vector $\mathrm{x} = (x_1, x_2, ..., x_n)^T$ belonging to a subset X of the n-dimensional real space $\Re^n$, that is $\mathrm{x} \in X \subseteq \Re^n$. The choice of $\Re^n$ is made because the vast majority of the design problems have variables that are continuous. Equation (1.1) represents the objective function; Equation (1.2) and Equation (1.3) represent functional relations among the design variables, otherwise known as inequality and equality constraints respectively. Equation (1.4) represents lower and upper bounds on design variables.

Often, the development of the design model indicates more than one objective function that needs to be minimised, e.g. the design of a building may need to minimise capital/operating cost, while maximising the availability of daylight. These objectives may be competing; i.e. reducing the capital cost of glazing and operating cost (e.g. energy) will require glazing and daylight availability to be reduced. More than one function may be an objective; in this case the optimisation model will have a vector objective rather than a scalar one. Optimisation problems with multiple objectives are termed as multi-objective or multi-criteria optimisation. For a vector objective c, the minimisation formulation of the multi-criteria optimisation problem is expressed as:

Minimise: $\mathrm{c}(\mathrm{x})$ (1.5)

Subject to: $g_j(\mathrm{x}) \leq 0 \qquad j = 1,2,...,m$ (1.6)

$h_k(\mathrm{x}) = 0 \qquad j = 1,2,...,m$ (1.7)

$s_i^{lower} \leq s_i \leq s_i^{upper} \qquad i = 1,2,...,n$ (1.8)

where: c is the vector of I real-valued criteria c_i.

Several methods exist for converting the multi-criteria formulation into a scalar objective to solve optimisation problem using single objective optimisation methods. The scalar objective has the form $f(\mathrm{c}, \mathrm{M})$, where M is a vector of preference parameters that can be adjusted to tune the scalarisation to the designer's subjective preferences. The simplest scalar substitute objective is obtained by assigning subjective weights to each objective and summing up all objectives multiplied by their corresponding weights. For min $c_1(\mathrm{x})$ and max $c_2(\mathrm{x})$, the problem may be formulated as:

$$\min f(\mathrm{x}) = w_1 c_1(\mathrm{x}) + w_2 [c_2(\mathrm{x})]^{-1} \tag{1.9}$$

A generalisation of the function is:

$$f = \sum_i f_1(w_i) f_2(c_i, \mathrm{m}_i) \tag{1.10}$$

where: the scalars w_i and vectors m_i are preference parameters.

This approach involves subjective information and can be misleading concerning the nature of the optimum design. Importance is placed on tracing the effect of subjective preferences on the decisions suggested by the optimal solution obtained after solving the substitute problem. Precise design preferences are rarely known *a priori*, so preference values are adjusted gradually and trade-offs become more evident with repeated solutions of the substitute problem with different preference parameter values (Papalambros and Wilde 2000). A common preference is to reduce at least one criterion without increasing any of the others. Using this assumption the set of solutions for consideration can be reduced to a subset of the attainable set, termed the *Pareto set*, which consists of *Pareto optimal points*. Further discussions on Pareto optimality and its application in informed decision making are carried out in Chapter 4.

Building simulation

Drawing on resources from diverse disciplines such as physics, mathematics, computer science, materials science and physiology, building simulation intends to predict the behaviour of a building during various lifecycle stages. Benefits offered by these tools and concepts include but not limited to better return on investment, reduction in waste, increased energy efficiency and occupant comfort. Applications of building simulation range from *building physics*-based thermal performance simulation to *time*-based construction process simulation. Albeit only one of the various building simulation tools, building energy simulation has the potential to be the most influential on sustainable building design. Building energy simulation is discussed, as it relates to this thesis; but all simulation tools share the same principles and use as a decision aide in the design process. Key aspects dominating the evolution of building simulation are:

- increasing the efficiency and accuracy of simulation engines in predicting performance, and
- integration of tools and expertise in the overall building process.

Use of simulation in design

Developed mostly in research organisations, building energy simulation programs focus on modelling and simulation and not on integration with the design process. Enormous amount of input processing is required even to simulate a small subset of the domain. Complicated processes to accomplish tasks make their use limited to the occasional validation of the proposed idea rather than to assist in the design development. Specialist knowledge and expertise are often required, hindering the use of simulation at the early design stages. Extending the capabilities of simulation software can play a vital role in early stages of design, in which most of the decisions relating to energy-efficiency of the building are made. Poor decisions made at this stage are often irreversible and expensive to reverse at later stages (Mourshed *et al.* 2003b).

Figure 1.4 reinforces the need to incorporate simulation as early as possible in the design process in the context of building energy simulation. Figure 1.4a shows the involvement of the professionals concerned at the different stages of design; architects and building services engineers are specifically considered here. Figure 1.4b shows the impact of the design efforts in potential energy savings. The overarching decisions, such as the form of the building, are made at a very early stage and considered to be vital for enhancing overall energy efficiency of the building. Availability of energy simulation tools at different lifecycle stages of a building are shown in Figure 1.4c.

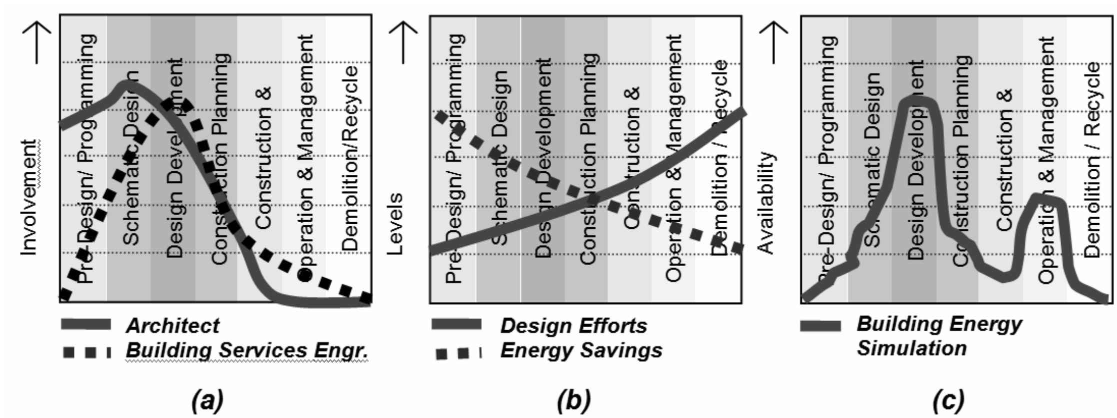


Figure 1.4: (a) Professional involvement, (b) design efforts vs. energy savings and (c) contemporary use of building energy simulation in different lifecycle stages.

Architectural design is exploratory by nature; the better the search in solution space, the better the outcome. Exploring design at conceptual stages considering multi-disciplinary aspects constitutes the work process of an architect. Particular tasks such as form finding do not depend on environmental performance only; efficiency of space layout, aesthetics, circulation, etc. also play important roles. Delegating performance evaluation to domain experts at this stage will generally hinder effective exploration of the solution space and the cognitive process. It is argued that architects' access to simulation in the form of advanced decision making platform is essential for integration of simulation in design.

Integration of building simulation in design

Integrating simulation tools in the design process has been of utmost importance in the AEC community (Papamichael *et al.* 1997; McElroy and Clark 1999; Augenbroe 2002; Mourshed *et al.* 2003b; Malkawi *et al.* 2005). Efforts at integrating building simulation in design can be broadly classified as:

- User Interface developments using principles of decision sciences,
- Data integration and tool interoperability, and

- Re-engineering the process of design.

The separation of the *simulation engine* from the *User Interface* (UI) in the development of EnergyPlus (Crawley *et al.* 2001), building on the development experiences from two existing programs: DOE-2 and *Building Loads Analysis and System Thermodynamics* (BLAST), paved the way for better integration of energy simulation in design. Domain and task specific UIs can now be developed by domain experts considering the specific decision-making requirements of the user groups concerned. The requirements of a tool vary considerably at different stages of design. For example, building services engineers are more concerned with the description and sizing of systems, while architects are concerned more with the form and envelope of the building. Increasing number of specialised interfaces is now available to accomplish tasks to varying degrees of complexity. To enhance the capabilities of the specialised interfaces, investigations have been carried out on the applications of optimisation techniques (Al-Homoud 1997a; Wright and Loosemore 2001; Mourshed *et al.* 2003b; Choudhary *et al.* 2005; Wang *et al.* 2005b; Wetter and Polak 2005), coupling building simulation and decision making in environmental design of buildings.

Data integration and tool interoperability efforts are directed at two levels: (a) *integration of multi-domain simulation tools* and (b) *interoperability among industry wide software applications*. Examples of integrated simulation environments facilitating access to simulation engines from multiple domains through a common interface and data structure are: *Building Design Advisor* (BDA) (Papamichael *et al.* 1997), *Computer Models for Building Industry in Europe* (COMBINE) (Augenbroe and Laret 1989) and *Simulation Environment for Modelling Performance* (SEMPER) (Lam *et al.* 2004). The next development in integration was aimed at achieving interoperability among industry wide software applications, coordinated through the implementation of *Building Product Models* (BPM) such as STEP and IFC. A detailed discussion on interoperability and ongoing and previous efforts is carried out in Chapter 3. The use of IFC in interoperable building simulation is discussed by Bazjanac (Bazjanac 2004) along with various methods of implementation using IFC HVAC schema.

Re-engineering of the design process is emphasised by researchers and expertise in building simulation is considered indispensable. Two methods of design process re-engineering through increased use of simulation are proposed: *inclusion of specialists* (i.e., Building Services Engineer) at the earliest in the design process, and the development of simplified *designer-friendly* simulation tools. Results from the research (McElroy and Clark 1999) at ESRU, the University of Strathclyde on the inclusion of simulation experts at the earliest in the design process suggest significant advantages in rendering energy efficiency in the proposed environmental design. Although the input from experts in the form of specialist knowledge is considered as an option, delegating design exploration to professionals other than architects in the conceptual stages will inevitably affect design outcome. The development of the designer-friendly simulation tools is based on a simplified representation of the problem domain. Their use is often discouraged by researchers on the basis that they do not provide accurate results (Augenbroe 2002) and in most cases do not cater for the multi-domain problems. Both the arguments have strong followings although the need for designer-friendly tools or the expert knowledge at the earliest should not be understated.

Green building design concepts and the ERI

Further to the developments in interoperability based integration, a framework titled *Green Building Project* has been proposed by Keane and Kelliher (Keane and Kelliher 2001), considering the role of simulation and the advancements in sensor and visualisation technologies. The project proposed to specify, design and construct an example building - Environmental Research Institute (ERI) at the National University of Ireland, Cork, to underpin interdisciplinary research activities from both the engineering and scientific communities relating to both the internal and external environment. The framework aimed at developing an integrated web-based software environment to provide collaborative support for the design and construction of ERI and future green buildings.

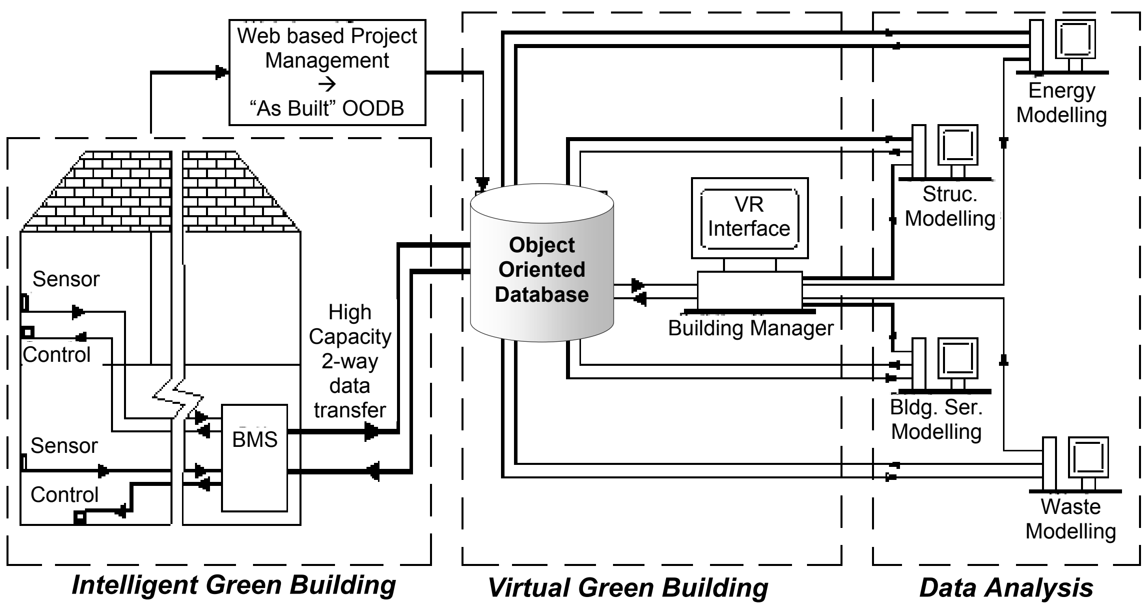


Figure 1.5: Schematic of ERI demonstrating the IT systems and their relationships with the building's OODB, after (Keane and Kelliher 2001).

Central to the design and construction of ERI was the development and implementation of the proposed collaborative framework with the following objectives:

- Support the project manager or the architect at the conceptual and detailed design stage of the project,
- Support the project (or construction) manager during the construction phase of the project,

- Provide the building manager with a software system which will allow real time acquisition of performance data from the *Building Management System* (BMS) and,
- Allow the building manager to initiate changes to the building control systems.

At the heart of the framework, there is an OODB (Object Oriented Database) , otherwise known as *Product Model Database* (PMD) which enables storing and retrieval of IFC objects. The database is also capable of storing information about the building during all lifecycle stages, from conception to operation/demolition. The information may come from drawings, specification and live data from the installed sensors during operation stage. To enable virtual monitoring of the building, the functionality of 3D geometry in the product model database is to be enhanced allowing stakeholders of the building to browse relevant information using visualisation techniques. Figure 1.5 depicts the schematic of the ERI and the associated tools. The information flow between the building sensors, the BMS, the buildings OODB and the visualisation software are also detailed.

Motivation behind this research

Participation, collaboration and integration have been promoted by professionals, researchers, policy makers as being instrumental in achieving sustainability (Roseland 2000). In the globalised economy, the design and construction of buildings has become a distributed activity connecting stakeholders from different backgrounds - professional, political and cultural. The aspects of sustainability affecting built environment ranges from social to environmental and economical to political. Fuel poverty and eminent replenishment of fossil fuels call for increased energy efficiency in the built environment (Goodacre *et al.* 2002). The debate on sustainable development as a science remains open as the concept encompasses a broad spectrum of fields of knowledge. Hjorth and Bagheri (Hjorth and Bagheri 2006) argued that traditional fragmented and mechanistic science is unable to cope with the issues of sustainability as they are often related to complex, self-organising systems.

Whether it is promoting Agenda 21[1] (UNCED 1992) or implementing a part of it, the existence of collaboration and stakeholder participation is vital. The goal, if translated into the language of design, is the ability to exchange data, collaborate to achieve sustainable design, provide a scientific basis for making design decisions and optimise the use of resources. Better integration of advanced analysis tools such as building simulation within an interoperable and integrated platform fostering distributed partnerships can be considered an objective. Resolution of conflicts across stakeholders

[1] Agenda 21 is a comprehensive plan of action to be taken globally, nationally and locally by organizations of the United Nations System, Governments, and Major Groups in every area in which human impacts on the environment. Source: http://www.un.org/esa/sustdev/

and performing better risk and uncertainty analyses of performance predictions throughout all life cycle stages of the building, as described in (de Wit 2001) are bi-products of such platform.

Architectural design often involves minimisation or maximisation of certain design criteria based on predefined objective(s). For example, reduction in construction/operating cost, maximisation of daylight availability in a certain space, etc. Computer aided building simulation tools can offer significant leverage in decision making while minimising or maximising certain criteria. It is shown in Figure 1.3 that the process of heuristics/experience based design (Figure 1.3a) can be improved by introducing formal optimisation methods in the process (Figure 1.3b). Unlike conventional architectural design process, this thesis proposes simulation oriented performance based design, where design iterations are guided by optimisation algorithms. It helps designers to search design/solution space effectively, realising the full potential of simulation rather than occasional validation of a solution through single-pass simulation or by using rules of thumb.

The recent move towards BIM (Building Information Modelling) based integration by software vendors (e.g. Autodesk, Graphisoft, Microstation) recognises the need for an integrated platform and collaboration. The way these narrow-focused integrations have been implemented does not share the collective vision of the community for an industry-wide collaboration based on vendor independent standards. Vendors promoting their own platforms can further complicate collaboration between non-proprietary systems, although they provided some mechanisms to exchange data using standardised representations such as STEP in the mechanical industry and IFC in the architectural industry. This thesis sees integration as an activity based on internationally agreed standards for modelling AEC information.

Research objectives

As part of the *Green Building Project* described in section 0, this thesis was primarily aimed at integrating building simulation during architectural design stages for enhanced sustainability. The objectives of this thesis can be summarised as:

1. To understand *the integration of building simulation in the early stages of design* for informed decision making, considering the aspects of interoperability and collaborative design. Of particular interest was *modelling for simulation* as the details about the building are not fully known at conceptual stages.

2. To adopt *formal optimisation techniques as part of the architectural design process* to realise the potential offered by simulation through directed exploratory search of the solution space.

3. To gain an understanding of the contemporary developments in *BPM* and to determine their effectiveness in modelling building information at the early stages of design. Emphasis was given on how BPMs cater for the dynamic information modelling requirements for architectural design; i.e. task/process specification.

4. Development of an *interoperability-based software environment* to test the ideas in 1, 2 and 3.
5. To test this software environment in solving environmental design problems using the principles of performance-based design.

Structure of the thesis

This thesis specifically deals with building simulation, interoperability and mathematical optimisation; reviews of existing and ongoing efforts in each of the three areas are given in Chapter 2, 3 and 4 respectively.

Considering the aspects of global climate change and the need for a sustainable built environment, Chapter 2 begins by arguing the case for the integration of multi-domain performance assessments in the architectural design process. Concepts of building simulation are introduced; applications in simulation-based decision making are discussed. The barriers to successful integration of building simulation in the design process are elaborated. The case for the incorporation of mathematical optimisation techniques to better integrate simulation tools in design is made. Ongoing efforts at integration of simulation in design using various approaches are briefly reviewed.

Chapter 3 introduces BIM (Building Information Modelling) and the concepts of interoperability in the AEC industry. Historical overview of interoperability standards is given. Major standards based on EXPRESS modelling language such as STEP and IFC are described and analysed for adoption in an interoperability-based framework. Implementation issues related to building information modelling are discussed. Recent developments in BIM such as XML-based technologies are introduced. The role of EXPRESS and XML based standards in AEC information modelling and the way they can coexist in semantics-based information world are analysed. A brief overview of previous efforts in the development of interoperability-based integrated framework is given.

Chapter 4 is devoted to the study of optimisation methods and their applications in design, in particular design of buildings. Classifications of optimisation methods are discussed including reviews of the principles of the algorithms implemented in this thesis. Single and multiple criteria approaches to optimisation are introduced. Pareto optimality for informed decision making in multi-criteria optimisation problems is described. A review of the applications of optimisation methods in architectural design such as automated space layout and environmental design of buildings is provided.

Chapter 5 describes the methodology used to develop the framework by considering the aspects of building simulation, interoperability and mathematical optimisation, introduced in chapters 2, 3, and 4. Optimisation of environmental design of buildings during conceptual stage is introduced as the design activity for implementation. The rationale behind the selection of domain and simulation engines is described.

In Chapter 6, results from the implementation of the framework in optimisation of environmental design of buildings are described and analysed. The role of pre-design climate analysis in the selection of building form and envelope is discussed. Some selected design problems related to energy efficiency at early architectural design stage are tested with the implemented framework in two locations with distinct climate characteristics. Rationale behind the selection of locations is discussed along with their climate characteristics. Climate analysis is preceded by the methodology of analysis of climates. Discussions of optimisation results focus on the effectiveness of the optimisation process in reaching optimum. Convergence to optimum from different starting points using a single zone building is analysed and discussed. It is shown that the use of rules of thumb in constraint-based problem formulation offers advantage in design exploration at early stages. Study case involving multi-criteria optimisation using Pareto optimality is presented.

Chapter 7 outlines the conclusions and presents a short discussion and future work.

References

Al-Homoud, M.S. (1997). Optimum thermal design of air-conditioned residential buildings. *Building and Environment*, **32**(3), 203-210.

Al-Homoud, M.S. (2000). Computer-aided building energy analysis techniques. *Building and Environment*, **36**(4), 421-433.

Augenbroe, G. (2002). Trends in building simulation. *Building and Environment*, **37**(8-9), 891-902.

Augenbroe, G. and Laret, L. (1989). *COMBINE pilot study report*. CEC-JOULE. Brussels: European Commission.

Austin, S., Newton, A., Steele, J. and Waskett, P. (2002). Modelling and managing project complexity. *International Journal of Project Management*, **20**(3), 191-198.

Austin, S.A., Baldwin, A.N. and Newton, A.J. (1994). Manipulating the flow of design information to improve the programming of building design. *Construction Management and Economics*, **12**(5), 445-455.

Bazjanac, V. (2004). Building energy performance simulation as part of interoperable software environments. *Building and Environment*, **39**(8), 879-883.

Choudhary, R., Malkawi, A. and Papalambros, P.Y. (2005). Analytic target cascading in simulation-based building design. *Automation in Construction*, **14**(4), 551-568.

Clayton, M.J., Kunz, J.C. and Fischer, M.A. (1996). Rapid conceptual design evaluation using a virtual product model. *Engineering Applications of Artificial Intelligence*, **9**(4), 439-451.

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl, W.F., Huang, Y.J., Pedersen, C.O., Strand, R.K., Liesen, R.J., Fisher, D.E., Witte, M.J. and Glazer, J. (2001). EnergyPlus: creating a new-generation building energy simulation program. *Energy and Buildings*, **33**(4), 319-331.

de Wit, S. (2001). *Uncertainty in predictions of thermal comfort in buildings*. PhD Thesis. Delft: Technische Universiteit Delft.

Egan, J. (1998). *Rethinking Construction*. London: DETR.

Fischer, M.A., Waugh, L.M. and Axworthy, A. (1998). IT support of single project, multi-project and industry-wide integration. *Computers in Industry*, **35**(1), 31-45.

Gero, J.S. and Kazakov, V.A. (1998). Evolving design genes in space layout planning problems. *Artificial Intelligence in Engineering*, **12**(3), 163-176.

Goodacre, C., Sharples, S. and Smith, P. (2002). Integrating energy efficiency with the social agenda in sustainability. *Energy and Buildings*, **34**(1), 53-61.

Hjorth, P. and Bagheri, A. (2006). Navigating towards sustainable development: A system dynamics approach. *Futures*, **38**(1), 74-92.

IAI (2005). *IAI international: mission, vision and goal*. [Online]. Available from: <http://cig.bre.co.uk/iai_uk/iai/page3.htm>

ISO (1994a). *Industrial automation systems and integration - Product data representation and exchange - Part 1: Description methods: Overview and fundamental principles*. ISO 10303-1:1994. Geneva: ISO.

ISO (1994b). *Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual*. ISO 10303-11:1994. Geneva: ISO.

ISO (1998). *Industrial automation systems and integration - Product data representation and exchange - Part 22: Implementation methods: Standard data access interface*. ISO 10303-22:1998. Geneva, Switzerland: ISO.

ISO (2002). *Industrial automation systems and integration - Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure*. ISO 10303-21:2002. Geneva: ISO.

Jian, L., Lizhong, Y. and Daoliang, Z. (2005). Simulation of bi-direction pedestrian movement in corridor. *Physica A: Statistical Mechanics and its Applications*, **354**, 619-628.

Keane, M. and Kelliher, D. (2001). Green building design management. *International Journal of Architectural Management, Practice and Research*, **16**, 263-271.

Kelliher, D. (1999). *An Adaptive Finite Element Method for Optimal Shape Design of Elastic Continua*. PhD Thesis. Cork: National University of Ireland, Cork.

King, S. (1997). Tool support for systems emergence: A multimedia CASE tool. *Information and Software Technology*, **39**(5), 323-330.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004). SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. *Automation in Construction*, **13**(5), 651-663.

Latham, M. (1994). *Constructing the Team: Final Report to the Government/Industry Review of Procurement and Contractual Arrangements in the United Kingdom Construction Industry.* London: HMSO.

Liggett, R.S. (2000). Automated facilities layout: past, present and future. *Automation in Construction*, **9**(2), 197-215.

Mahdavi, A. (2004). Reflections on computational building models. *Building and Environment*, **39**(8), 913-925.

Malkawi, A.M. (2004). Developments in environmental performance simulation. *Automation in Construction*, **13**(4), 437-445.

Malkawi, A.M., Srinivasan, R.S., Yi, Y.K. and Choudhary, R. (2005). Decision support and design evolution: integrating genetic algorithms, CFD and visualization. *Automation in Construction*, **14**(1), 33-44.

McElroy, L.B. and Clark, J.A. (1999). Embedding simulation within energy sector businesses. In: *BS '99: Sixth International IBPSA Conference*, Kyoto, Japan, 1197-1202.

Moeller, G. (2000). STandard for the Exchange of Product model data. *Program Manager*, **XXIX**(6), 52-56.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003a). Integrating Simulation in Design. *IBPSA News*, **13**(1), 21-26.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003b). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

Papalambros, P.Y. and Wilde, D.J. (2000). *Principles of optimal design: modeling and computation*. New York: Cambridge University Press.

Papamichael, K., LaPorta, J. and Chauvet, H. (1997). Building Design Advisor: automated integration of multiple simulation tools. *Automation in Construction*, **6**(4), 341-352.

Radford, A.D. and Gero, J.S. (1988). *Design by Optimization in Architecture, Building and Construction*. New York: Van Nostrand Reinhold Co.

Roseland, M. (2000). Sustainable community development: integrating environmental, economic, and social objectives. *Progress in Planning*, **54**(2), 73-132.

Sutherland, I. (1963). SKETCHPAD: A Man Machine Graphical Communication System. In: *AFIPS Spring Joint Computer Conference*.

Tucker, S.N., Mohamed, S., Johnston, D.R., McFallan, S.L. and Hampson, K.D. (2001). *Building and Construction Industries Supply Chain Project (Domestic)*. BCE DOC. 01/214. Victoria, Australia: CSIRO.

UNCED (1992). *Agenda 21, the Rio declaration on Environment and Development*. New York: The United Nations.

Wang, W., Zmeureanu, R. and Rivard, H. (2005). Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment*, **40**(11), 1512-1525.

Wetter, M. and Polak, E. (2005). Building design optimization using a convergent pattern search algorithm with adaptive precision simulations. *Energy and Buildings*, **37**(6), 603-612.

Wright, J. and Loosemore, H. (2001). The multi-criterion optimization of building thermal design and control. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 873-880.

Wright, J.A., Loosemore, H.A. and Farmani, R. (2002). Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and Buildings*, **34**(9), 959-972.

Chapter 2

Sustainability and integration of simulation in design

Considering the aspects of global climate change and the need for a sustainable built environment Chapter 2 begins by arguing the case for the integration of multi-domain performance assessments in architectural design process. Concepts of building simulation are introduced and applications in simulation-based decision-making are discussed. The barriers to the successful integration of building simulation in the design process are elaborated. The case for the incorporation of mathematical optimisation techniques to better integrate simulation tools in design is made. Ongoing efforts at integration of simulation in design using various approaches are briefly reviewed.

Climate change and the environment

Concerns over global climate change, its causes and effects, are increasing (Houghton *et al.* 2001; Hulme *et al.* 2002). *Climate change* is referred to as the gradual change in the global temperature by the accumulation of GHGs (Greenhouse Gases). Although the term *climate change* is often synonymous with *global warming*, some regions may become colder even though there is a rise in overall average global temperature. GHGs are essentially transparent to the short-wave energy of the sun, but opaque to the long-wave infrared radiation emitted by the Earth's surface. The effect is similar to having a *blanket* of gasses around the Earth keeping it warm by trapping infrared radiation close to the surface of the Earth. The six main GHGs are *Carbon dioxide* (CO_2), *Methane* (CH_4), *Nitrous oxide* (N_2O), *Hydrofluorocarbons* (HFCs), *Perfluorocarbons* (PFCs), and *Sulphur hexafluoride* (SF_6). The most significant of them is CO_2 which makes up nearly 80% of GHG emissions (Ahmad and Wyckoff 2003).

The level of CO_2 in the atmosphere has risen by more than a third since the industrial revolution (DTI 2003). Figure 2.1 shows that the earth warmed up by about 0.6°C over the 20th century largely due to GHG emissions such as CO_2 from human activities (Houghton *et al.* 2001). The 1990s were the warmest decade since records began. The more industrialised the nation, the larger their contribution to global warming. Figure 2.2 shows regional footprints of contributions to global warming with industrialised nations having significant shares.

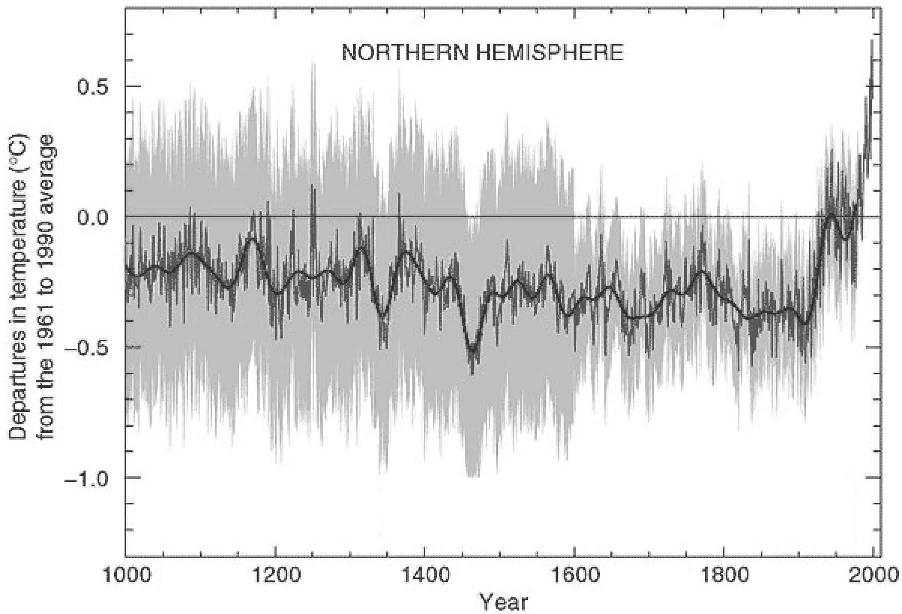


Figure 2.1: Variations of the earth's temperature over the last millennium, after Houghton et al. (2001).

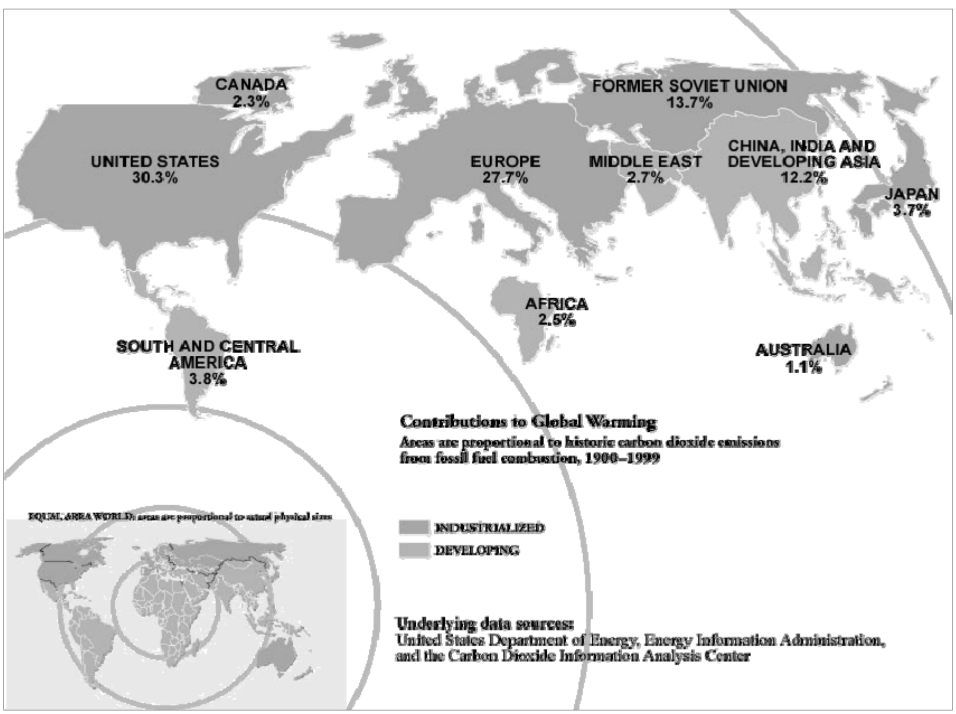


Figure 2.2: Contributions to global warming by regional footprint, after: http://climate.wri.org/pubs_description.cfm?PubID=3982.

Impacts of climate change

GHGs remain in the atmosphere for decades or centuries, depending on the type of gas. Their potential impacts may escalate over time. The rising average global temperature has catastrophic consequences accompanied by changes in the world around us.

Rising sea levels and flooding: According to a recent energy white paper (DTI 2003) published by the United Kingdom Department of Trade and Industry, Global mean sea level rose by an average of 1-2mm a year during the 20th century. This will threaten buildings, roads, and other infrastructure in vulnerable places such as coastal areas.

Negative economic impacts: Destruction of infrastructure due to extreme weather disrupts economic activities. The insurance industry in the US has lost billions of dollars over the last few years because of this. Growing population in areas considered vulnerable to the extremes of weather such as flooding (South East Asia, US East Coast – Florida) could exacerbate the economic impacts. For a worst-case scenario involving category 3 hurricane, surge levels could rise 25 feet above mean sea level at JFK airport and 21 feet at the Lincoln Tunnel entrance in the US (USGCRP 2000).

Depletion of natural resources: The cumulative impacts of other ecosystem stresses caused by human development, such as environmental pollution and habitat destruction could be magnified by climate change. For aquatic systems, the combined effects of climate change and other stresses are likely to bring large-scale irreversible impacts (Burns 2000).

Habitat destruction: In 50-100 years, some major mammals may be threatened by reductions of sea ice from warming. For example, local extinction of polar bears is also possible as ice melts sooner in the spring in the northern ocean and forms later in the fall reducing their hunting season and forcing them to rely on fat reserves longer (James *et al.* 2001).

Disease: Climate change could increase the prevalence/occurrence of some infectious diseases such as malaria, dengue fever, yellow fever, and encephalitis, which are carried by mosquitoes and other insects. Warmer weather would allow those insects to travel farther north (US-EPA 2005).

Water shortages: Rising sea levels will increase the salinity of water in many countries. Climate change will cause further water shortages in regions where summer water supplies are dependent on winter snowfall.

Glacial melting: Summer and autumn arctic sea ice has thinned by 40% in the recent decades. Global snow cover has decreased by 10% since the 1960s. In Montana's Glacier National Park, the largest remaining glaciers are now only a third as large as they were in 1850.

Increased precipitation: Evaporation and average global precipitation have increased as the climate warmed. Soil moisture is likely to decline in many regions, and intense rainstorms are likely to become more frequent. In developed countries, weather-related economic losses to communities have increased ten-fold over the last 40 years. Extreme weather conditions like El Nino and hurricanes have become more frequent and intense during the last 20-30 years.

Reducing the impacts of climate change

Without considerable effort from international communities to reduce emissions, the earth's temperature is likely to rise at a faster rate than anytime in the last 10,000 years or more (DTI 2003). Escape from climate change is unlikely but worst affects can be avoided if GHGs in the atmosphere are stabilised instead of being allowed to increase. It has been suggested that the developed economies need to cut GHG emissions by 60% by around 2050 (Hammond 2004). In carbon terms a reduction to 4.5 tCO_2/cap/yr from 10 tCO_2/cap/yr in the European Union (EU), 20 tCO_2/cap/yr in the United States (US), and 16 tCO_2/cap/yr in Canada and Australia is required to achieve the goals set up in international forums (Engleman 1998). Recent agreements such as Kyoto protocol (UNFCCC 1997) paved the way for concerted international effort by setting up targets to reduce GHG emissions in certain developed countries. Although there are disagreements with regard to the effectiveness (Victor 2004) of the Kyoto protocol as a legal instrument, the protocol brings forward the issue of climate change and the need to reduce GHG emissions.

Energy use in the built environment

The built environment is the major contributor to the factors affecting climate change and global warming representing nearly 50% of greenhouse gas emissions in the UK (Harman and Benjamin 2004). Energy, predominantly in the form of fossil fuel is consumed by buildings over their lifecycle: during construction, operation and demolition/reuse - directly or indirectly. This contributes to the rising levels of GHGs. Buildings also have immediate impacts in terms of the use of other natural resources. They affect the environment in their locality and in the locations that provide materials for construction. Because buildings are typically used for a longer period (generally 50-100 years), their inertia influences future energy use patterns.

Figure 2.3 shows sector wise end user energy consumption in the UK from 1970 to 2004 and Figure 2.4 compares energy consumption of 1990 and 2004. Domestic consumption in 2004 was 48.73 *million tonnes of oil equivalent* compared to 34.08 in industry, 57.45 in transport, and 20.76 in services (other final uses) sector.

Energy use in the domestic sector is on the up and in the UK and is dominated by space heating, which on average accounts for 60% of the total energy consumed (BRE 1992); shown in Figure 2.5. Space heating is mostly affected by design of the buildings, the remaining consumption being largely determined by occupant needs and not strongly dependent on climate (Steemers 2003).

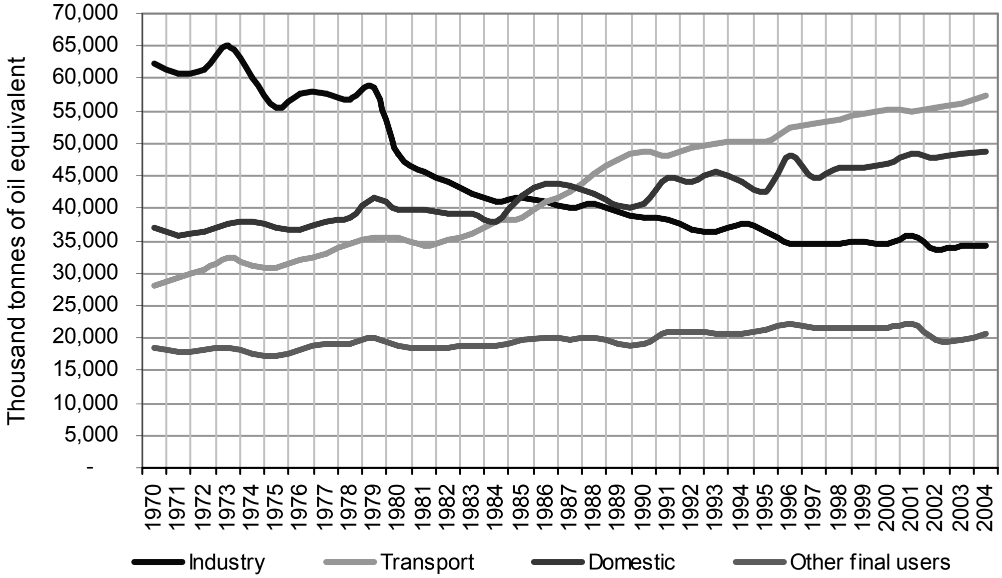


Figure 2.3: Energy Consumption in the UK by sector, 1970-2004, source of data: DTI, UK (2004).

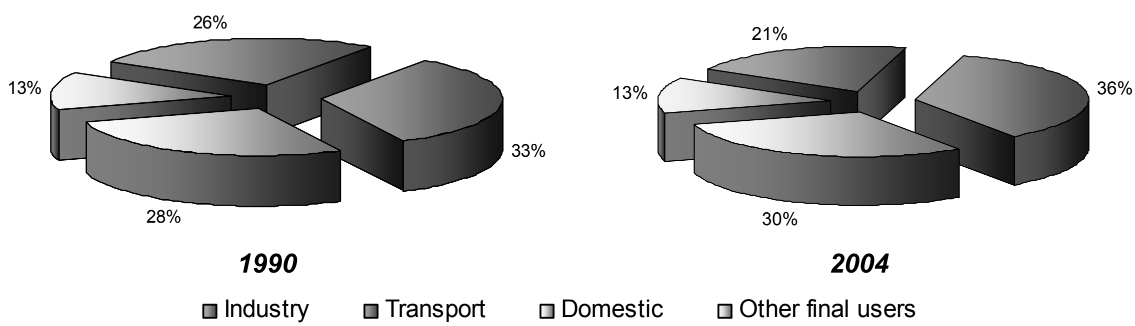


Figure 2.4: Final energy consumption by sector in the UK, in primary energy equivalents, 1990 and 2004, source of data: DTI, UK (2004).

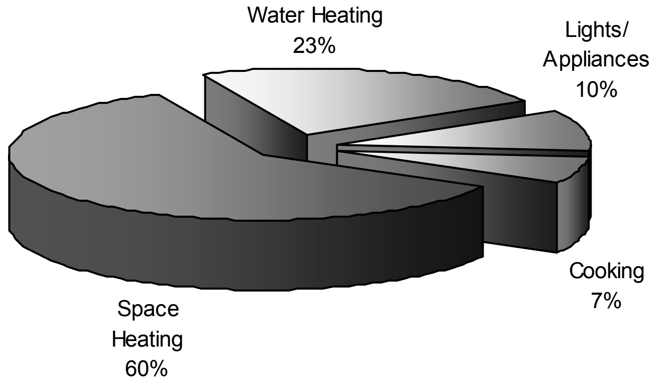


Figure 2.5: Energy use breakdown for UK, domestic sector, source of data: DTI, UK (2004).

Economic impacts of energy consumption

Apart from environmental impacts, consumption of fossil fuel for energy has economic impacts as well. According to the Commission of the European Communities (CEC) (CEC 2001), the EU is becoming increasingly dependent on external energy sources and further enlargement will reinforce this trend. Based on current forecasts, if measures are not taken, import dependence will reach 70% in 2030, compared to 50% in 2001. The demand side, mainly in the sectors of energy efficiency in buildings and transportation has been identified by the CEC as having more potential over the supply side to achieve GHG emission targets set in the Kyoto protocol. Translating the economic and environmental data into actions in the built environment suggests that buildings be made more energy efficient in every life-cycle stage - from inception to recycle/demolition.

Sustainable buildings

Although the concept of the finite global resources dates back to the early 20th century (Fuller 1969), sustainability as an environmental agenda got momentum in the early 1970s with over three times increase in oil prices between 1973 and 1974. Over the years, the scope of sustainability concepts has broadened. Pfeiffer *et al.* (Pfeiffer *et al.* 2005) have identified three main areas of sustainability: ecology, economy and society, encompassing most human activities; shown in Figure 2.6. They are closely intertwined with each other with ecology being the determinant factor for sustainability. There are variations in the definitions for sustainability, mainly how a fourth factor is incorporated in the sustainability equation. Evidence provided in support of the modification of established climate patterns as a result of the combustion of fossil fuel for building activities is overwhelming. Sustainability in the built environment is thus necessary for achieving overall sustainability.

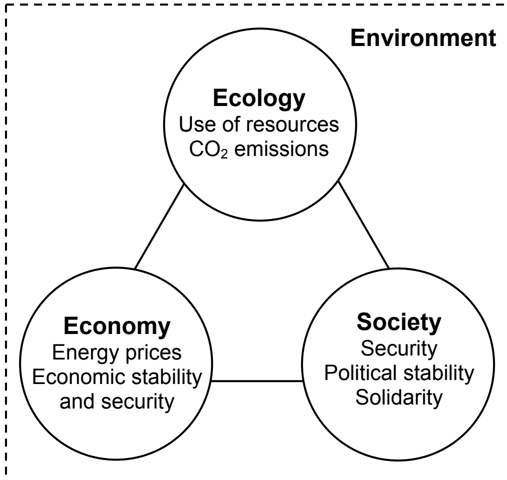


Figure 2.6: Three aspects of sustainability, adopted from Pfeiffer et al. (2005).

The built environment as the focal point for all human activities is of primary concern to the sustainability stakeholders. Numerous suggestions exist to enhance sustainability in the built environment: from increasing the density of urban built form (Steemers 2003) to the promotion of renewable energy (Hughes and Johnson 2005). A close look at the suggestions reveals a renewed emphasis on the integration of *sustainability concepts* and *performance based decision-making* at the earliest in the building lifecycle. Building simulation, by providing quantitative indication of performance, offers significant advantage over qualitative assessments prevalent in the design and construction of buildings.

Efforts at integrating simulation in design

Advances in building simulation, engineering analysis and ever increasing computing power have resulted in building simulation being used for producing designs based on performance analysis. Deployment of building simulation in the design of building services is a natural evolution as the simulation parameters and the performance indicators can be quantified. Developed mostly in research organizations, building simulation programs focus on modelling and simulation rather than on the integration with the design process. Building simulation programs have originally been developed to undertake non-trivial performance appraisals dealing only with a small subset of the problem domain (Citherlet *et al.* 2001). The development efforts, in the last decades were directed at improving the accuracy and efficiency of simulation programs to predict performances. The usability aspects remained largely ignored, particularly those at the early stages of design. Enormous amount of input processing is required even to simulate of a small subset of the domain. Complicated process to accomplish tasks made their use limited to the occasional validation of the proposed idea than to assist in design development holistically. Specialist knowledge and expertise are required more often, hindering the use of simulation. Decisions made at early design stage are important as they dictate the role and behaviour of the building, even environmental ones, at later stages. Mistakes made at this stage are often irreversible and expensive to correct at later stages. The use of building simulation in architectural design, where most of the decisions affecting building's performance are made; is under extensive review and research (Hand 1998; Morbitzer *et al.* 2001; Mourshed *et al.* 2003a). Some key projects aiming at integration of simulation in the design process are described here:

Computer Models for the Building Industry in Europe (COMBINE), one of the early efforts at integration was a major research project within the JOULE programme of the European Commission's Directorate General XII for Science, Research and Development (Augenbroe 1994). The project intended to develop an operational computer based *Integrated Building Design System* (IBDS). The system opted for two integrated systems: the first system integrated more than ten design tools into an Intergraph-based architectural CAD system. The second system, dedicated to HVAC design, is an AutoCAD-based HVAC computer-aided drafting system which integrates *Superlink* for lighting design; *Thermal Simulation of Building Installations 3* (TSBI3), *Energy Simulation Program - research* (ESP-r) and *DOE-2* for detailed thermal simulation; *VENT* for duct sizing; a cost evaluator; a HVAC components database; and *DocLinks* for document management (Hong *et al.* 2000). IBDS architecture of COMBINE-2 project is illustrated in Figure 2.7.

Building Design Advisor (BDA) is a computer program that supports concurrent, integrated use of multiple simulation tools and databases through a single, object-based representation of building components and systems (Papamichael *et al.* 1997). BDA acts as a data manager and process controller, allowing designers to benefit from the capabilities of multiple analysis and visualisation tools throughout the building design process. Decision making as part of the design process is recognized and implemented. Figure 2.8 shows the screenshot of *building browser* and Figure 2.9 shows the screenshot of *decision desktop* of BDA. Elements of building are accessed through the building browser, while the decision desktop allows designers to compare design alternatives with respect to performance indicators assessed by integrated tools. Current version of BDA is linked to *Daylighting Computation Module* (DCM), Electric Lighting Computation Module (ECM) and DOE-2 as energy analysis module. BDA depends on parametric runs to produce comparison data for decision-making. Depending on the number (n) of parameters (where $n > 2$) and number of steps involved, whole process may take hours of computation time and may become hard to visualise for decision making. Integration achieved in BDA is selective, only the implemented simulation tools are available at the disposal of the user. The issues of wider interoperability remain unresolved.

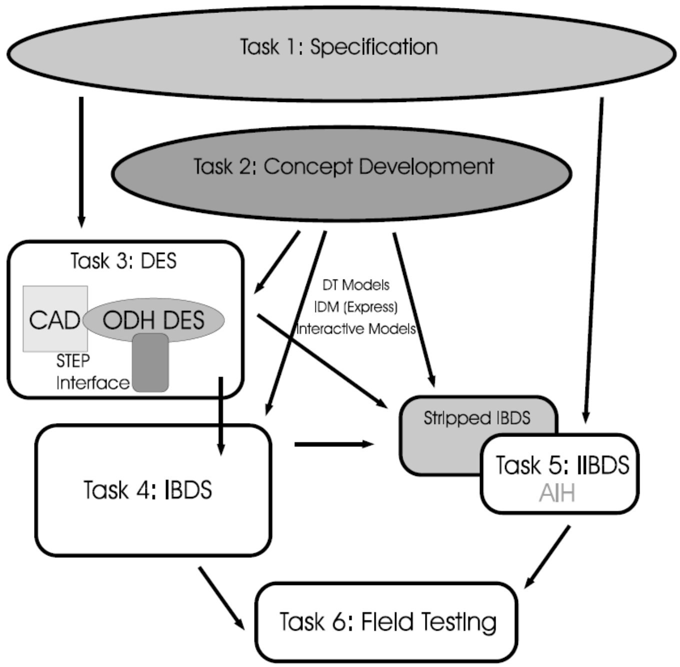


Figure 2.7: COMBINE 2 IBDS architecture, after Augenbroe (1994).

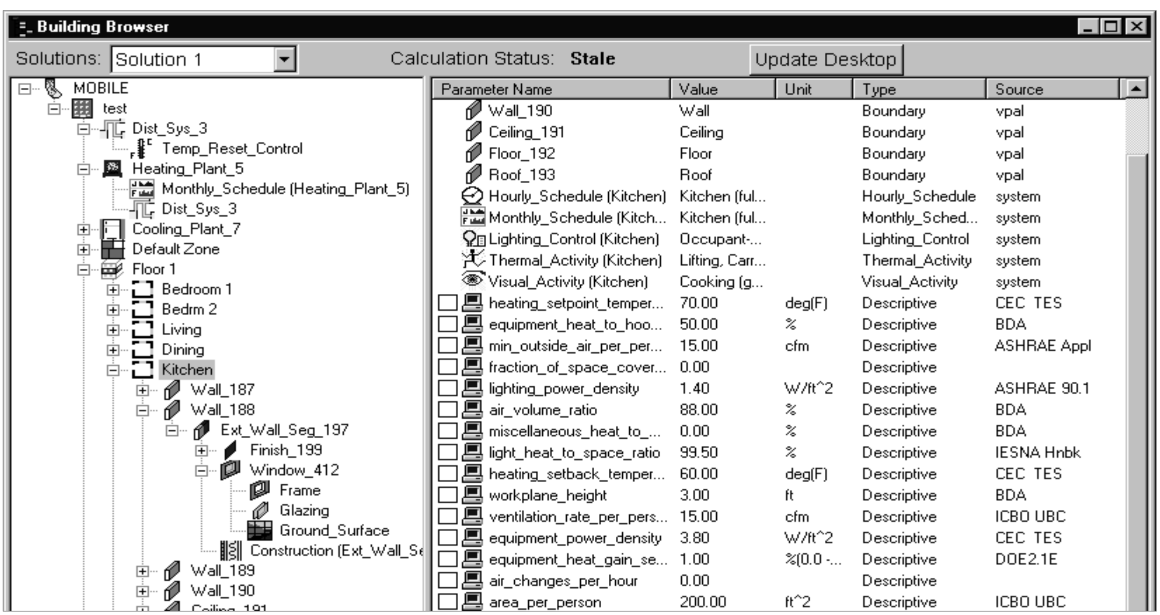


Figure 2.8: BDA Building browser, source: http://gaia.lbl.gov/bda/ bdainfo.htm.

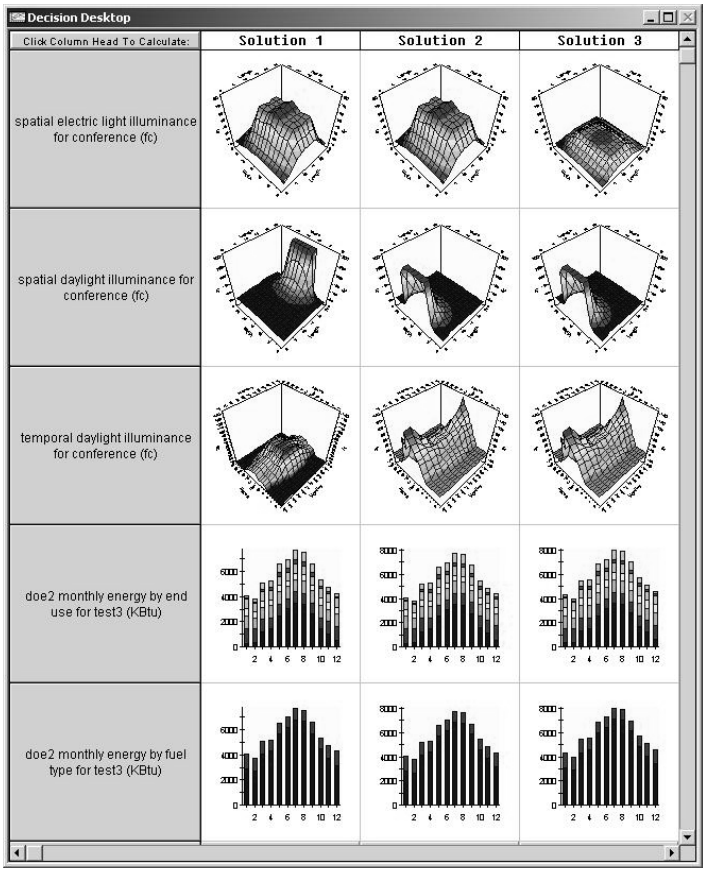


Figure 2.9: BDA Decision Desktop, source: http://gaia.lbl.gov/bda/bdainfo.htm.

Simulation Environment for Modelling Performance (SEMPER) is a design performance simulation environment designed to facilitate iterative, multi-criteria, and multi-agenda building design. SEMPER has been developed at Carnegie Melon University (CMU) and extended as an internet based computational design support environment into SEMPER-II through collaboration between CMU, National University of Singapore and Temasek Polytechnic (Lam *et al.* 2004). SEMPER is discussed further in Chapter 3.

The need for optimisation-based integration

The common theme among the approaches described above is the recognition of the need of several simulation tools for informed decision making in design. Even with the seamless integration of a set of tools, exploration of the solution space remains the primary concern. Number of variables involved in architectural design problems and the permutations of solutions ask for something more than mere interoperability based integration. Mathematical optimisation methods can be applied for effective search in the solution space, furthering automation and integration of simulation in design. Further investigations have been carried out by researchers on the applications of optimisation techniques (Al-Homoud 1997a; Wright and Loosemore 2001; Mourshed *et al.* 2003b; Choudhary *et al.* 2005; Wang *et al.* 2005b; Wetter and Polak 2005). Building simulation tools coupled with optimisation algorithms act as a powerful analysis, visualisation and decision making platform for green building design. Common applications of optimisation techniques include but not limited to: form finding (Mourshed *et al.* 2003b), building envelope optimisation (Hauglustaine and Azar 2001), material selection (Wang *et al.* 2005b) and system sizing (Wright *et al.* 2002; Wetter and Wright 2003). Discussions on the general concepts of optimisation can be found in Chapter 1. Methods and applications of optimisation in architectural and environmental design are elaborated in Chapter 4. Incorporation of optimisation in this thesis is discussed in Chapter 5. Results obtained from case studies involving environmental design of buildings are analysed and compared with previous works in Chapter 6. The following sections of this chapter elaborate on different types of building simulation programs and justify the selection of EnergyPlus for implementation in this thesis.

Building Simulation

Sustainable building regulations, energy labelling, and tax exemption for low-energy buildings - all are contributing towards the increased use of building simulation programs in the design process (Mourshed *et al.* 2003a). Drawing on resources from diverse disciplines such as physics, mathematics, computer science, materials science and physiology, building simulation provides designers with an indication of performance and helps to make informed decisions. Benefits offered by these tools and concepts include but not limited to better *return on investment*, *reduction in waste*, *increased energy efficiency*, *occupant comfort*, and *sustainability*. With the increase in computing power, applications previously restricted to specialized domain problems such as Computational Fluid Dynamics (CFD) is now available at the disposal of AEC professionals for smaller projects. Building simulation continues to evolve - the results are increasingly being incorporated in the process of design and other lifecycle stages, including operation and maintenance. Applications of building simulation range from building thermal performance simulation to *n*-Dimensional (*n*D) simulations such as constructability. Building simulation tools can be classified based on:

- Domain representation,
- Purpose, and
- Theoretical implementation.

Domain representation

Depending on the domain they represent, building simulation tools are of two types: *single domain tools* and *multiple domain* or *integrated tools*.

Single domain simulation

Single domain tools, as the name suggests, have been designed over the past decades to predict the performance of a building in the context of a single domain, such as thermal, lighting, acoustic, structural, etc. As different problems require different simulation algorithms, a number of computational simulation tools exist. The ranges of domain resolution they solve vary from single approximate performance evaluation to the very precise. Solar tool, developed by Square One is an example of a single domain tool in which *3-dimensional* shading on vertical and horizontal surfaces can be modelled, with the help of sun-path diagram to aid decision making in design; illustrated in Figure 2.10.

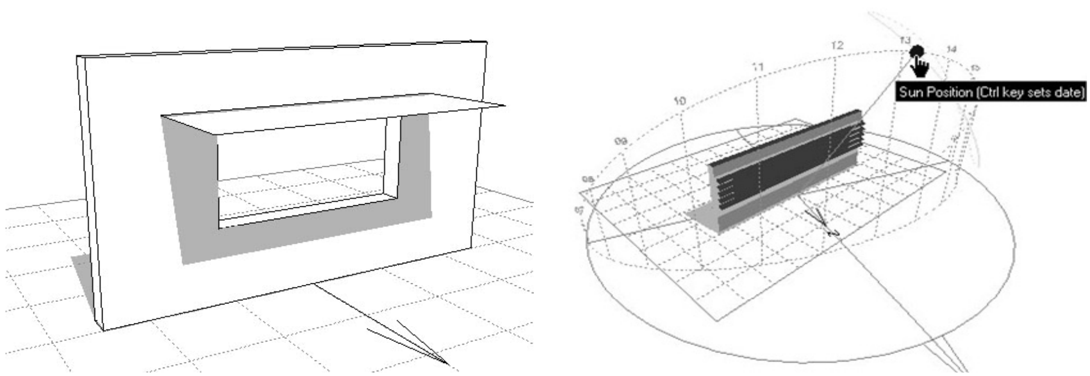


Figure 2.10: Shading study with Solar Tool, source: http://www.squ1.com/

Multiple domain simulation

Multiple domain simulation, on the other hand, facilitates multi-variate performance appraisal by treating several aspects of a building simultaneously. Equation sets defining each domain are processed by customised solvers, while the domain interactions are handled by ensuring that the equation-sets for a given domain are established and solved as a function of information definition of the coupled domains (Clarke 2001a). Multiple domain tools, also known as *integrated* or *whole building* simulation tools, render the accuracy of the appraisal by summing the responses from the component parts, which would not have been possible, had the responses been determined independently. Integrated tools aim to preserve the integrity of the entire building system by simultaneously processing all energy transport paths to a level of detail commensurate with the objectives of the problem to hand and the uncertainties inherent in the describing data. Two examples of widely used integrated simulation tools are: *ESP-r* (Clarke 2001b), developed at Strathclyde University, UK and *EnergyPlus* (Crawley *et al.* 2001), developed by a consortium in the United States.

ESP-r: Principal domains coupled in ESP-r are: building thermal and visual domains, building/HVAC thermal and distributed fluid flow domains, inter- and intra-room air flow domains, and construction heat and moisture flow domains. ESP-r addresses the issue of modelling integrity by ensuring that the mathematical models for conduction, air movement, radiation exchange, moisture flow, electrical power flow, light flow, etc. are processed simultaneously within a simulation. A set of precedence protocols are used to determine the order of invocation of different domain solution; e.g., the air flow domain is solved prior to the building thermal domain because the latter has a larger time constant. The coupling between heat and fluid flow in ESP-r is achieved by iteratively solving the building/HVAC thermal model and a fluid flow network. The flow network is composed of nodes (e.g. rooms and locations within the HVAC system), components (e.g. the leakage paths and pressure drops associated with doors, windows, etc.) and connections joining nodes by components.

EnergyPlus: EnergyPlus is the result of a concerted effort of a consortium consisting of Construction Engineering Research Laboratories (CERL), University of Illinois, Lawrence Berkeley National Laboratory (LBNL), Oklahoma State University, GARD Analytics, and United States Department of Energy (DOE). The development of EnergyPlus began to streamline two separate developments funded by the United States government for building energy simulation: DOE-2 (Winkelmann *et al.* 1993) - sponsored by DOE which uses a room weighting factor approach, and Building Loads Analysis and System Thermodynamics (BLAST) (BSL 1999) - sponsored by the Department of Defense (DOD) that uses a heat balance approach. EnergyPlus combined best features and resources from both BLAST and DOE-2.

EnergyPlus architecture consists of three basic components: a *simulation manager*, a *heat and mass balance simulation module* and a *building systems simulation module*, which is shown in Figure 2.11. The heat balance calculations are based on IBLAST (a research version of BLAST) with integrated HVAC systems and building loads simulation. A building systems simulation manager handles communication between the heat balance engine and various HVAC modules and loops; such as coils, boilers, chillers, pumps, fans and other equipment/components while the simulation manager controls the entire simulation process (Crawley *et al.* 2001). *Simulation Problem Analysis and Research Kernel* (SPARK) (Buhl *et al.* 1993) and *Transient Energy System Simulation Tool* (TRNSYS) (SEL 2000) simulations can also be integrated into EnergyPlus. Interactions between SPARK, EnergyPlus and TRNSYS are controlled by the building systems simulation manager, shown in Figure 2.12.

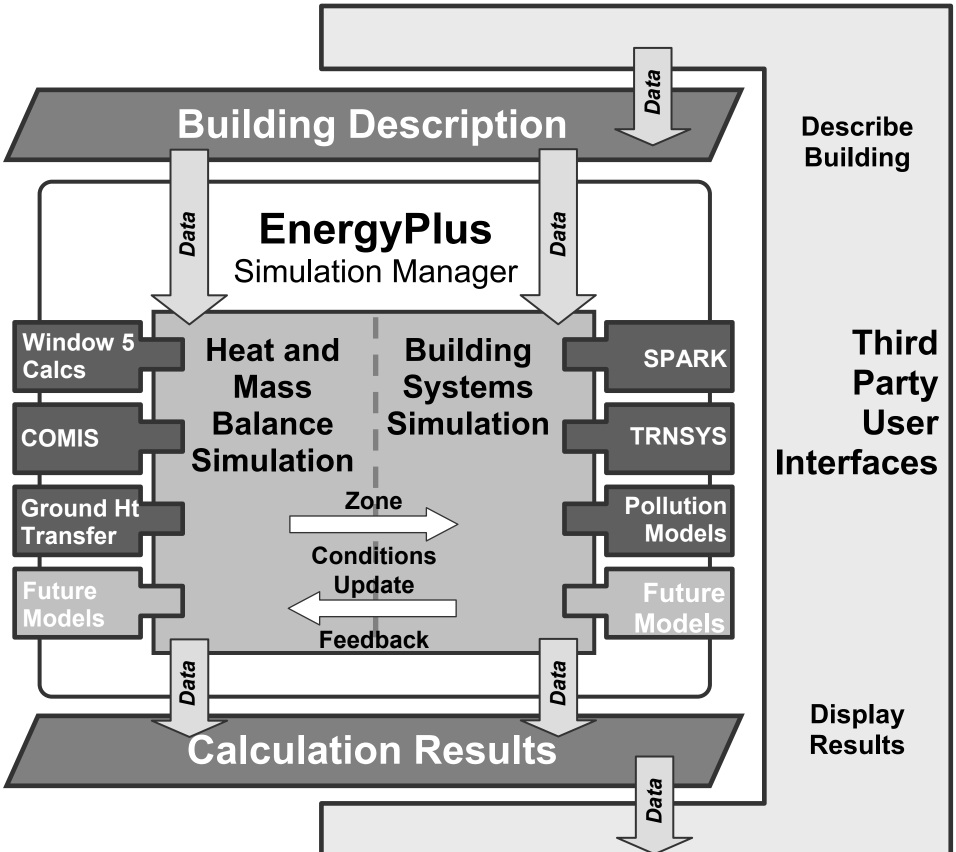


Figure 2.11: Overall structure of EnergyPlus, after Crawley et al. (2001).

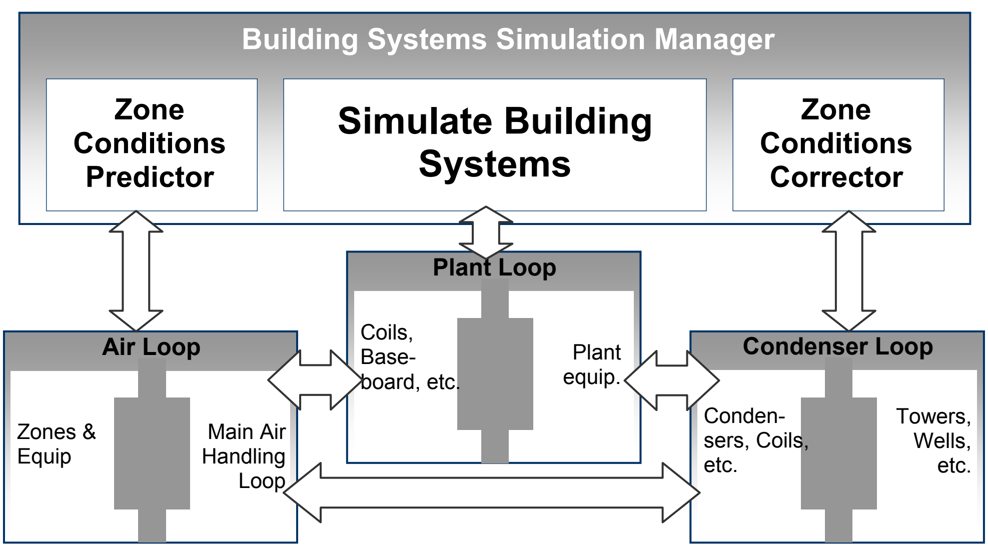


Figure 2.12: Systems simulation manager in EnergyPlus, after Crawley et al. (2001).

Purpose

Based on purpose building simulation tools can be classified as *System Sizing Tools* and *System Performance Evaluation Tools*. System Sizing Tools are used to directly size and detail individual components of a system. System Performance Evaluation Tools simulate the steady or dynamic response of a technical system to specified excitations.

Theoretical implementation

Based on theoretical implementations or modelling approach, building simulation tools are of two kinds: *zonal* and *Computation Fluid Dynamics (CFD)*.

Zonal

Zonal tools are termed as *Macroscopic Analysis Tools* by Axley (2004) and defined as *"tools - based on the application of fundamental conservation principles to building idealisations described in terms of discrete control volumes that lead to systems of algebraic and/or ordinary differential equations"*. Zonal modelling based simulation tools give statistical indication of energy performance of the building for a particular period, defined ahead of simulation run. To reduce complexity and computation time, these models are simplified where every point in space/zone is considered to be in similar thermal state. These tools can again be categorised into two: *steady-state* and *dynamic*. Zonal tools are limited in capabilities to simulate large single space with spatial differences, such as atrium, lecture hall, etc.

Computational Fluid Dynamics

CFD tools, referred to as *Microscopic Analysis Tools* by Axley (2004) are programs based on approximate solutions using relatively fine spatial and/or temporal discretisations of the problem domain where *finite element*, *finite difference* and *finite volume* approaches are used to effect the approximate solution. Broadly, the strategy of CFD is to replace the continuous problem domain with a discrete domain using a grid; e.g., 3D space of a building is divided into a large number of grids. Each node in the grid is assigned an initial value for different environmental parameters. Based on the equations of mass, momentum and enthalpy conservation; assigned values are replaced by solving the equations numerically. CFD tools are able to represent real-life situations more accurately than their zonal counterparts are. However, these programs are generally not suitable for whole-system or longer time-period analyses but provide within-room detailed results.

Adoption of EnergyPlus in this thesis

EnergyPlus has been selected for implementation in this thesis. The selection was based on the following considerations:

Integrated simulation: EnergyPlus performs a comprehensive simulation of the building envelope, HVAC systems, and daylighting and runs at timesteps from 10-minutes to 1-hour. EnergyPlus also eliminated limitations inherent in two of the most popular simulation tools: BLAST and DOE-2 with regard to the number of zones, schedules or systems. This was an essential criterion for selection, as optimisation techniques used in this thesis require overall energy consumption from several domains of discourse to search the solution space effectively. The results obtained form integrated simulations are more accurate than from the combination of multiple single domain tools.

Segregation of engine and interface: The separation of the simulation engine from the interface in EnergyPlus allowed input-output messaging (ASCII based) without the need for *Application Programming Interface* (API) programming. This made it possible to connect EnergyPlus to the optimisation engine - implemented in this thesis.

Modularity: The modular structure of EnergyPlus enabled future addition of modules without re-coding the engine, keeping pace with the new building technologies. Examples include addition of Photovoltaic (PV) and solar thermal hot water systems in a recent EnergyPlus release (Griffith and Ellis 2004).

Accuracy of results: Reports from EnergyPlus testing (Henninger *et al.* 2004) using International Environment Agency's (IEA) BESTEST test suites suggest that the EnergyPlus results generally agreed to within 1% of the analytical results.

Integration with other simulation programs: EnergyPlus facilitates native connection to multiple simulation programs; e.g. *SPARK*, *TRNSYS* and *Conjunction of Multizone Infiltration Specialists* (COMIS) (Feustel and Smith 1997). This allowed flexibility in modelling a diverse set of optimisation problems (e.g. buildings with natural ventilation could be modelled using COMIS).

Summary

Sustainable built environment has been argued here as a way forward to slow down or reverse the adverse impacts of climate change on environment. Consumption of fossil fuel in the construction and maintenance of buildings is shown to be on the up by referring to various data sources. Energy efficiency and conservation in the built environment have been suggested to reduce AEC industry's contribution to climate change. Incorporation of energy efficiency concepts at the earliest in the design process is emphasised. Building simulation has been introduced and discussed in details with regards to the technology and applications. Developments in simulation-assisted design methodologies are elaborated. Barriers to integration of simulation in design are discussed along with the recent efforts at integration. As the success of integration efforts depend on the effectiveness of the system in making decisions, the use of mathematical optimisation methods in simulation-assisted design is proposed.

References

Ahmad, N. and Wyckoff, A. (2003). *Carbon dioxide emissions embodied in international trade of goods*. DSTI/DOC(2003)15. OECD Publications.

Al-Homoud, M.S. (1997). Optimum thermal design of air-conditioned residential buildings. *Building and Environment*, **32**(3), 203-210.

Augenbroe, G. (1994). An overview of the COMBINE project. In: *The first European Conference on Product and Process Modelling in the Building Industry*, Dresden, Germany, 5-7 October 1994, 547-554.

Axley, J.W. (2004). Design and simulation in innovative practice: new directions for building tech research, practice and education. *Journal of Architectural and Planning Research*, **21**(4), 273-291.

BRE (1992). *Domestic Energy Fact File*. Watford: Building Research Establishment.

BSL (1999). *BLAST 3.0 Users Manual*. Urbana-Chapaign, IL: Building Systems Laboratory, University of Illinois.

Buhl, W.F., Erden, A.E., Winkelmann, F.C. and Sowell, E.F. (1993). Recent improvements in SPARK: strong component decomposition, multilayered objects and graphical editors. In: *BS'93: Third International IBPSA conference*, Adelaide, Australia, 283-289.

Burns, W.C.G. (2000). *From the harpoon to the heat: Climate change and the international whaling commission in the 21st century*. Oakland, CA: Pacific Institute for Studies in Development, Environment and Security.

CEC (2001). *Directive of the European Parliament and of the Concil on the Energy Performance of Buildings*. Brussels: Commission of the European Communities.

Choudhary, R., Malkawi, A. and Papalambros, P.Y. (2005). Analytic target cascading in simulation-based building design. *Automation in Construction*, **14**(4), 551-568.

Citherlet, S., Clarke, J.A. and Hand, J. (2001). Integration in building physics simulation. *Energy and Buildings*, **33**(5), 451-461.

Clarke, J.A. (2001a). Domain integration in building simulation. *Energy and Buildings*, **33**(4), 303-308.

Clarke, J.A. (2001b). *Energy Simulation in Building Design*. Woburn, MA: Butterworth-Heinemann.

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl, W.F., Huang, Y.J., Pedersen, C.O., Strand, R.K., Liesen, R.J., Fisher, D.E., Witte, M.J. and Glazer, J. (2001). EnergyPlus: creating a new-generation building energy simulation program. *Energy and Buildings*, **33**(4), 319-331.

DTI (2003). *Our Energy future - Creating a low carbon economy*. Norwich: TSO.

DTI (2004). *Digest of the United Kingdom Energy Statistics 2004*. London: Department of Trade and Industry.

Engleman, R. (1998). *Profiles in Carbon: An Update on Population, Consumption, and Carbon Dioxide Emissions*. Washington: Population Action International.

Feustel, H.E. and Smith, B.V., Eds. (1997). *COMIS 3.0 - User's Guide*. California, CA: Lawrence Berkeley National Laboratories.

Fuller, R.B. (1969). *Operating Manual for Spaceship Earth*. Carbondale: Southern Illinois University Press.

Griffith, B.T. and Ellis, P.G. (2004). Photovoltaic and Solar Thermal Modeling with the EnergyPlus Calculation Engine. In: *World Renewable Energy Congress VIII and Expo*, Denver, Colorado, August 29-September 3 2004.

Hammond, G.P. (2004). Towards sustainability: energy efficiency, thermodynamic analysis, and the two cultures. *Energy Policy*, **32**(16), 1789-1798.

Hand, J.W. (1998). *Removing barriers to the use of simulation in the building design professions*. PhD Thesis. Strathclyde: The University of Strathclyde.

Harman, J. and Benjamin, V. (2004). *Better Buildings Better Lives*. London: Department of Trade and Industry.

Hauglustaine, J.M. and Azar, S. (2001). Interactive tool aiding to optimise the building envelope during the sketch design. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 387-394.

Henninger, R.H., Witte, M.J. and Crawley, D.B. (2004). Analytical and comparative testing of EnergyPlus using IEA HVAC BESTEST E100-E200 test suite. *Energy and Buildings*, **36**(8), 855-863.

Hong, T., Chou, S.K. and Bong, T.Y. (2000). Building simulation: an overview of developments and information sources. *Building and Environment*, **35**(4), 347-361.

Houghton, J.T., Ding, Y., Griggs, D.J., Noguer, M., van der Linden, P.J., Dai, X., Maskell, K. and Johnson, C.A., Eds. (2001). *Climate Change 2001: The Scientific Basis*. Cambridge: Cambridge University Press.

Hughes, B.B. and Johnson, P.D. (2005). Sustainable futures: policies for global development. *Futures*, **37**(8), 813-831.

Hulme, M., Jenkins, G.J., Lu, X., Turnpenny, J.R., Mitchell, T.D., Jones, R.G., Lowe, J., Murphy, J.M., Hassell, D., Boorman, P., McDonald, R. and Hill, S. (2002). *Climate Change Scenarios for the United Kingdom: The UKCIP02 Scientific Report*. Norwich: Tyndall Centre for Climate Change Research, University of East Anglia.

James, J., McCarthy, K., Canziani, O.F., Leary, N.A., Dokken, D.J. and White, K.S. (2001). *Climate change 2001: Impacts, adaptation and vulnerability*. Cambridge: Cambridge University Press.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004). SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. *Automation in Construction*, **13**(5), 651-663.

Morbitzer, C., Strachan, P., Webster, J., Spires, B. and Cafferty, D. (2001). Integration of building simulation into the design process of an architecture practice. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 697-704.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003a). Integrating Simulation in Design. *IBPSA News*, **13**(1), 21-26.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003b). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

Papamichael, K., LaPorta, J. and Chauvet, H. (1997). Building Design Advisor: automated integration of multiple simulation tools. *Automation in Construction*, **6**(4), 341-352.

Pfeiffer, A., Koschenz, M. and Wokaun, A. (2005). Energy and building technology for the 2000 W society--Potential of residential buildings in Switzerland. *Energy and Buildings*, **37**(11), 1158-1174.

SEL (2000). *TRNSYS Version 15 User Manual and Documentation*. Madison, WI: Solar Energy Laboratory, University of Wisconsin.

Steemers, K. (2003). Energy and the city: density, buildings and transport. *Energy and Buildings*, **35**(1), 3-14.

UNFCCC (1997). *Kyoto protocol to the United Nations Framework Convention on Climate Change*. Germany: UNFCCC.

US-EPA (2005). *Global Warming: Climate*. [Online]. Available from: <http://yosemite.epa.gov/oar/globalwarming.nsf/content/Climate.html>

USGCRP (2000). *Climate Change Impacts on the United States: The Potential Consequences of Climate Variability and Change*. Washington: United States Global Change Research Program.

Victor, D.G. (2004). *Collapse of the Kyoto Protocol*. Princeton, NJ: Princeton University Press.

Wang, W., Zmeureanu, R. and Rivard, H. (2005). Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment*, **40**(11), 1512-1525.

Wetter, M. and Polak, E. (2005). Building design optimization using a convergent pattern search algorithm with adaptive precision simulations. *Energy and Buildings*, **37**(6), 603-612.

Wetter, M. and Wright, J. (2003). Comparison of a generalized pattern search and a genetic algorithm optimization method. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 1401-1408.

Winkelmann, F.C., Birdsall, B.E., Buhl, W.F., Ellington, K.L., Erdem, A.E., Hirsch, J.J. and Gates, S. (1993). *DOE-2 Supplement - Version 2.1 E*. Virginia, VA: Lawrence Berkeley National Laboratories.

Wright, J. and Loosemore, H. (2001). The multi-criterion optimization of building thermal design and control. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 873-880.

Wright, J.A., Loosemore, H.A. and Farmani, R. (2002). Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and Buildings*, **34**(9), 959-972.

Chapter 3
Information modelling and interoperability

Chapter 3 introduces Building Information Modelling (BIM) and the concepts of interoperability in the AEC industry. Historical overview of interoperability standards is given. Major standards based on EXPRESS modelling language such as STEP and IFC are described and analysed for adoption in an interoperability-based framework. Implementation issues related to building information modelling are discussed. Recent developments in BIM such as XML-based technologies are introduced. The role of EXPRESS and XML based standards in AEC information modelling and the way they can coexist in semantics-based information world are analysed. A brief overview of previous efforts in the development of interoperability-based integrated framework is given.

Background

Compared to other large industries, AEC (Architecture, Engineering, and Construction) industry is highly fragmented (Latham 1994; 1998; Austin *et al.* 2002; Dawood *et al.* 2002). The fragmentation is characterised by the existence of a large number of relatively small companies employing fewer than eight people (Egan 1998). Buildings are built by temporary (Josephson and Hammarlund 1999), project based organisation of such small firms; not by tightly integrated companies (Cohen 2003) as seen in other industries such as aerospace, automotive, etc. The organisation usually breaks up when the project is completed. Although providing flexibility to deal with highly variable workloads and volatile market conditions, such fragmentation is seen as a contributing factor in the poor communication between stakeholders having significant impact on productivity and performance (Latham 1994; Egan 1998). The ownership of information, the control of information exchange and their associated processes in project lifecycle reside in the hands of separate organisations with their distinctive cultures and work practices.

In addition to the fragmentation at the level of ownership is the associated fragmentation at the level of project team membership which manifests itself in three ways. Firstly, the nature of team membership during the course of a particular construction project; *teams* can be composed of new members joining with an established team and old members breaking off to join new groupings on different projects (Orange *et al.* 1999). Secondly, the fragmentation that exists among the working team comprised of professionals from different backgrounds with distinctive approach to communication and information exchange. Thirdly, AEC enterprises are increasingly becoming part of the global economy through internationalisation of the supply chain and by partnering with geographically dispersed companies in project teams.

Collaboration and integration

Project team members in fragmented and complex organisation of companies in a construction project contribute to the overall success of the project through collaborating among themselves. The term *collaboration* needs to be elaborated as the success of a construction consortium pivots on the concept and parameters of collaboration. Schrage (Schrage 1990) defined collaboration as "*the process of shared creation: two or more individuals with complementary skills interacting to create a shared understanding that none had previously possessed or could have come to on their own*". The importance is on the process of value creation through human activity where technology acts as a platform to augment our ability to communicate. The key requirement here is the platform or the environment that effectively support interaction without treating collaboration as something that depends on technical and interpersonal skills. The interactive platform needs to be designed to cater for different representations that exist among groups of professionals involved in a collaborative activity. From collection to access and exchange of information, collaborative platforms help towards achieving the agreed objectives of a project.

Collaboration in the AEC industry is different from collaboration in other fields as it involves individuals representing often different professions with fundamentally different concepts of the product and the process. The stakeholders comprising a design/construction team rarely share a common educational background, unlike collaborators in other fields such as medicine, jurisprudence, etc. (Kalay 2001). This is further complicated by the duration over which collaboration exists in a construction project. Collaboration in the AEC industry tends to stretch over a prolonged period of time, even when the original participants are no longer involved but their decisions and actions still have impacts on the project.

The synergy that takes place between the stakeholders depends on how effectively they can communicate and exchange information throughout the lifecycle of a project. Failures of communication can lead to cost overruns, delays, errors, quality deterioration and disputes. In fact, most[1] construction problems are caused by inadequate coordination and inefficient means of communication of project information and data (Cornick 1990; Koskela 1992; Austin *et al.* 1994). A number of solutions have been proposed, both from the industry itself and the manufacturing sectors such as automotive and aerospace to enhance communication among collaborators. The solutions range from the development of guidelines for effective communication to the changes in working culture and the construction process. These strategies are geared towards improving the working relationship in the industry to drive technological innovation in construction (Holmen *et al.*

[1] The figure has been reported as two-thirds of the total in some sources (see Dawood *et al.* 2002).

2005). 'Design-build' is one example of *changes in working culture*, based on the concept of a *single point responsibility* where the design and construction of the building is the sole responsibility of a single organisation (Dawood *et al.* 2002). A close look at the proposed alternatives and variations to the existing process reveals that the development of these strategies resulted from the industry's recognition for *integration* of resources and processes.

It has been argued that collaboration as a holistic approach cannot be implemented through paper-based or hybrid computer-paper based communication system prevalent in the AEC industry (Eastman 1999). Accurate and timely access of product data for all concerned parties has always been a major communication challenge (Fischer *et al.* 1998) and at present, it is supported by some form of IT. IT, as implemented in the AEC industry today offers only piecemeal solution to individual disciplines concerned rather than to the overall process, resulting in incompatibility of tools and systems. Even the shared APIs for integrating inter-organisational information processes are based on narrow perspectives of the whole context and do not cater for effective collaboration. The use of the existing shared APIs is considered to be one of the major bottlenecks of collaborating information systems; hence the need for the development of richer and integrated information models are emphasised by researchers (Brown *et al.* 1995; Rezgui *et al.* 1996; Sun and Lockley 1997; Fischer *et al.* 1998; Zhu and Augenbroe 2006). They are envisaged to provide better support for inter-organisational collaboration taking advantage of developments in computer technologies.

Islands of information

Although creating new opportunities for collaboration, coordination and information exchange among organisations in the AEC industry; the standardisation efforts over the last two decades have created a host of new problems that did not exist before (Cohen 2003). Data standards are usually developed with a narrow focus; fosters vertical integration of software systems and business processes. New applications or subsystems of existing applications are developed to dovetail with the overall structure provided by the standard (Tolman 1999). The problem remains and in certain cases increases due to the need for horizontal integration segregating different professional groups further. The resulting *archipelagos of integration* is referred to as *islands of information*[2], shown in Figure 3.1. With the development of new standards and technologies, water level drops; exposing a greater range of applications and greater opportunity for information exchange (Froese 1994).

[2] The term *islands of information*, also referred to as *islands of automation* by many researchers as computer systems automate business processes.

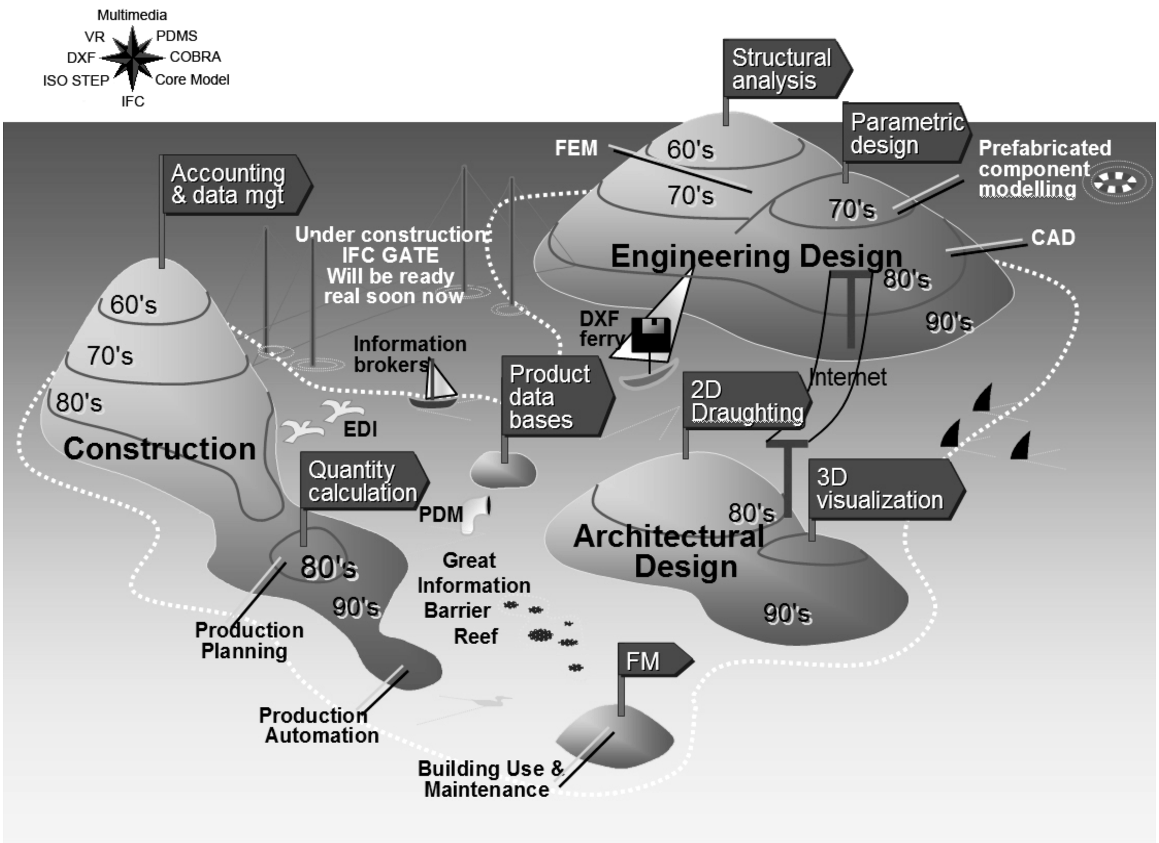


Figure 3.1: Islands of Information in the AEC industry, after Hannus et al. (1995).

Financial case for interoperability

Focus has been shifted over the years from piecemeal automation of business processes towards the achievement of interoperability. It has been reported that in the 1990s, while interoperability productivity benefits were being realised in other industries, the AEC industry went largely unaffected. In fact, during the 1990s AEC was the only segment in the US economy to experience a decline in productivity (AISC 2004). The NIST study (NIST 2004) into the cost component of inadequate interoperability in the US capital facilities industry suggests that nearly US$15.8 billion was lost in 2002 due to the lack of interoperability in the construction industry. More than US$10.6 billion, about 68% of the cost was borne by owners/operators and the rest were borne by other stakeholders. Table 3.1[3] shows the cost component for different stakeholder groups in different

[3] Source: NIST (2004). Includes commercial, institutional and industrial buildings totalling 1.1 billion sq. ft in "new" and 39 billion sq. ft. in "set in place" construction.

lifecycle phases. The significance of interoperability is evident in the fact that for 1.1 billion sq. ft. in new construction, which correlates to the planning, engineering, design, and construction phases of a facility's lifecycle; the cost is US$6.18 per sq. ft. as opposed to US$0.23 per sq. ft. for operation and maintenance of existing 39 billion square feet.

NIST report also suggests that 85% of the operating and maintenance costs, approximately US$9 billion, were borne by the owners/operators. This is due to the failure to manage activities upstream in the design and construction process. Poor management and maintenance of as-built data, communication failures, inadequate standardisation and inadequate oversight during each lifecycle stages add up to the downstream costs. Industry stakeholders indicated that there are additional costs for inefficiencies such as lost business opportunities that were not calculated into the final figures due to its speculative nature; in addition to the quantifiable costs cited in the study.

Table 3.1: Cost of inadequate interoperability by stakeholder groups, by lifecycle phase (in US$ millions).

Stakeholder Group	***Planning, Engineering, Design Phase***	***Construction Phase***	***O&M Phase***	***Total***
Architects and Engineers	1,007.2	147.0	15.7	1,169.8
General Contractors	485.9	1,265.3	50.4	1,801.6
Specialist Contractors/Suppliers	442.4	1,762.2	---	2,204.6
Owners and Operators	722.8	898.0	9,027.2	10,648.0
All Stakeholders (Total)	2,658.3	4,072.4	9,093.3	15,824.0

Achieving interoperability

Software interoperability[4] can be achieved by three methods: developing point-to-point data translators; mandating the use of proprietary software tools across an industry or a project; or establishing vendor-independent neutral data standards.

Point-to-point

Point-to-point data translators are the crudest of all three. It involves expensive development of pairs of data translators. Number of translators required to transfer data both ways between n different software applications using point-to-point method can be defined as

[4] Interoperability has been referred here in its broader context. File-based point-to-point transfer also contributes in achieving interoperability which may or may not offer seamless integration.

$$N = n(n-1) \tag{3.1}$$

where N, the number of translators required and n, the number of software programmes involved. Equation 3.1 is illustrated in Figure 3.2 which shows that the required number of translators increases with the number of software applications involved in a process. With every new release of software system, updates of the corresponding translators become necessary, making it expensive to maintain and develop.

The development of point-to-point translators is not always error free. The underlying data structure upon which a software system is built may be different from that of the target software. In such cases, customised mapping of two data structures, i.e. mapping of object classes and relationships, is performed. Object classes and relationships may also be different between the source and the target systems making it necessary to introduce the missing object classes and relationships. Assuming that all information is available in the source system and the target system is capable of importing all information from the source system, four different mapping scenarios (de Vries 1996) are shown in Table 3.1.

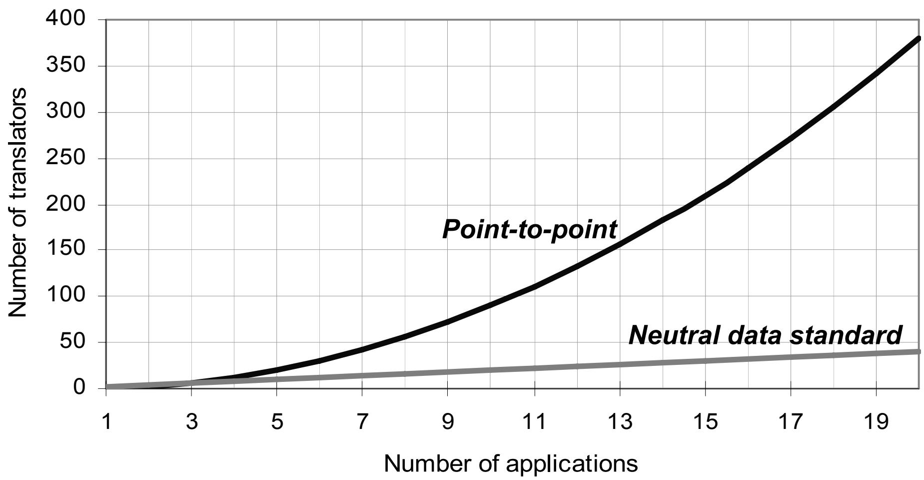


Figure 3.2: Number of translators required for n different systems for information exchange using point-to-point and neutral data standard.

Table 3.2: Mapping scenarios between two systems.

	Object class	***Relationship***
1	Identical	Identical
2	Different	Identical
3	Identical	Different
4	Different	Different

Mandating the use of proprietary software

Mandating the use of proprietary software in a construction project is the most restrictive of the three approaches in adapting to new technologies. Guidelines are usually provided in the form of a toolkit for project participants to follow. Proprietary solutions are often used in large supply chains that require stakeholders to conform to a vendor-specific software solution. Although this method is not a long term solution to the age old problem of interoperability, there is anecdotal evidence that this has been the most preferred of the three in the construction industry.

New software tools automating certain parts of the process cannot be integrated into a close-loop software arrangement even though there are business and environmental benefits. For example, building simulation software becomes difficult to integrate into the loop, as often it does not conform to the data structure of the dominant stakeholder's software environment.

Neutral data standards

As opposed to point-to-point method, neutral data standards allow stand-alone software programmes to communicate with each other by translating a programme's native format into a neutral format to allow information exchange across multiple platforms. The term *neutral* refers to the standards independent of vendors and proprietary intellectual rights. Some neutral data standards have *Standard Data Access Interface* (SDAI) enabling stand-alone programmes to directly transfer data to other programmes or databases without using file-based transfer. In comparison with point-to-point method, the number of translators required are greatly reduced which can be defined as

$$N = 2n \quad (3.2)$$

where N, the number of translators required and n, the number of software programmes involved. Equations 3.1 and 3.2 are illustrated together in Figure 3.2 which shows that to achieve interoperability among a network of 20 software programmes using a neutral data standard only 40 translators are required as opposed to 380 for point-to-point method. Figure 3.3 illustrates different information exchange methods.

Neutral data standards vary from vendor driven exchange formats to agreed-upon standards developed collaboratively by industry stakeholders. The context in which neutral data standards have been established has changed over the years. Preliminary efforts at standardising data exchange format have resulted in *Initial Graphic Exchange Specification* (IGES) which establishes information structure to be used for the digital representation and communication of product definition data. *Drawing eXchange Format* (DXF) and *Data eXchange Binary* (DXB) are proprietary file formats for transferring simple drawings, developed by Autodesk™. Recent efforts are geared towards semantics based *information modelling* considering the objects, attributes of objects, relationships and constraints of objects/attributes.

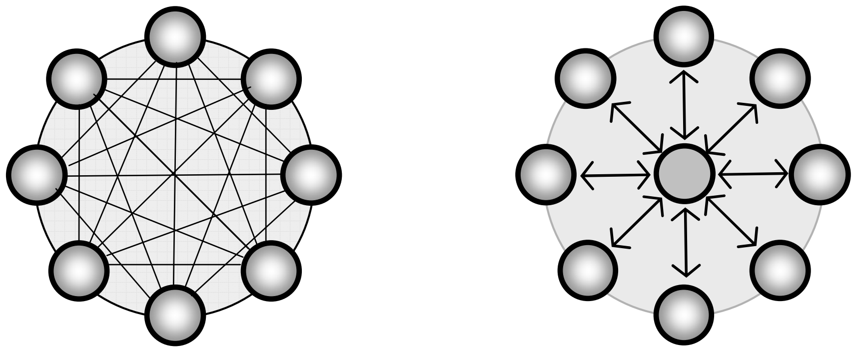


Figure 3.3: Methods of information exchange.

IGES (Initial Graphic Exchange Specification)

The file format defined by this specification treats product definition as a file of entities represented in an application-independent format, to and from which the native representation of a specific CAD/CAM system can be mapped (NBS 1998). The entity representations provided in the specification includes forms common to the then existing or emerging CAD/CAM systems. The products are described in terms of geometric and non-geometric information. Non geometric information is divided into annotation, definition and organisation. The geometry category consists of elements such as points, curves, surfaces, and solids that model the product being described. The annotation category consists of the elements which are used to clarify or enhance the geometry, including dimensions, drafting notation and text. The definition category identifies groupings of elements from geometric, annotation, or proprietary data which are to be evaluated and manipulated as single items.

The IGES file may be in one of the three formats: ASCII[5], compressed ASCII, or binary. The ASCII based IGES file format consists of five sections: a *start* section, a *global* section, a *directory* section, a *parameter* section, and a *terminate* section. The start section is a place for human-readable comments. The global section contains information such as the units used in the IGES file, scale factor, the date the file was generated, and so forth; for use by the post-processor. The pointers in the directory section that indicate where other data are to be found are simply sequence numbers. The parameter section contains detailed data for each entity in a free-form style with each field separated by an end-of-field delimiter and each record terminated with an end-of-record delimiter. The terminate section consists of a single 80-character record containing the number of records

[5] ASCII (American Standard Code for Information Interchange) is a character encoding based on English alphabet.

used for each section of the file. Example of an IGES 3D model is given in Figure 3.4. Partial IGES file format, encoded in ASCII is shown in Figure 3.5.

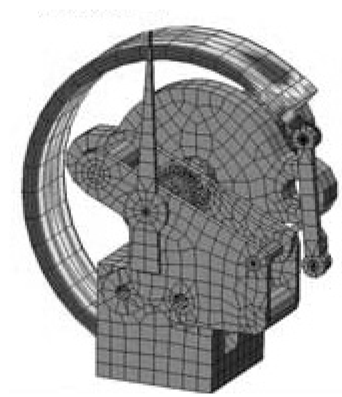


Figure 3.4: Example of an IGES model.

```
IGES translator from Autodesk Inc., translator version IGES3.0          S      2
                                                                        S      3
1H,,1H;,4Htest,8Htest.igs,13HAutoSolid 3.1,7HIGES3.0,32,8,24,12,52,     G      1
4Htest,1.0,1,4HINCH,32767,32.766998,13H910326.082948,9.9999999E-09,2.0, G      2
12HWiley Coyote,10HACME, Inc.,4,0;                                      G      3
     106       1       0       1       1       0       0       000000001D      1
     106       0       0      34      11                               0D      2
     106      35       0       1       1       0       0       000000001D      3
     106       0       0      34      11                               0D      4
     106      69       0       1       1       0       0       000000001D      5
     106       0       0      34      11                               0D      6
     106     103       0       1       1       0       0       000000001D      7
     106       0       0      34      11                               0D      8
     106     137       0       1       1       0       0       000000001D      9
     106       0       0      34      11                               0D     10
     106     171       0       1       1       0       0       000000001D     11
     106       0       0      34      11                               0D     12
.
.
.
     106    3557       0       1       1       0       0       000000001D    371
     106       0       0      17      11                               0D    372
106,1,100,0.0,4.6999984,4.8564682,4.6507363,4.7730103,4.5999112,       1P      1
4.6905069,4.5475774,4.6090484,4.4937944,4.5287275,4.4386225,           1P      2
4.4496326,4.3821225,4.3718529,4.3243594,4.2954755,4.2653952,           1P      3
4.2205844,4.205297,4.1472645,4.1441326,4.0755973,4.0819688,            1P      4
4.0056624,4.0188761,3.9375384,3.9549243,3.8713012,3.8901856,           1P      5
3.8070247,3.7586362,3.6846399,3.6248152,3.57093,3.4893196,             1P      6
3.4664021,3.3527555,3.3715241,3.2157328,3.2867198,3.0788629,           1P      7
3.2123673,2.9427569,3.1487987,2.8080232,3.0962982,2.6752639,           1P      8
3.0551007,2.5450716,3.0253897,2.4180274,3.0072975,2.2946992,           1P      9
3.0009055,2.1756375,3.006242,2.0613742,3.0232835,1.9524195,            1P     10
3.0519536,1.8492601,3.092124,1.7523569,3.143616,1.6621425,             1P     11
3.2061987,1.5790197,3.2795932,1.5033599,3.363472,1.435501,             1P     12
3.4574594,1.3757461,3.5611365,1.3243622,3.6740403,1.2815784,           1P     13
3.7956665,1.2475864,3.9254713,1.2225373,4.0628753,1.2065434,           1P     14
.
.
.
-1.7722948,-3.7763236,-1.7160469,-4.0600858,-1.6852584,              371P   3571
-4.3611131,-1.6806873,-4.6719942,-1.7024462,-4.9850731,              371P   3572
-1.7499994,-5.2926416;                                               371P   3573
S      3G      3D     372P   3573                                       T      1
```

Figure 3.5: ASCII encoding of an IGES file, source: http://www.brlcad. org/reports/arl-tr-315/iges.html.

IGES is an American national standard (ANSI Y14.26M); not without some serious limitations. It is restricted to partial geometry and annotation although it includes structure and relationships (Choo 2004) among entities. The other criticism on IGES is that vendor implementers developing IGES processors have the freedom to describe certain entities in a way of their choosing which creates potential entity-mismatch situations; hence IGES-based data exchange between different CAD systems require a lot of manual intervention. IGES still commands a sizeable stake in the *Mechanical CAD* (MCAD) industry due to the fact that a large of sum has been invested in IGES technologies and also a large amount of data exists in IGES.

DXF (Drawing eXchange Format)

Being a vendor driven specification DXF lacks the backing of an international standard. This lightweight specification is suitable for transferring simple drawings only if it is the highest common exchange factor linking the systems. Developed by Autodesk™, DXF may be of ASCII or binary format. DXF is intended for drawing interchange between AutoCAD systems and lacks semantics offered by neutral standards based on information modelling approach. Figure 3.6 shows an example of ASCII DXF file and corresponding 3D model.

```
999          20          0           0          -0.5   LINE
VISION3D DXF 0.0         ENDTAB      ENDSEC     33     8
0            9           0           0          -0.5   1
SECTION      $LINMAX     TABLE       SECTION    0      62
2            10          2           2          3DFACE 1
HEADER       1000.0      LAYER       ENTITIES   8      10
9            20          70          0          1      0
$ACADVER     1000.0      6           3DFACE     62     20
1            0           0           8          1      -0.5
AC1006       ENDSEC      LAYER       1          10     30
9            0           2           62         -0.5   0.5
$INSBASE     SECTION     1           1          20     11
10           2           70          10         -0.5   0
0.0          TABLES      64          -0.5       30     21
20           0           62          20         -0.5   0.5
0.0          TABLE       7           -0.5       11     31
30           2           6           30         0.5    -0.5
0.0          LTYPE       CONTINUOUS  -0.5       21     0
9            70          0           11         -0.5   ENDSEC
$EXTMIN      1           ENDTAB      -0.5       31     0
10           0           0           21         -0.5   EOF
0.0          LTYPE       TABLE       0.5        12
20           2           2           31         0
0.0          CONTINUOUS  STYLE       -0.5       22
9            70          70          12         -0.5
$EXTMAX      64          0           0.5        32
10           3           0           22         0.5
1000.0       Solid line  ENDTAB                 13
20           72          0           0.5        0
1000.0       65          ENDSEC      32         23
9            73          0           -0.5       -0.5
$LINMIN      0           SECTION     13         33
10           40          2           0.5        0.5
0.0          0.000000    BLOCKS      23         0
```

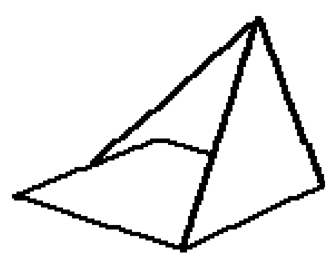


Figure 3.6: ASCII encoding of a DXF file and corresponding 3D model, source: http://astronomy.swin.edu.au/~pbourke/geomformats/dxf/min3d.html

Other efforts at the development of neutral data exchange standard

Due to the fact that IGES started out as an American standard, it prompted standards organisations in other countries mainly in Europe to create their own IGES-like product data exchange standards. There was SET (Standard d'Echange et de Transfert), developed by the French Aérospatiale in the early 1980's which resulted in a smaller file size than the equivalent IGES. SET failed to gain a sizeable share in product data exchange community outside France. VDA-FS (Verband Der Automobilindustrie Flachen Schnittstelle) was developed by the German automotive industry to exchange 3D wireframe and surface geometry only. The resulting file format was very concise and quite widely used within its scope. In the US a new standardisation effort was ongoing at the same time in the early 1980s called PDES (Product Data Exchange Standard). ISO (International Standards Organisation) in Geneva, Switzerland initiated the development of STEP (STandard for the Exchange of Product Model Data) in response to the need for a homogenous data exchange standard across Europe. After initially operating as parallel but separate activities, the PDES and STEP efforts merged in 1991. STEP and the latter specification for the AEC industry, IFC (Industry Foundation Classes) are discussed in detail later as they directly relate to this thesis.

Information modelling

A model refers to some description of a system that may be physical or virtual. Models help to design, visualise, plan and communicate. They are considered essential for complex projects. Information on the other hand concerns knowledge, communication and data (Schenck and Wilson 1994). For *data* to be considered information there needs to be an agreement as to the meaning of the data. Same data could change meaning based on the context; e.g. the number *20* could be a length measure if it refers to the length of a wall in metres or it could the number of participants in a construction project.

An information model is a formal specification in which an unambiguous definition of a domain of interest is provided. The information model defines a given domain in terms of the *entities* (objects) that exist in that domain, their characterising *attributes* (fields), the *relationships* between those entities/attributes and *constraints* on those entities/attributes. Furthermore, it addresses the underlying meaning of data, independent of implementation technology.

The existence of a formal description of a domain of interest allows domain experts to work towards a common understanding of the subject area; in other words, information is modelled based on semantics. When the structure and description in an information model is agreed upon by stakeholders; the consensus removes the main obstacle to standardisation. It allows software based on the information model to be interoperable and in the AEC industry it is used as the basis for the creation of a syntax optimised for the efficient support of information exchange during all lifecycle stages.

Building product models

Information modelling efforts in the AEC industry is generally termed as building product modelling and resultant models are termed as BPMs (Building Product Models). Before Object oriented information modelling, geometric representations of shape of an object were used as the basis of a model. Semantics are added to the geometric models, either by means of ‘attributes’ added to geometric entities, or by means of external databases linked to geometric entities (van Leeuwen 1999). This was proved to be unsuitable for the AEC industry because (Luiten 1994):

the shape of a product may change at different stages in its lifecycle; e.g., composition of a wall may change from blocks to bricks changing its construction and shape,

information may exist before the shape information is known; e.g., information about project team members or spatial programmes defining spatial functions and relationships, and

different shape representations are used by different project participants; e.g., A beam and column in the view of an architect are regarded as physical elements with certain dimensions in a space which may be different from the perceptions of a structural engineer, where the primary interest is in the functions of the building’s structure. Representational difference is illustrated in Figure 3.7.

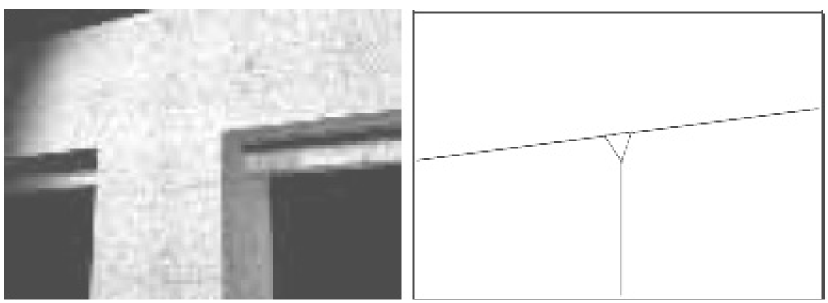


Figure 3.7: Distinct views on a beam and column in architectural design and structural engineering respectively, after van Leeuwen (1999).

Objectives of building product modelling

Information modelling throughout all lifecycle stages was the basis for the development of STEP and IFC, which sets them apart from other data exchange standardisation efforts. Eastman (Eastman 1999) summarised the objectives of product modelling efforts as to:

- incorporate new programming language concepts, especially those dealing with OOP (Object Oriented Programming),
- incorporate formal specifications of the structures defined, using the new recently developed data modelling languages,
- separate the data model from the physical file format,

- support subsets of a total model, allowing clusters of applications to be integrated without the overhead of having to deal with parts of a model irrelevant to a task,
- support alternative physical level implementations, including files, databases and knowledge based systems, and
- incorporate reference models that are common shared subsets of larger standard models.

Concepts of building product modelling

The specific views a professional in the design-construction consortium hold on information related to building as a product and a process have important bearings on approaches to product modelling. The notion of views can be explained in the way an object is understood, described and represented by different professionals. For example, in its simplest form a *space* is understood by an architect as a three-dimensional entity enclosed by vertical and horizontal elements such as walls, floor and ceiling/roof. To a building services engineer, a *space* or *zone* is a collection of rooms having same temperature with/without internal walls. The set of information relevant to different participants may also vary - e.g. a cost engineer may not be interested in geometry. These different views on building information lead to the definition of different information models for the particular requirements of each participant (van Leeuwen 1999).

When humans interpret documents, translation of information from one view to the other depends on the cognitive ability, developed over the years through training and practice. Deciphering information in this way involves project participants to be able to read and make sense of information stored in all documents produced by the stakeholders. Computer software is expected to behave in similar ways - to read and make sense of information which may not be in the structure or details the software in question requires. This has been the primary form of software development and standardisation efforts. Many researchers in the AEC industry - notably, Eastman (Eastman 1992), Rosenman and Gero (Rosenman and Gero 1996), van Nederveen and Tolman (van Nederveen and Tolman 1992), and Amor and Hosking (Amor and Hosking 1993) have worked on the notion of views to define information models and data exchange methods. The concepts of views has also been adopted in the definition of IFC, the most accepted data exchange standard in the AEC industry. Three different types of models can be found: *view model*, *core model* and *aspect model*.

Models defining the concepts of a discipline, application or a certain view of the building as a product or process are called *view model*s. Scope of a *view model* is limited by the domain of the particular area of application. Different *view model*s can exist separately to each other to describe information concerning a single building for different areas of application.

*Core model*s or *kernel model*s can be used to define the relationship between different *view model*s. Information relevant to all or most of the application/participants are defined in *core model*s in a way convenient and agreeable to all of them. For example, information regarding the coordinates of a wall is required by both the architectural and structural applications, regardless of their interpretation of the object (wall). Coordinates can be modelled as part of the core model and shared by architectural and structural view models. This eliminates data duplication and problems regarding data persistence.

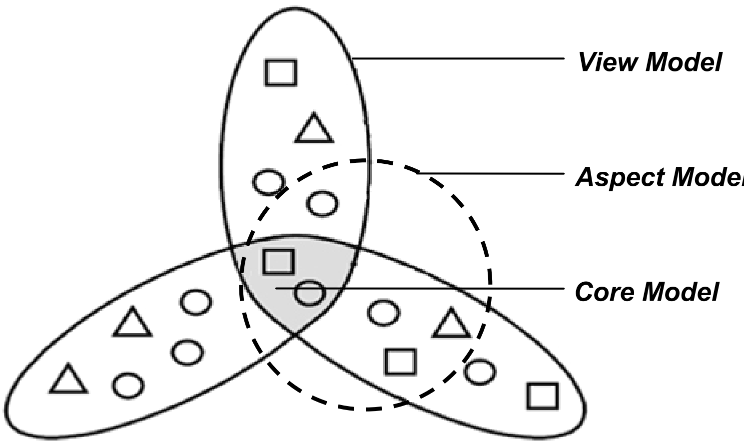


Figure 3.8: Product modelling concepts.

As design and construction of a building depends on the successful understanding of information from more than one discipline, computer software for a particular domain often needs to extract and consume data modelled for other domains. Such combination of data collected from different *view model*s for consumption by software is known as *aspect model*s. One of such scenarios has been addressed in this research where data for optimisation of environmental design of buildings come from architectural and HVAC views. It needs mentioning that, *aspect model*s are task oriented and a flexible method of defining them is necessary for integration of building product models into work practice. The difference between *view model*s and *aspect model*s is clear; *view model*s are defined in the product model whereas *aspect model*s can be defined during software development or at run time. Aspect models are also referred to as sub models in some texts (van Leeuwen 1999). Figure 3.8 shows different product modelling concepts.

Product modelling architectures

Concepts of product modelling, particularly those of different types of models can be organised and defined in different ways. Hannus *et al.* (Hannus *et al.* 1995) have identified six different product modelling architectures. Four types of architecture: *application domain models*, *mutually exclusive common models*, *common resources* and *common core models* are described here. The remaining two: inter-application mapping and neutral model are not described as they define data exchange standards than interoperable building product models. For graphical representation of all four described here please refer to Figure 3.9.

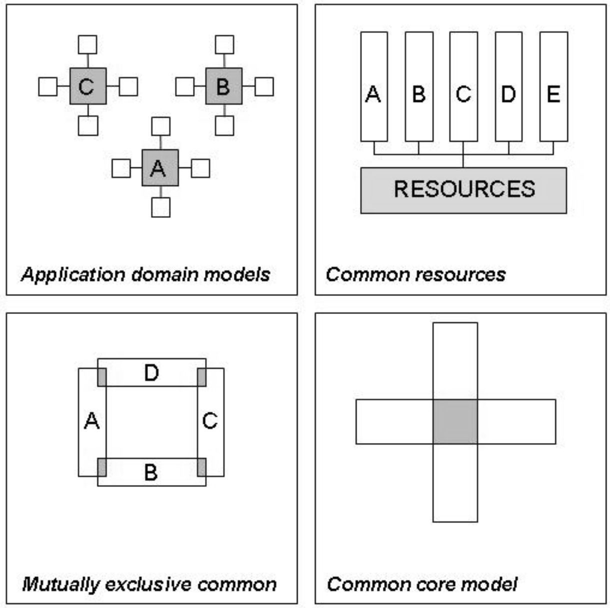


Figure 3.9: Product modelling architectures, after Hannus et al. (1995).

Application domain models

In this approach, neutral models are defined that are valid only for a limited group of applications; preferably when the domain of the group can be identified. Application domain models are neutral models of a smaller scope and are easier to define, maintain and standardise. The problem with cross-domain information exchange and sharing still remains as the architecture is defined to cater for a particular domain. Some of the problems of cross-domain exchange and sharing can be ameliorated if combined with other architecture(s); e.g. STEP has been defined using two architectures: application domain models and common resources. APs (Application Protocols) defined in the STEP are based on application domain models.

Mutually exclusive common models

Two applications sharing data may define a common model that may not be of interest to a third application. This peer-to-peer architecture of inter-application mapping through exclusive integration allows information sharing of a high semantic content. It can thus improve communication between two applications considerably. This architecture is a modified variant of the point-to-point data transfer methodology with the advantages of semantics sharing between two applications. When daisy chained in closed loop integrated system this architecture may offer leverage where the number of systems involved are less and manageable. Disadvantage of this approach are typical of point-to-point data transfer methods: expensive to define and manage.

Common resources

The problem of isolated neutral models not being able to communicate is overcome in the STEP developments by using common resources as the basis for different application specific neutral models or so called APs. Applications can communicate across domains by means of their common basis defined in the resources. This approach essentially avoids the cumbersome task of defining a standardised and neutral model covering all data exchange and sharing requirements across all domains in the industry. Common resources architecture allows application domains to communicate low level data representations.

The architecture lacks ideally seamless communication of high-level data semantics because incorporating them in the resources level is difficult. STEP efforts are mainly focused on the development of the APs, which are based on the common resources containing generic STEP data models. However, it has been argued that the common basis is not sufficient for data exchange between different domains of applications, which is indicated by the recent developments within STEP on standardised procedures for the translation of one application model to another, mapping technologies that are integrated in EXPRESS data definition language (Amor and Hosking 1994; Verhoef *et al.* 1995; van Leeuwen 1999).

Common core model

Representing the approach to data integration, this architecture truly integrates data models into a common core or kernel model. Core model consists of data definitions that are independent of the associated application domains. The data definitions in the common core are usually industry-wide accepted as being relevant for common use. This approach offers advantages over other architectures by allowing inter-application interactions with a minimal loss of semantics. Data defined outside the common core model cannot be communicated still without the loss of semantics. Any update of the common core model also requires all applications to be updated. Although this architecture is mostly evident in closely integrated systems from one or a group of vendor, implementation of this architecture can also be found in open integrated systems. IFC, a significant step forward for the AEC industry is based on the kernel/schema establishing the root information for all leaf node classes (Liebich *et al.* 2004).

STEP (STandard for the Exchange of Product Model Data)

STEP, also known as ISO 10303 (ISO 1994a) is an international product data standard to provide a complete, unambiguous, computer-interpretable definition of the physical and functional characteristics of a product throughout its life cycle; developed by ISO-TC184-SC4 sub committee. The mission of SC4 has been stated in ISO (ISO 2005) as *"to develop and promulgate standards for the representation of scientific, technical and industrial data, to develop methods for assessing conformance to these standards, and to provide technical support to other organisations seeking to deploy such standards in the industry"*. STEP is targeted at the exchange of data describing a product between CAx and PDM systems; and storage of data in repositories throughout the lifecycle (Moeller 2000). It is a much broader standard than previous data exchange standards such as IGES, *Standard d'Echange et de Transfert* (SET) and *Verband Der Automobilindustrie Flachen Schnittstelle* (VDA-FS).

Structure of STEP

Fundamental to STEP approach is the layers segregating the logical structure of the information from the format in which it is carried on in some media. The 'layering' was inspired by the work of ANSI (*American National Standards Institute*) on database architectures and standards (Tischritzis and Klug 1978). The work distinguished between database definition and implementation, defining a three-layer architecture of abstractions and mappings. The three layers are (see Figure 3.10):

- Physical: defines the physical structure of data on disk or other media,
- Logical: defines the information of interest in an implementation-independent logical structure; the logical level maps to the physical level, and
- Application: specifies the information needed by a specific application and its format; the application level is implemented on top of and is derived from the logical level.

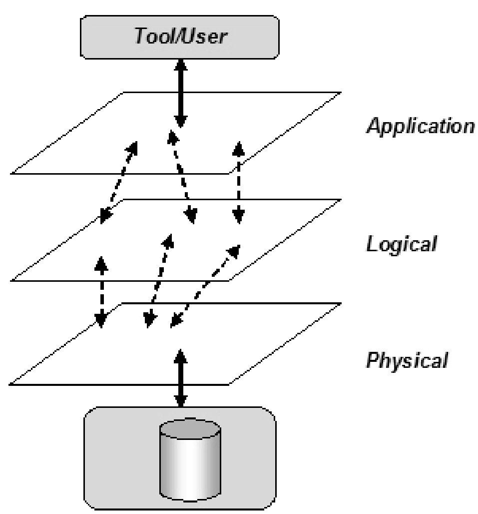


Figure 3.10: ANSI layers of abstractions and mappings.

The layering or separation was a fundamental idea in the STEP approach. It separates an application level from the logical level, suggesting a way to define subsets of a complete model. Information modelling methods are used to specify required conceptual structure of the information to be represented. IDEF1x (Bruce 1992), *Natural Language Information Analysis Method* (NIAM) (Nijssen and Halpin 1989) and EXPRESS-G (ISO 1994b) are the information modelling techniques initially accepted by the standardisation committee for this purpose. Later IDEF1x and NIAM have been discarded as modelling techniques because of the reasons stated in section 0. In addition, an intermediate-level specification language for defining the logical structure of a model, separate from its physical implementation has been developed. The resultant description method is known as EXPRESS (ISO 1994b) language.

The specification for STEP is structured in different sections with the following numbering methodology, details of which are shown in Figure 3.14 (Neil 2001):

Part 1 - Overview and fundamental principles

Parts 11-13 - *EXPRESS*

Parts 21-30 - Implementation methods (SPF, SDAI, C++, XML, Java)

Parts 31-35 - Conformance testing methodology and framework

Parts 41-58 - Integrated-Generic Resources

Parts 101-110 - Integrated-Application Resources

Parts 201-240 - Application Protocols (APs)

Parts 501-523 - Application-Interpreted Constructs

Parts 1001-1514 - Application Modules

Conventionally, information models were developed as a closed loop where a complete information model was first developed then subsets were taken. STEP development is based around various part models, also known as APs (Application Protocols), with the expectation that the APs would later be re-established as larger domain specific models. The incremental development approach of STEP has significant advantages over previous efforts as existing applications with rich semantic content can be clustered into specialised domains before a complete model can be developed. This has been particularly useful to larger industries like the AEC. The STEP system architecture identifies five classes of tools (Figure 3.11) (Eastman 1999):

Description methods

Description methods are the information modelling languages employed in specifying the information models used in the architecture, i.e., to define the Integrated Resources, Application Reference Models and Application Interpreted Models. The formal description methods include

EXPRESS and EXPRESS-G. IDEF1x and NIAM diagrams were popular with the early STEP modellers, but had been proved unacceptable for several reasons. Neither of the languages was an international standard and could not be normatively referenced by the STEP standard. It was seen as difficult to drive either IDEF1x or NIAM through the standardization process because other groups had ownership of them (Loffredo 1998).

Integrated resources

Application models defined by the common model subsets are known as Integrated Resources. Models used in different domains are called Integrated-Generic Resources. They include geometry, material properties and project classifications - i.e., items that can be shared across multiple application domains. Model subsets that are industry specific are called Integrated-Application Resources. These include subsets for electronics, drafting, kinematics, finite elements and building. Presentation formats are called application-interpreted constructs.

Application protocols

Using the description methods and integrated resources, Application Protocols or APs are developed that describe the scope of the information required and understood by the application in context (Mason 2002). An application protocol is partitioned into two aspects: an *Application Reference Model* (ARM) and an *Application Interpreted Model* (AIM). ARM represents requirements for an application in easily understood form by domain experts to design and assess the information model. Graphical languages such as EXPRESS-G are used for defining ARMs. An ARM is interpreted into an AIM, defined using EXPRESS which can be consumed by both computers and humans. The AIM resolves all uses of the Integrated-Generic Resources and integrates the model with Integrated-Application Resources.

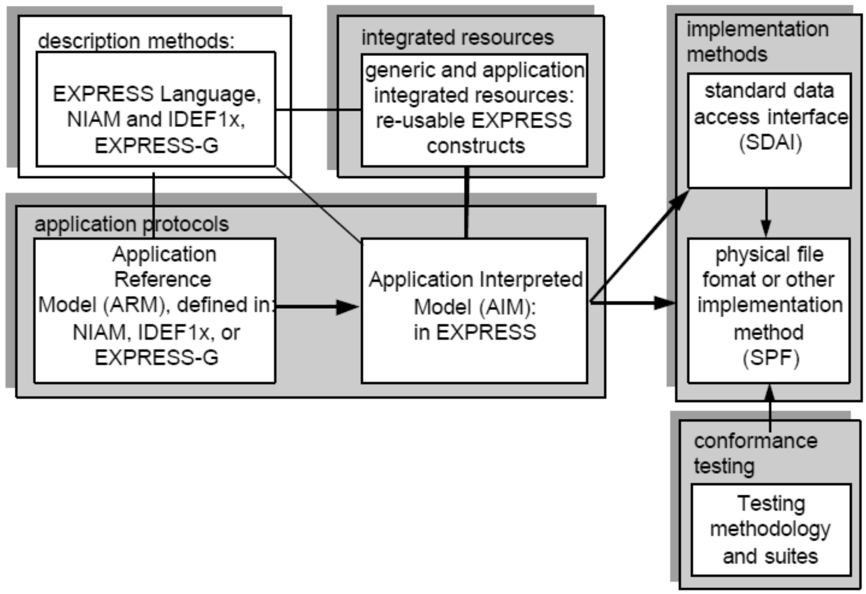


Figure 3.11: Diagrammatic representation of the different parts of the STEP.

Implementation methods

An application protocol is combined with an IM (Implementation Method) to form the basis for a STEP implementation. An implementation method typically includes multiple resources. *STEP Physical File* (SPF) (ISO 2002) and the SDAI (ISO 1998b) are the most commonly used implementation methods. Bindings to SDAI (Part 22) has been developed for C++ (Part 23), C (Part 24) and Java (Part 27) programming languages.

Conformance testing

A Conformance Test assesses the implementation in terms of its ARM and AIM and confirms that the STEP languages and tools have been properly used and interpreted. A Conformance Testing methodology is applied by accredited organizations. The testing includes application and interpretation of test suites.

EXPRESS modelling language

Also known as part 11 (ISO 1994b) in the STEP specification, EXPRESS was developed for representing application interpreted models. It started as a US Air Force project named *Product Data Definition Interface* (PDDI); received its current name in 1986 and became international standard in 1994. EXPRESS consists of language elements to define unambiguous data and to specify constraints on the data defined. EXPRESS represents knowledge embedded in a product model in an implementation independent manner, as interpreted from an ARM to take advantage of shared

Integrated Resources. EXPRESS is used in the same way data structures are defined in a programming language although there are certain dissimilarities. It specifies how instances of defined objects will be organised for use. Data models can be defined with EXPRESS in two ways: textually and graphically. Graphical EXPRESS is known as EXPRESS-G - uses graphical constructs to define entities and relationships. EXPRESS-G is often more suitable for human consumption and not able to represent everything that can be defined using the textual form, EXPRESS. The strengths of EXPRESS are summarised by Choo (Choo 2004) as:

- the language can be used to describe constraints as well as data structures and relationships. These constraints form an explicit correctness standard for an information model,
- EXPRESS models can be consumed and processed by computers; hence software is able to take advantage of the definitions without human transcription, and
- EXPRESS has undergone the international standardisation process, which represents a significant consensus that the language meets the needs of the industry.

In EXPRESS, a schema defines the UoD (Universe of Discourse) in which declared objects are given mutually dependent meanings and purposes. Schemas include definitions of things (entities, types, functions and procedures), rules defining relationships between things and rules on relationships. EXPRESS, a block-structured language like Pascal or C, includes a procedural language syntax for specifying rules. All types have scopes that are identified by the block in which they are defined. A block begins with the declaration of an Entity, Function, Procedure Rule or Schema and ends at the end of Entity, Function, Procedure Rule or schema. When an identifier in one block is redefined in an inner block, the inner block declaration overrides the outer one, for the extent of the inner block. Multiple schemas may define an information model. Special mechanisms exist to make cross references across schemas. Using syntaxes such as `USE` and `REFERENCE`, one schema can be linked to the other. An overview of EXPRESS-G notations is shown in Figure 3.12 and an example schema *family* is shown in both EXPRESS and EXPRESS-G in Figure 3.13[6]. Further details on EXPRESS *types*, *constructors*, *entities*, *attributes*, *rules* and *programming constructs* can be found in Appendix D.

[6] Source: http://en.wikipedia.org/wiki/EXPRESS_%28ISO_10303-11%29

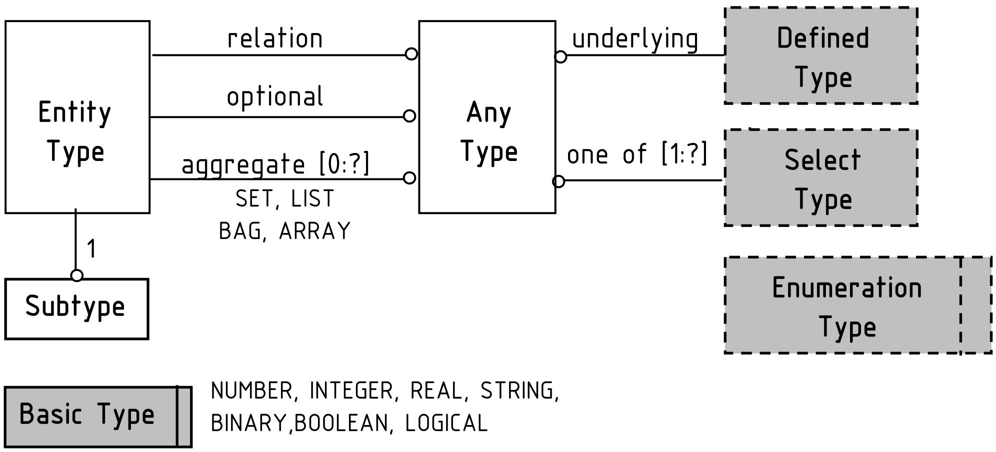
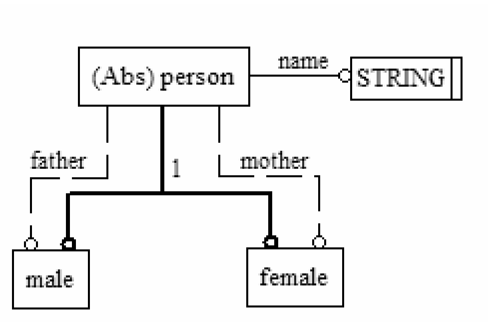


Figure 3.12: Overview of EXPRESS-G notations.

```
SCHEMA Family;
ENTITY Person
   ABSTRACT SUPERTYPE OF
(ONEOF (Male, Female));
     name: STRING;
     mother: OPTIONAL Female;
     father: OPTIONAL Male;
END_ENTITY;
ENTITY Female
   SUBTYPE OF (Person);
END_ENTITY;
ENTITY Male
   SUBTYPE of (Person);
END_ENTITY;
END_SCHEMA;
```

Schema 'Family' in EXPRESS



Schema 'Family' in EXPRESS-G

Figure 3.13: An example schema in EXPRESS and EXPRESS-G.

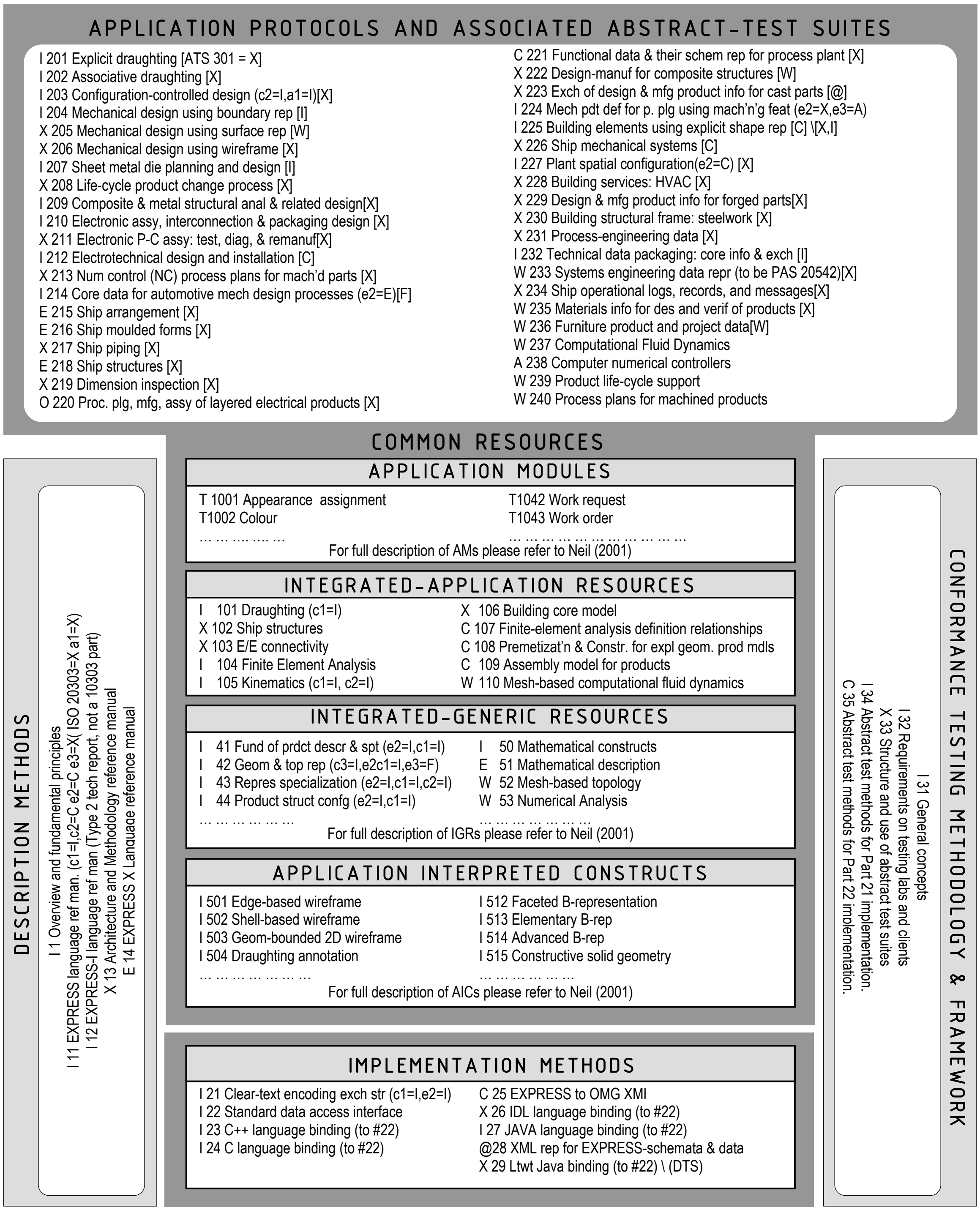


Figure 3.14: STEP framework.

SPF (STEP Physical File)

Clear text encoding of the STEP exchange structure is known as STEP Physical File or SPF, defined originally in ISO 10303-21:1994 (ISO 1994c). The second edition, ISO 10303-21:2002 (ISO 2002) includes all the fixes and extensions for several data sections of ISO 10303-21:1994. SPF is also known as p21 file, the name derived from the part number of the ISO 10303 specifications. The structure of SPF is based on ASCII and is easy to read with typically one instance per line. ISO 10303-21 does not define the EXPRESS schema but the encoding mechanism on how to represent data according to a given EXPRESS schema. Part 21 also defines two conformance classes based on how complex entity instances are encoded. Conformance class 1 is more compact of the two where data is defined using internal mapping. Conformance class 2 uses external mapping and said to be theoretically better than the former as it allows AP interoperability; since a pre-processor may know how to handle some supertypes - not used in practice. Further details on the structure of SPF can be found in Appendix E.

Industry Foundation Classes

Industry Foundation Classes (IFC) is an information standard for the AEC/FM (Architecture, Engineering, Construction and Facilities Management) industry, developed by *International Alliance for Interoperability* (IAI). IAI's vision for IFC development is "*to improve communication, productivity, delivery time, cost and quality throughout the whole building life cycle by providing a universal basis for process improvement and information sharing in the AEC/FM industry*" (IAI 2005a). IAI aims to build on the collective knowledge of the global construction and facilities management industries to define IFC.

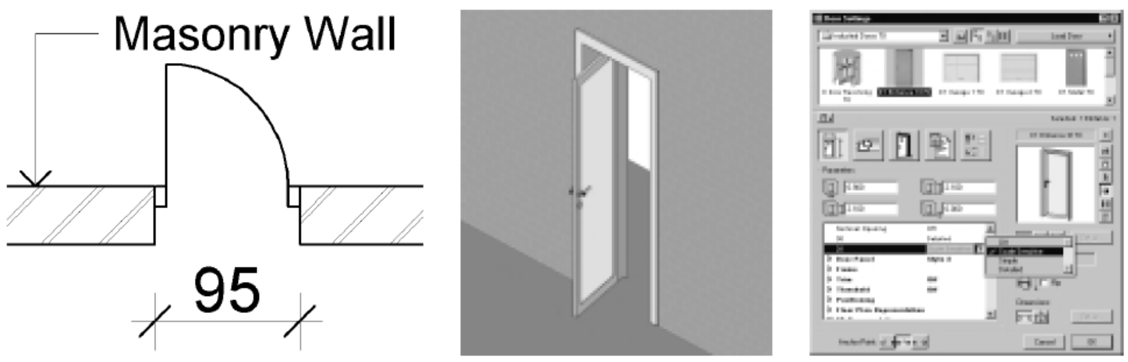
Background of IFC

IFC is based on the idea of digital representation of objects that occur in a constructed facility. The objects include real things such as walls, windows, doors, etc. and abstract concepts such as space, organisation, process, etc. These specifications represent a data structure supporting a digital project/product model for use in data exchange and sharing across applications and domains throughout the lifecycle stages. Each specification is referred to as a *class* which is used to describe a range of things that have common characteristics (BRE 2005). For instance, every door has the characteristics of opening to allow entry to a space; every window has the characteristic of transparency so that it can be seen through. Door and window are objects and names of classes. The classes defined by the IAI are termed *Industry Foundation Classes* or IFCs. The reasons for this are:

- IFC is defined by the AEC/FM industry,
- they provide a foundation for the shared project model, and

- they specify classes of things in an agreed manner that enables the development of a common language for construction.

Objects specified in IFC are more than simple collections of lines and geometric primitives to recognise them visually. Specifications of objects contain attributes and characteristics that make them. IFC-based objects not only allow stakeholders to share a project model but also define specialist views of the objects contained in that model. This leads to improved efficiency and enhanced interoperability as one object defined at an early stage can be reused during latter stages of the lifecycle; avoiding data redundancy. This shared data among project participants can continue to evolve after design, through construction, and occupation of the building. Figure 3.15 shows the difference between a product model and a graphic entity (DXF, DWG, DGN, etc.) using IFC representation in ArchiCAD (Graphisoft 2001).



Graphic entity, drafting by computer | Object model | Entity data

Figure 3.15: Graphic entity vs. product model.

Part 11 of ISO STEP specification (EXPRESS) has been adopted as data definition language for IFC model. EXPRESS-G provides graphical representation of EXPRESS as in ISO STEP, which is readily accessible and easily understandable. IFC data can be encoded to a Part 21 file or SPF according to Part 21 specification of ISO-STEP. Data can also be shared and accessed using SDAI repository. Thus, the IFC data model corresponds with the STEP standard and consequently contributes to the evolution that permits the exchange of building data between different programmes. Unlike STEP, evolution is not planned for a norm, but aims at direct application in the industry (Choo 2004); e.g. IFC has a much broader scope than AP 225 (Building elements using explicit shape representation) of the STEP. Details on IFC specification development process can be found in Appendix F.

Structure

IFC architecture is divided into four conceptual layers, namely *domain layer*, *interoperability layer*, *core layer* and *resource layer*. Each layer consists of modules containing reusable constructs or classes, and it is within each schema that the individual entities are defined. For instance, the Wall entity (called `IfcWall`) falls in the Shared Building Elements schema, which in turn belongs to the interoperability layer. The layered architecture allows an entity at a given level to relate to or reference an entity at the same level or at a lower level, but not an entity at a higher level.

Figure 3.16 shows different layers of IFC architecture. The modular design of the overall architecture makes it easier to maintain and extend the model, allowing lower-level entities to be reused in higher-level definitions. It also makes a clearer distinction between AEC/FM disciplinary entities so that the model can be more easily implemented in individual discipline-specific applications.

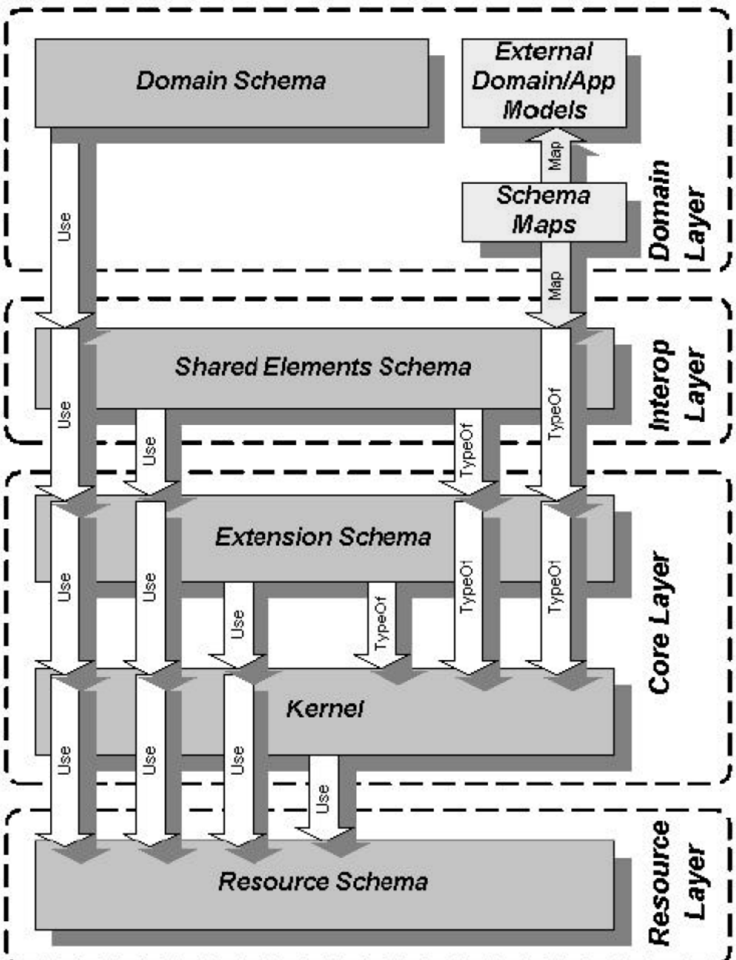


Figure 3.16: Layered architecture of IFC, after (Liebich and Wix 2000).

Four layers of IFC are described here (Khelmani 2004):

Resource Layer: Entities in this layer represents basic properties such as geometry, material, quantity, measurement, data and time, cost, etc. These are generic and not specific to buildings and function as resources that are used in defining the properties of entities in layers above.

Core Layer: This layer contains entities that represent non-industry and industry-wide specific, but abstract concepts that are used to define entities in the higher layers. For instance, the Kernel schema defines core concepts such as actor, group, process, product, relationship, and so on, which are used in all the higher-level entities of the model. The Product Extension schema defines abstract building components such as space, site, building, building element, annotation, etc. The other two Extension schemas define process and control related concepts such as task, procedure, work schedule, performance history, work approval, and so on.

Interoperability Layer: This level comprises entity categories that are commonly used and shared between multiple building construction and facilities management applications. Thus, the Shared Building Elements schema has entity definitions for a beam, column, wall, door, etc.; the Shared Building Services Elements schema defines entities such as a flow segment, flow controller, fluid flow properties, sound properties, etc.; the Shared Facilities Elements schema has entity definitions for an asset, occupant, and furniture type; and so on. Most of the common building entities would be defined in this layer.

Domain Layer: The highest level of the IFC model contains entity definitions for concepts specific to individual domains such as architecture, structural engineering, facilities management, and so on. Examples include a space program for architecture; footing, pile, and plate entities for structural engineering; boilers, chillers, and coils for HVAC, etc.

An example of specialised extensions such as `IfcLanding`, specific to disciplines, in this case architectural design, are derived from lower classes and further distinguished through property sets. Figure 3.17 shows an annotated inheritance tree for `IfcLanding` (Owolabi *et al.* 2003). IFC 2x2 architecture in shortform distribution is shown in Figure 3.18.

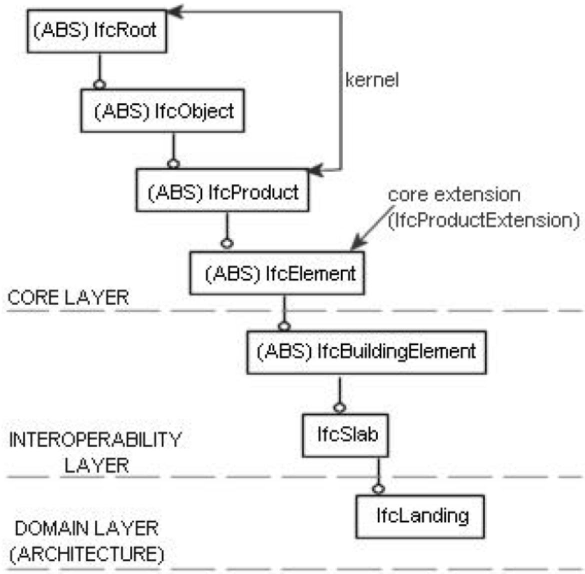


Figure 3.17: Inheritance tree for IfcLanding.

Implementation

IFC can be implemented in one of the three ways shown in Figure 3.19:

- By using Part 21 physical file. Information can be shared across a network, by email or on a physical medium such as CD ROM. The EXPRESS language specification (ISO 10303:11) view of the IFC Object Model determines the structure of the file and part 21 (ISO 10303:21) determines the syntax of the file. Most IFC enabled software applications share information using physical files.
- By placing information in database which has an interface defined according to ISO 10303:22 (ISO 1998b), otherwise known as SDAI (Standard Data Access Interface) for putting in and getting out data. The EXPRESS language specification view of the IFC Object Model determines the structure of the information stored in the database.
- By using software interfaces that can expose the information content of defined groups of attributes within an object. Software interfaces allow for direct communication between applications without the need to translate application model into an intermediate file or a database.

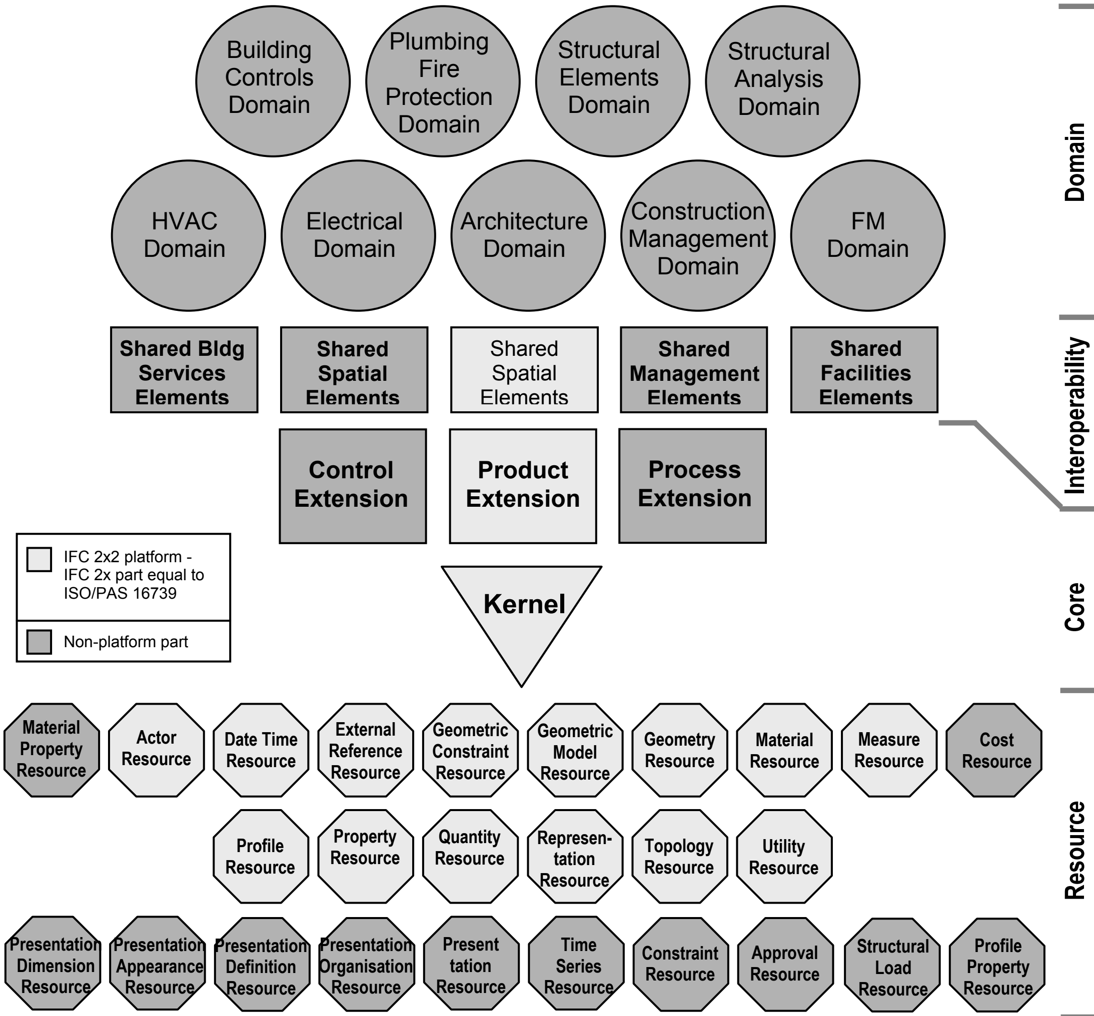


Figure 3.18: IFC 2x2 architecture, shortform distribution, after (IAI 2005b).

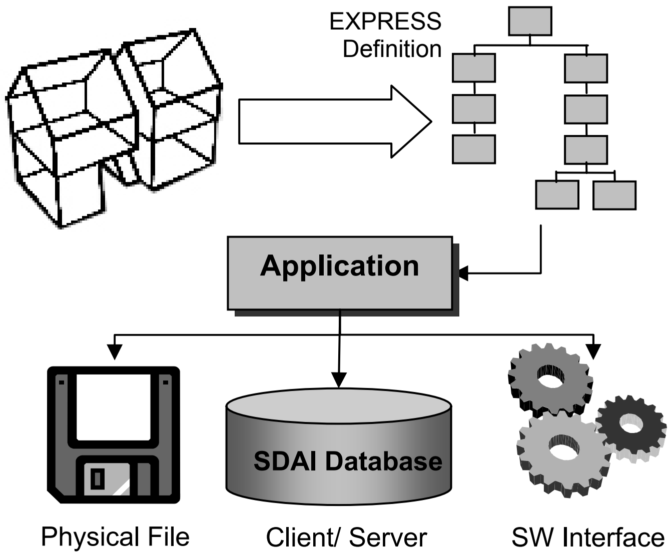


Figure 3.19: Implementation methods for IFC, after (Mourshed et al. 2003b).

Contemporary issues in BIM implementation

Although Information Modelling in the AEC industry has come a long way since proprietary file formats such as DWG and DGN, the developments in product modelling have not changed the fragmented industry the way it was envisaged in the 1990s. The industry's reluctance to embrace new technologies has always been suggested by researchers as the primary factor behind the lack of integration (Egan 1998; Dawood *et al.* 2002; Owolabi *et al.* 2003). This statement overlooks the factors related to work practice, regional diversity, and software vendors' unwillingness to promote open standards, among others.

Proprietary systems

The objectives of IFC development was to enhance interoperability through agreed upon data exchange standards. The way IFC has been implemented by vendors does not guarantee the maintenance of semantics. Available CAD software is still based on proprietary information model requiring internal mapping to export data to the IFC format. Internal mapping is not always flawless where the translation involves a large number of classes, which may or may not be available in either the source or the target system. For example, when building information is exported to IFC format during concept development using ArchiCAD IFC add-on, materials data is not transferred which may be required by processes in latter lifecycle stages. If data from some upstream application contained semantics not available in ArchiCAD, the latter update from ArchiCAD may well render the data redundant.

The full potential of IFC are never realised due to the fact that major CAD vendors have commercial interests in keeping their proprietary file formats rather than promoting a public industry standard. The consensus on the ways IFC standard will be implemented has not been reached; e.g. `IfcSpace` boundary is represented differently in the IFC file exported from ArchiCAD and ADT, creating incompatibility that is imposed by the vendors, not the standard itself. It has been argued that a large-scale adoption of IFC standard in the industry will eventually influence CAD vendors to move towards public standardisation. Examples of public standardization driving major software vendors to either develop software based on industry specification or make their specification publicly available can be found in Office Applications. Due to the fact that several EU states opted for OpenDocument in eGovernment applications, Microsoft initiated the release[7] of some of their proprietary file specifications in the public domain.

Complexity of a single product model

IFC is an ambitious standardisation effort requiring stakeholders from different disciplines to agree on the proposed standard before it can be published for implementation. The time required by the development process deters frequent upgrade of the specification to keep up with the changes in technology and industry practices. On the other hand, a stable release is necessary to allow software vendors to implement and provide IAI with feedback and proposal for changes. This is why a new release of IFC will be delayed to allow vendors to invest in implementation and to establish a critical mass of followers. Building design is a creative problem solving activity where value is added by being more creative. Information models developed to support such activity must not inhibit creativity. On one hand, product model needs to be able to evolve by incorporating new definitions or concepts; on the other hand, they need to remain comprehensible to other software applications by adhering to a specification. This conflicting requirement is usually resolved by trading off flexibility with the rigidity of the specification.

XML vs. EXPRESS

The rapid proliferation of the Internet and the developments in XML and web services prompted the debate whether the AEC industry should begin developing XML based standards. XML proposes to simplify communication between systems and to speed up the retrieval of data - contained in a web page, other structured data or programming code - from different systems (Hunter 1999). XML is being hailed as the technology that will take the Internet to a new level and make it truly suited to business; semantic web is an example of such efforts where homogenisation of information is sought in the sea of unstructured World Wide Web. As a subset of SGML, XML is intended to make it easy to interchange structured documents over the Internet. XML differs from SGML primarily in

[7] Microsoft submitted 12 XML specifications related to MS Office applications to ECMA International for approval as an open standard in 2005.

simplifying the sometimes intimidating formalisms of SGML, in order to ensure that an XML parser is simple enough to embed in even lightweight software, including Web browsers (Underwood and Watson 2003). The practical benefits of XML implementations are:

- *Structure*: to model data to any level of complexity,
- *Extensibility*: to define new tags as needed,
- *Validation*: to check data for structural correctness,
- *Media independence*: to publish content in multiple formats, and
- *Vendor and platform independence*: to process any conforming document using standard commercial software or even simple text tools.

The XML vs. EXPRESS debate originated when researchers started looking at XML as a replacement for EXPRESS. XML is being widely used in the enterprise sector, particularly for business transactions, because of the ubiquity of the number of toolkits and applications supporting the technology. It has been observed that a greater number of programmers are already more familiar with XML rather than EXPRESS, which happens to be used only in niche areas within the industry. In XML, information can be modelled using a number of specifications such as XML Schema (Fallside and Walmsley 2004), RELAX NG (Clark and Makoto 2001), Schematron (Jelliffe 2005) and XSLT (Clark 1999). EXPRESS can represent complex inheritance relationships and functions, and includes a rich set of constructs for specifying constraints on populations of instances. Moreover, STEP data (i.e. an instance population of an EXPRESS schema) are typically exchanged using an ASCII character based syntax defined in ISO 10303:21. The Part 21 syntax, although adequate for the task at hand, lacks extensibility. Part 21 is also hard for humans to read, and – perhaps the most limiting - is computer interpretable only by software supporting STEP. Numerous efforts at developing XML standards, mainly vendor led began in the latter half of 1990s, but none gathered much of a following. Subsequently these standards were incorporated into IAI. Most important of these efforts were aecXML and ifcXML. aecXML is an effort independent of IFC development, whereas ifcXML is a part 28 (ISO 2003a) representation of the IFC standard.

ISO 10303:28: XML representation of EXPRESS

ISO 10303:28 also known as part 28 specifies means by which schemas specified using EXPRESS and data governed by EXPRESS schemas can be represented as an XML document. For the representation of EXPRESS schemas, part 28 specifies an XML markup declaration set based on the syntax of the EXPRESS language. The specification also supports EXPRESS text representation of schemas. To represent data corresponding to an EXPRESS schema, ISO 10303:28 takes two broad approaches: *late binding* and *early binding*. Where a single markup declaration set independent of the EXPRESS schema is specified it can represent data for any schema. This approach is called *late binding*. *Early binding* refers to the specification of the results of the generation of a markup declaration that is dependent on a specific EXPRESS schema. The markup declarations can be specified either in DTD or in XML Schema. The following are within the scope of part 28 specifications:

- specification of XML markup declarations that enable EXPRESS schemas to be represented using XML,
- specification of a single XML markup declaration set that is independent of the EXPRESS schema and formally describes the XML representation of data governed by any schema; also known as late binding,
- for an arbitrary EXPRESS schema, specification of an XML markup declaration set that corresponds to the schema and formally describes the XML representation of data governed by that schema; also known as early binding,
- specification of the mapping between XML markup declarations corresponding to a specific schema and the XML markup declarations independent of any schema.
- specification of the form of XML documents containing EXPRESS schemas and data governed by EXPRESS schemas, and
- specification of the representation of EXPRESS primitive data type values as element content and as XML attribute values.

ifcXML

ifcXML is being developed by IAI and covers the methodology for generating the XML schema out of the IFC EXPRESS definition. It also specifies how the XML schema and the corresponding XML will be written. It is anticipated that by offering an XML representation of IFC data, a broader community of applications will be able to access a unified schema representing the built environment and related resources. Business motivation behind the development of ifcXML was that XML has a broader range of supporting utilities and database implementations and is the basis for most eCommerce messages and Web services; XML is also supported by some web browsers especially with XSLT style sheets, making the information immediately accessible on workstations and most other handheld computing devices (Nisbet and Liebich 2005a). It was decided to adopt IFC as the baseline standard because IFC is an internationally agreed upon standard for the AEC/FM industry that has already been widely tested in several domains using other representations. ifcXML will have a role in application areas such as mapping between the IFC object model and:

- document based representations such as schedules, quantity take-offs and product datasheets,
- message based representations such as RFI’s (Request-For-Information), orders and other eCommerce communications, and

- communication with XML based domains, such as the GIS object models based on the gml3 standards.

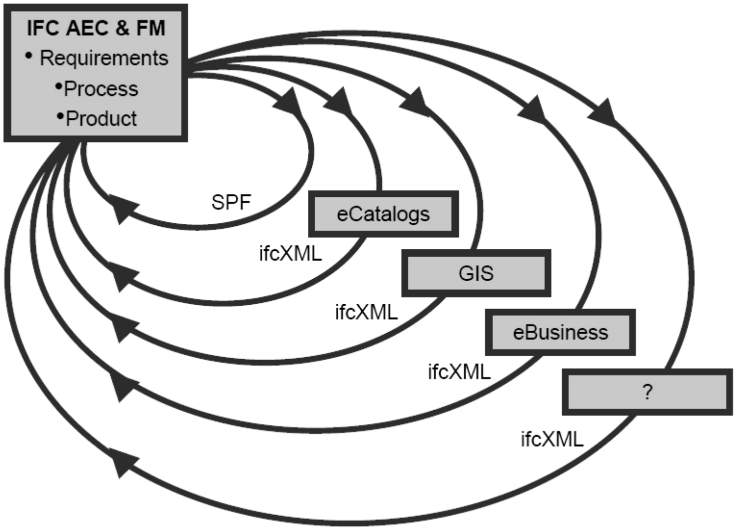


Figure 3.20: Anticipated role of ifcXML and SPF, after (Nisbet and Liebich 2005a).

The ifcXML representation is also anticipated to facilitate the retrieval, transmission and merging of partial models during AEC/FM processes in a collaborative environment. ifcXML application contexts are shown in Figure 3.20. It is envisaged that ifcXML will be used as part models for file-based transfer and database implementation of the resulting XML will be based on Express-X, SQL and XSLT. SDAI will be used for server based IFC transactions and SPF will dominate file-based exchange of IFC data as shown in Figure 3.21.

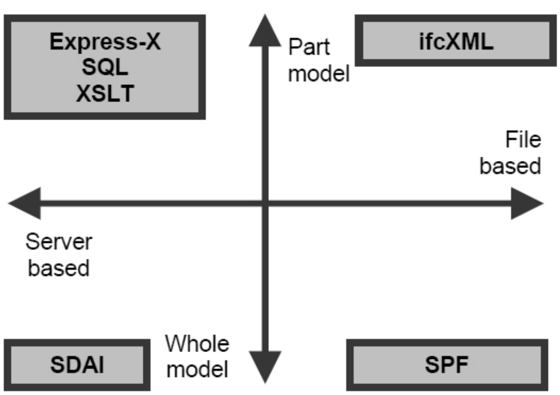


Figure 3.21: ifcXML and alternatives.

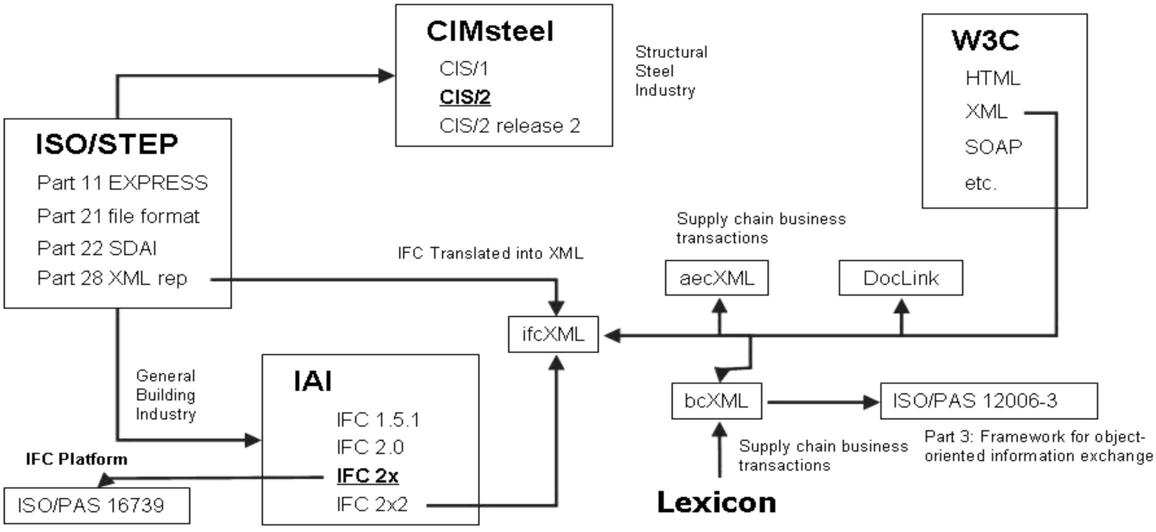


Figure 3.22: Complementary technologies, enhancing interoperability.

IFC and ifcXML are seen as complimentary technologies supporting each other to enhance interoperability in the AEC/FM industry. Applications supporting ifcXML are not envisioned to replace those supporting IFC. XML and IFC as complementary technologies enhancing interoperability are described in Figure 3.22. Potential applications could benefit from the use of the ifcXML representation include *product and material libraries*, *briefing, maps and GIS context, eCommerce messages, asset registers, BMS*, and *cost applications*.

```
Entity IfcProperty
ABSTRACT SUPERTYPE OF (ONE OF (IfcComplexProperty,IfcSimpleProperty));
   Name : IfcIdentifier;
   Description : OPTIONAL IfcText;
END_ENTITY;

<xs:element name="IfcProperty" type="ifc:IfcProperty" abstract="true"
 substitutionGroup="ex:Entity" nillable="true"/>
<xs:complexType name="IfcProperty"
 Abstract="true">
 <xs:complexContent>
  <xs:extension base="ex:Entity">
   <xs:sequence>
    <xs:element name="Name" type="ifc:IfcIdentifier"/>
    <xs:element name="Description" type="ifc:IfcText"
     nillable="true" minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

Figure 3.23: EXPRESS schema and corresponding XML schema generated by ISO 10303:28.

To describe the structure of ifcXML files, XML schema (XSD), the most widely used schema definition language, is used. The XSD is automatically generated from the IFC EXPRESS schema longform in compliance to ISO 10303:28 edition 2. An example EXPRESS schema and the corresponding XML schema generated by ISO 10303:28 is show in Figure 3.23. Part 28 edition 2 provides a mapping between any EXPRES schema and an XSD; the semantics of the resulting XSD conforms to the original EXPRESS schema.

Brief overview of Interoperability-based frameworks

Many researchers have attempted at integrating interoperability standards in industry-wide collaborative environments applying the principles and concepts described in previous sections in this chapter. Early studies (Syal *et al.* 1991; Aouad *et al.* 1994; Alshawi *et al.* 1997) into integration suggested almost similar solutions involving bringing together various design and construction functions such as project design, estimating, construction management into a single computer environment - in other words, development of an integrated construction environment. Examples of integrated construction environments include, but are not limited to: ATLAS (Greening and Edwards 1995), COMBINE (Augenbroe 1995), RATAS (Björk 1994), ICON (Aouad *et al.* 1994), COMBI (Scherer 1995), SEMPER (Lam *et al.* 2004), WISPER (Faraj *et al.* 2000), GBP (Keane and Kelliher 2001), OSCON (Aouad 1997), OPIS (Froese and Paulson Jr. 1994), SPACE (Alshawi *et al.* 1997), ToCEE (Amor *et al.* 1997). Although having similar objectives, their implementation method varied considerably; from the implementation of point-to-point data translators to neutral data standards and integrated project databases. Integrated project database usually acted as a central repository of information related to the building project. The following sections contain descriptions of some of the previous efforts at developing interoperable systems.

WISPER

WISPER (Web-based IFC Shared Project EnviRonment) has been developed at the University of Salford, UK as a collaborative working environment for construction. The architecture of WISPER uses a three-tier client-server infrastructure (see Figure 3.24) to demonstrate the integration between detailed design, building element based cost estimating, and construction scheduling, in addition to a VRML viewer that allows the graphical querying of a project database. Each tier of the application, i.e., *presentation*, *logic* and *data storage* is isolated to provide benefits such as maximum control, scalability and flexibility. The project builds on previous research projects at Salford: SPACE and OSCON. WISPER enabled users to exchange project information through STEP Part 21 file and/or share the information through IFC database. Faraj (Faraj *et al.* 2000) stressed the need for process-based model for the full implementation of such environments.

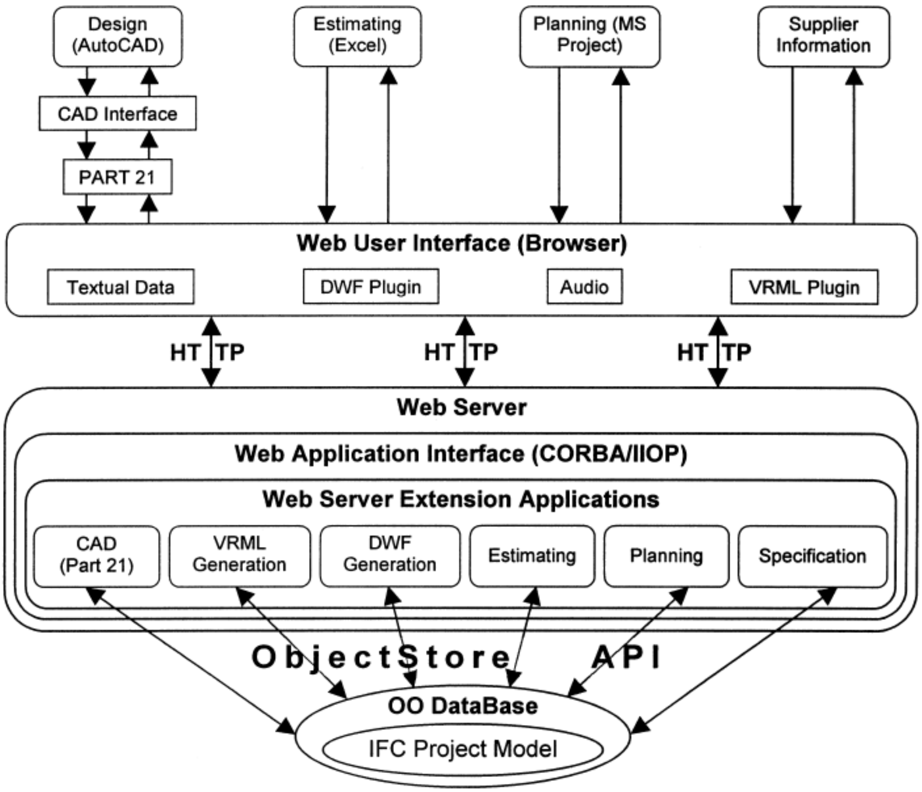


Figure 3.24: Overview of WISPER, after (Faraj et al. 2000).

SEMPER

SEMPER (Simulation Environment for Modelling PERformance) is an active, active multi-aspect prototype design and simulation environment that incorporates an object-oriented, space-based design tool, with dynamic links to different building performance evaluation tools (Mahdavi 1996; Lam *et al.* 2004). Building on SEMPER prototype, SEMPER-II provided an internet-based computational environment supporting geographically distributed users in collaborative performance-based building design. The building representation in SEMPER-II is defined as the SOM (Shared Object Model), which is a hierarchically structured template to capture the essential elements of a building and their properties, to the extent required by the simulation applications in the S2 environment (Mahdavi *et al.* 1999) and implemented as a *universal building model*. SOM contains a tightly structured notation of constitutive building elements, with pointers to the detailed information. For each disciplinary domain, the simulation application's representation, DOM (Domain Object Model) is created through filtration and modification of information in SOM according the specific view of the building in that domain (Lam *et al.* 2004). Within the framework of the SEMPER project, a functional homology-based SOM-2-DOM mapping technology has been implemented that uses the configurational isomorphism between SOM and various DOMs to derive the latter automatically from the former (Mahdavi *et al.* 1999). Figure 3.26 shows Homology-based mapping of the space-based architectural representation (left) to the nodal building representations in SEMPER's thermal and air flow simulation modules (middle) with their corresponding simulation results (right). SEMPER-II Architecture is illustrated in Figure 3.25.

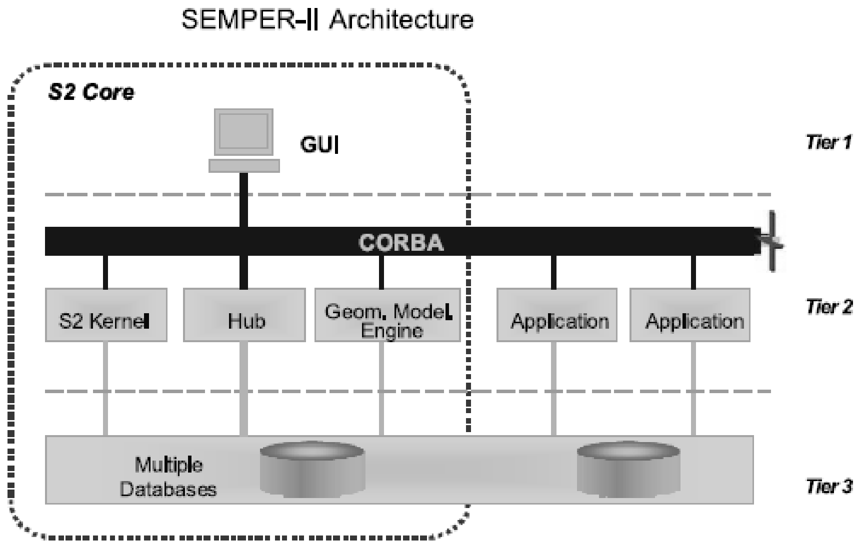


Figure 3.25: SEMPER-II Architecture, after (Lam et al. 2004).

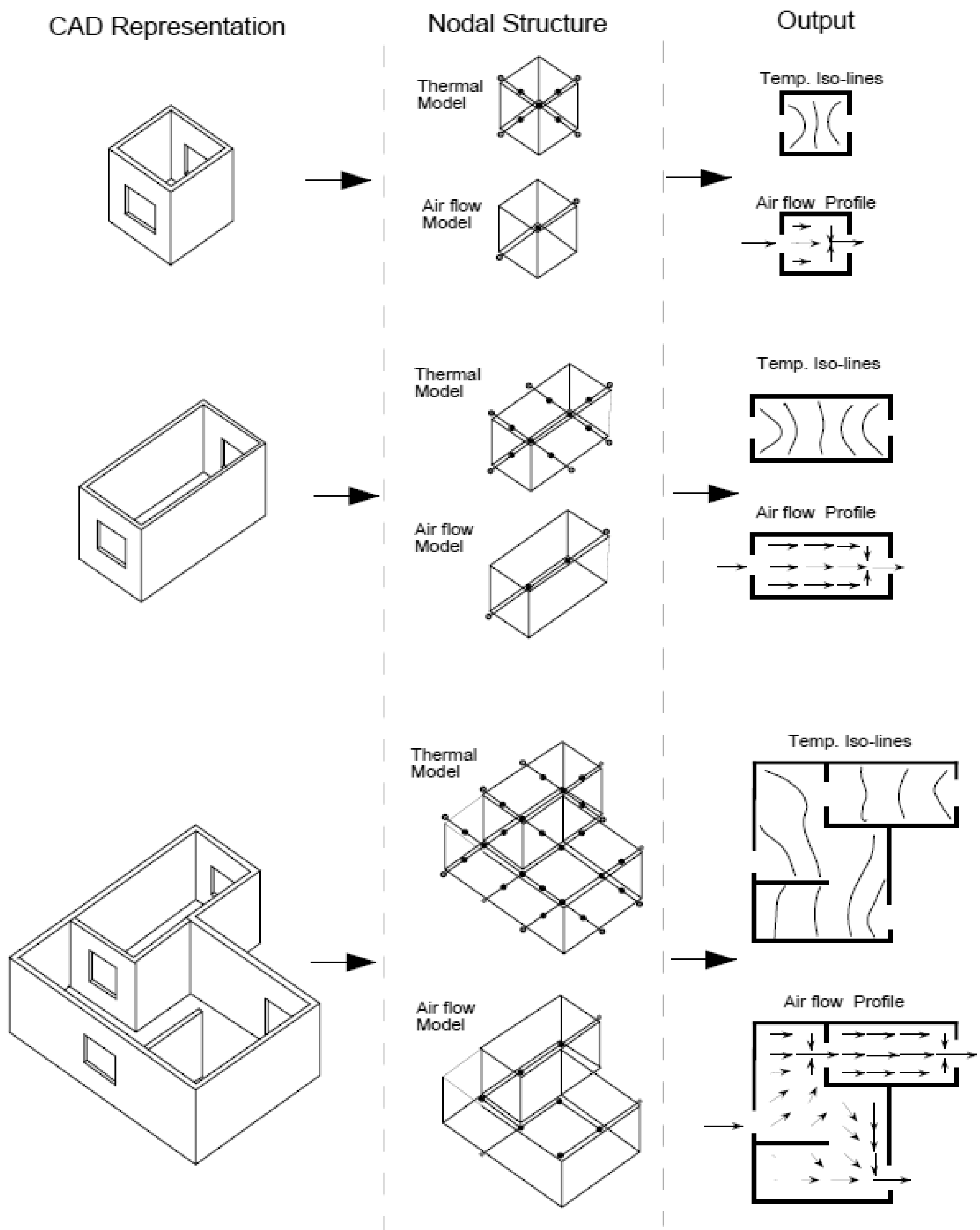


Figure 3.26: Homology-based mapping of the space-based architectural representation, after (Mahdavi et al. 1999).

Remarks

Among the previously developed systems, those with integrated project databases incorporated data storage and manipulation differently. In most of the cases, a multi-tiered architecture separated data representation and access mechanisms. CORBA (Common Object Request Broker Architecture) was implemented in some of the systems to allow distributed data repositories. A common data definition was chosen at the beginning for integration. After the formation of IAI, IFC was chosen for industry wide interoperability. Further discussions on integrated project databases is done by Amor *et al.* (2001).

Debate over applicability of a single model across the industry vs. multiple models can be found in (Amor and Faraj 2001), but directions on how multiple models can coexist were not provided. On the contrary, the existence of multiple models, even project specific ones are considered in the IFC specification. Property Sets, allow incorporation of models not defined in the IFC, pluggable into the interoperability layer. Haymaker *et al.* (Haymaker *et al.* 2004) argued that current project modelling approaches lacked *adequately simple, formal, generic expressive methods that engineers can use to automatically construct a new dependent view from information in one or many source views* - implying the need for an added layer over existing single or multiple data models. Their research resulted in *perspectors* - reusable reasoning modules which engineers can use to automatically construct a task-specific engineering view, called a *perspective*. Albeit similar arguments for task centric solutions can be seen in (Augenbroe *et al.* 2004) where Augenbroe et al. opted for a *language* to express both *analysis requests* and the *answers* that are generated by experts responding to these requests. Experience gathered from this research suggests both of these task centric approaches are best left to the application developers. Standardisation of these dynamic tasks and process would only complicate the process as work methods, particularly during design development varies significantly among team members. Industry-wide consensus over specific ways of accomplishing tasks would be difficult to obtain.

Summary

This chapter reviewed the concepts of Information Modelling in the context of AEC/FM industry. A historical overview of the development of data exchange is given starting from IGES. Building information modelling concepts are introduced and different developments are critically analysed for suitability in the design, construction and facilities management continuum. STEP and IFC are introduced and components comprising the specifications are shown with the examples that form the basis for the adoption of IFC in this thesis. IFC has been chosen for implementation, as it is the most mature and internationally agreed upon standard for the AEC/FM domain.

The potentials for IFC in data exchange and sharing in a collaborative environment are established; possible alternatives are discussed. Future developments in product modelling, in particular ifcXML and how they link with the existing IFC specification are discussed in detail. A brief overview of interoperability-based framework and integrated construction environment are given in this chapter. Other approaches to interoperability are discussed with reference to the experiences gained from this research. This will form part of the future directions the industry might take. The following chapters will elaborate more on the implementation method of IFC while describing the system developed as part of this thesis.

References

AISC (2004). *Interoperability and the construction process: a white paper for building owners and project decision makers*. Chicago: American Institute of Steel Construction.

Alshawi, M., Putra, C.W. and Faraj, I. (1997). A structured concept for objects life cycle in integrated environments. *Microcomputers in Civil Engineering*, **12**, 339-351.

Amor, R. and Faraj, I. (2001). Misconceptions about integrated project databases. *Electronic Journal of Information Technology in Construction*, **6**, 57-68.

Amor, R.W., Clift, M., Scherer, R., Katranuschkov, P., Turk, Z. and Hannus, M. (1997). A framework for concurrent engineering - ToCEE. In: *European Conference on Product Data Technology, PDT Days 1997,*, Sophia Antipolis, France, 15-16 April, 1997, 15-22.

Amor, R.W. and Hosking, J.G. (1993). Multi-disciplinary views for integrated and concurrent design. *Management of information technology for construction*. Mathur, K.S., Betts, M.P.*et al.* Singapore: World Scientific Publishing**:** 239-254.

Amor, R.W. and Hosking, J.G. (1994). Mappings: the glue in an integrated system. *Product and process modelling in the building industry*. Scherer, R.J. Rotterdam: Balkema**:** 117-123.

Aouad, G. (1997). Integration: from a modelling dream into an implementation reality. In: *CIB W55: Internatioanl support for building economics*, Lake District, UK, 7-29.

Aouad, G., Betts, M., Brandon, P., Brown, F., Child, T., Cooper, G., Ford, S., Kirkham, J., Oxman, R., Sarshar, M. and Young, B. (1994). *ICON (Integration of CONstruction information): integrated databases for the design, procurement and management of construction*. Final report. Salford, Manchester, UK: Department of Surveying and Information Technology Institute.

Augenbroe, G. (1995). An overview of the COMBINE project. In: *ECPPM'94: The first European Conference on Product and Process Modelling in the Building Industry*, Dresden, Germany, 5-7 October 1994, 547-554.

Augenbroe, G., Wilde, P.d., Moon, H.J. and Malkawi, A. (2004). An interoperability workbench for design analysis integration. *Energy and Buildings*, **36**(8), 737-748.

Austin, S., Newton, A., Steele, J. and Waskett, P. (2002). Modelling and managing project complexity. *International Journal of Project Management*, **20**(3), 191-198.

Austin, S.A., Baldwin, A.N. and Newton, A.J. (1994). Manipulating the flow of design information to improve the programming of building design. *Construction Management and Economics*, **12**(5), 445-455.

Björk, B.C. (1994). The RATAS project - an example of co-operation between industry and research toward computer integrated construction. *Journal of Computing in Civil Engineering*, **4**, 401-419.

BRE (2005). *Introduction to IAI and IFCs*. [Online]. Available from: <http://cig.bre.co.uk/iai_uk/iai/page5.htm>

Brown, F.E., Cooper, G.S., Ford, S., Aouad, G., Brandon, P., Child, T., Kirkham, J.A., Oxman, R. and Young, B. (1995). An integrated approach to CAD: modelling concepts in building design and construction. *Design Studies*, **16**(3), 327-347.

Bruce, T. (1992). *Designing Quality Databases with IDEF1X Information Models*. New York: Dorset House.

Choo, S.Y. (2004). *Study on Computer-Aided Design Support of Traditional Architectural Theories*. PhD Thesis. München: Technische Universität München.

Clark, J., Ed. (1999). *XSL Transformations (XSLT) Version 1.0*: W3: World Wide Web Consortium.

Clark, J. and Makoto, M., Eds. (2001). *RELAX NG Specification*: OASIS.

Cohen, J. (2003). *The New Architect: Keeper of Knowledge and Rules*. Berkeley: JC Architects.

Cornick, T. (1990). *Quality Management for Building Design*. Guildford, Surrey: Butterworth-Heinemann.

Dawood, N., Akinsola, A. and Hobbs, B. (2002). Development of automated communication of system for managing site information using internet technology. *Automation in Construction*, **11**(5), 557-572.

de Vries, B. (1996). *Communication in the Building Industry*. PhD Thesis. Eindhoven, Netherlands: Eindhoven University of Technology.

Eastman, C.M. (1992). Modeling of buildings: evolution and concepts. *Automation in Construction*, **1**(2), 99-109.

Eastman, C.M. (1999). *Building product models: computer environments supporting design and construction*. Boca Raton, USA: CRC Press.

Egan, J. (1998). *Rethinking Construction*. London: DETR.

Fallside, D.C. and Walmsley, P., Eds. (2004). *XML Schema Part 0: Primer*: W3: World Wide Web Consortium.

Faraj, I., Alshawi, M., Aouad, G., Child, T. and Underwood, J. (2000). An industry foundation classes Web-based collaborative construction computer environment: WISPER. *Automation in Construction*, **10**(1), 79-99.

Fischer, M.A., Waugh, L.M. and Axworthy, A. (1998). IT support of single project, multi-project and industry-wide integration. *Computers in Industry*, **35**(1), 31-45.

Froese, T. (1994). Information Standards in the AEC Industry. *Canadian Civil Engineer*, **11**(6), 3-5.

Froese, T. and Paulson Jr., B.C. (1994). OPIS: an object model-based project information system. *Microcomputers in Civil Engineering*, **9**(1), 13-28.

Graphisoft (2001). *IFC Reference Guide*. Budapest: Graphisoft.

Greening, R. and Edwards, M. (1995). ATLAS implementation scenario. In: *ECPPM'94: The first European Conference on Product and Process Modelling in the Building Industry*, Dresden, Germany, 5-7 October 1994, 467-472.

Hannus, M., Karstilla, K. and Tarandi, V. (1995). Requirements on standardised building product data models. *Product and process modelling in the building industry*. Scherer, R.J. Rotterdam: Balkema**:** 43-50.

Haymaker, J., Kunz, J., Suter, B. and Fischer, M. (2004). Perspectors: composable, reusable reasoning modules to construct an engineering view from other engineering views. *Advanced Engineering Informatics*, **18**(1), 49-67.

Holmen, E., Pedersen, A.-C. and Torvatn, T. (2005). Building relationships for technological innovation. *Journal of Business Research*, **58**(9), 1240-1250.

Hunter, P. (1999). The Internet language that understands you. *Computing*,(29th April 1999), 22-23.

IAI (2005a). *IAI international: mission, vision and goal*. [Online]. Available from: <http://cig.bre.co.uk/iai_uk/iai/page3.htm>

IAI (2005b). *IFC Release 2x2 Final Online Documentation*. [Online]. Available from: <http://www.iai-international.org/Model/documentation/R2x2_Final/Online_Documents/index.htm>

ISO (1994a). *Industrial automation systems and integration - Product data representation and exchange - Part 1: Description methods: Overview and fundamental principles*. ISO 10303-1:1994. Geneva: ISO.

ISO (1994b). *Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual*. ISO 10303-11:1994. Geneva: ISO.

ISO (1994c). *Industrial automation systems and integration - Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure*. ISO 10303-21:1994. Geneva: ISO.

ISO (1998). *Industrial automation systems and integration - Product data representation and exchange - Part 22: Implementation methods: Standard data access interface*. ISO 10303-22:1998. Geneva, Switzerland: ISO.

ISO (2002). *Industrial automation systems and integration - Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure*. ISO 10303-21:2002. Geneva: ISO.

ISO (2003). *Industrial automation systems and integration - Product data representation and exchange - Part 28: Implementation methods: XML representations of EXPRESS schemas and data*. ISO 10303-28:2003. Geneva: ISO.

ISO (2005). *TC184/SC4 Organization*. [Online]. Available from: <http://www.tc184-sc4.org/About_FTC184-SC4/About_SC4_Organization/>

Jelliffe, R., Ed. (2005). *Resource Directory (RDDL) for Schematron 1.5*. Nankang Taipei: Academia Sinica Computing Centre.

Josephson, P.E. and Hammarlund, Y. (1999). The causes and costs of defects in construction: A study of seven building projects. *Automation in Construction*, **8**(6), 681-687.

Kalay, Y.E. (2001). Enhancing multi-disciplinary collaboration through semantically rich representation. *Automation in Construction*, **10**(6), 741-755.

Keane, M. and Kelliher, D. (2001). Green building design management. *International Journal of Architectural Management, Practice and Research*, **16**, 263-271.

Khelmani, L. (2004). *The IFC Building Model: A Look Under the Hood*. [Online]. Available from: <http://www.aecbytes.com/feature/IFCmodel.htm>

Koskela, L. (1992). *Application of the new production philosophy to construction*. CIFE Technical Report 72. Stanford: CIFE.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004). SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. *Automation in Construction*, **13**(5), 651-663.

Latham, M. (1994). *Constructing the Team: Final Report to the Government/Industry Review of Procurement and Contractual Arrangements in the United Kingdom Construction Industry*. London: HMSO.

Liebich, T., Forester, J., Karstila, K. and Wix, J., Eds. (2004). *IFC 2x Edition 2: Model Implementation Guide*. Washington: International Alliance for Interoperability.

Liebich, T. and Wix, J., Eds. (2000). *IFC Technical Guide*. Washington: International Alliance for Interoperability.

Loffredo, D.T. (1998). *Efficient Database Implementation of EXPRESS Information Models*. PhD Thesis. Troy, NY: Rensselaer Polytechnic Institute.

Luiten, G.T. (1994). *Computer aided design for construction in the building industry*. PhD Thesis. Eindhoven, Netherlands: Eindhoven University of Technology.

Mahdavi, A. (1996). SEMPER: a new computational environment for simulation-based building design assistance. In: *International Symposium of CIB W67 (Energy and Mass Flows in the Life Cycle of Buildings)*, Vienna, Austria, CIB, 467-472.

Mahdavi, A., Ilal, M.E., Mathew, P., Ries, R., Suter, G. and Brahme, R. (1999). The Architecture of S2. In: *BS '99: Sixth International IBPSA Conference*, Kyoto, Japan, September 13-15.

Mason, H. (2002). ISO 10303 - STEP: A key standard for the global market. *ISO bulletin*,(April 2002).

Moeller, G. (2000). STandard for the Exchange of Product model data. *Program Manager*, **XXIX**(6), 52-56.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

NBS (1998). *Initial Graphics Exchange Specification (IGES) Version 4.0*. NBSIR 88-3818. US National Bureau of Standards (Now National Institute of Standards and Technology).

Neil, J. (2001). *STEP on a page*. [Online]. Available from: <http://www.mel.nist.gov/sc5/soap/>

Nijssen, G.M. and Halpin, T. (1989). *Conceptual Schema and Relational Database Design*. Sydney: Prentice Hall.

Nisbet, N. and Liebich, T., Eds. (2005). *ifcXML Implementation Guide*. Washington: International Alliance for Interoperability.

NIST (2004). *Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry*. NIST GCR 04-867. Gaithersburg, MD: US National Institute of Standards and Technology.

Orange, G., Burke, A. and Cushman, M. (1999). An Approach to Support Reflection and Organisation Learning with the UK Construction Industry. In: *BITWorld'99: Business Information Technology Management: The Global Imperative*, Cape Town, South Africa.

Owolabi, A., Anumba, C.J. and El-Hamalawi, A. (2003). Architecture for implementing IFC-based online construction product libraries. *Electronic Journal of Information Technology in Construction*, **8**, 201-218.

Rezgui, Y., Brown, A., Cooper, G., Yip, J., Brandon, P. and Kirkham, J. (1996). An information management model for concurrent construction engineering. *Automation in Construction*, **5**(4), 343-355.

Rosenman, M.A. and Gero, J.S. (1996). Modelling multiple views of design objects in a collaborative environment. *Computer-Aided Design*, **28**(3), 193-205.

Schenck, D. and Wilson, P. (1994). *Information Modelling - The EXPRESS Way*. New York: Oxford University Press.

Scherer, R. (1995). EU-project COMBI - Objectives and overview. In: *ECPPM'94: The first European Conference on Product and Process Modelling in the Building Industry*, Dresden, Germany, 5-7 October 1994, 503-510.

Schrage, M. (1990). *Shared Minds: the New Technologies of Collaboration*. New York: Random House.

Sun, M. and Lockley, S.R. (1997). Data exchange system for an integrated building design system. *Automation in Construction*, **6**(2), 147-155.

Syal, M.G., Parfitt, M.K. and Willenbrock, J.H. (1991). *Computer integrated design/drafting, cost estimating and construction scheduling*. 11. Department of Civil Engineering, The Pennsylvania State University.

Tischritzis, D. and Klug, A. (1978). The ANSI/X3/SPARC DBMS Framework Report of the Study Group on Database Management. *Information Systems*, **3**, 173-191.

Tolman, F.P. (1999). Product modelling standards for the building and construction industry: past, present and future. *Automation in Construction*, **6**(2), 227-235.

Underwood, J. and Watson, A. (2003). An XML Metadata Approach to Seamless Project Information Exchange between Heterogeneous Platforms. *Engineering, Construction and Architectural Management*, **10**(2), 128-145.

van Leeuwen, J.P. (1999). *Modelling Architectural Design Information by Features*. PhD Thesis. Eindhoven: Eindhoven University of Technology.

van Nederveen, G.A. and Tolman, F.P. (1992). Modelling multiple views on buildings. *Automation in Construction*, **1**(3), 215-224.

Verhoef, M., Liebich, T. and Amor, R. (1995). A multi-paradigm mapping method survey. *Modelling of buildings through their life-cycle*. Fischer, M.A., Law, K.H.*et al.* Stanford, CA: Stanford University**:** 233-247.

Zhu, Y. and Augenbroe, G. (2006). A conceptual model for supporting the integration of inter-organizational information processes of AEC projects. *Automation in Construction*, **15**(2), 200-211.

Chapter 4
Optimisation methods and applications

This chapter is devoted to the study of optimisation methods and their applications in design, particularly in the design of buildings. Classifications of optimisation methods are discussed including reviews of the principles of the algorithms implemented in this thesis. Single and multiple criteria approaches to optimisation are introduced. Pareto optimality for informed decision making in multi-criteria optimisation problems is described. A review of the applications of optimisation methods in architectural design such as automated space layout and environmental design of buildings is provided.

Mathematical optimisation methods

Mathematical optimisation methods can be classified into two distinct types: *unconstrained* and *constrained* depending on the presence of functional constraints. According to the number of objective functions they can also be classified into two: *single objective* and *multi-objective* or *multi-criteria*. Traditional optimisation methods can again be classified into two groups: *direct* and *gradient-based* methods (Deb 1995). Only objective function ($f(x)$) and constraint values ($g_j(x), h_k(x)$) are used to guide the search strategy in direct search methods. Gradient-based methods use the first and/or second-order derivatives of the objective function and/or constraints to guide the search process. On the other hand, GAs (*Genetic Algorithms*) are search and optimisation procedures based on the principles of natural genetics and natural selection; these are also known as zero order methods. Optimisation methods discussed here can be seen as extensions to the general concepts of optimisation (Equations 1.1 to 1.10) introduced in Chapter 1.

Unconstrained optimisation

Unconstrained optimisation problems have an objective function but no constraints. The objective function must be nonlinear because the minimum of an unconstrained linear objective function is obviously $-\infty$ (Bhatti 2000). There are several numerical methods available for solving unconstrained optimisation problems such as *Newton's method*, *steepest descent*, and *conjugate gradient*. Since architectural design problems are normally constrained, no further discussion is carried out on unconstrained optimisation. Further information on unconstrained methods can be found in (Fox 1971; Haftka and Gürdal 1992).

Constrained optimisation

There are many techniques available for the solution of a constrained nonlinear optimisation problem. Rao (Rao 1996) classified these methods into two broad categories: *direct methods* and *indirect methods*, as shown in Table 4.1. In the direct methods, the constraints are handled in an explicit manner, whereas in most of the indirect methods, the constrained problem is solved as a sequence of unconstrained minimisation problems. Direct methods such as *Sequential Linear Programming* and *Sequential Quadratic Programming*, implemented in this thesis are discussed. Further information on constrained optimisation can be obtained from (Polak 1997; Bonnans *et al.* 2003).

Table 4.1: Constrained optimisation techniques, after (Rao 1996).

Direct Methods	***Indirect Methods***
Random search	Transformation of variables technique
Heuristic search	Sequential unconstrained minimisation
Complex method	Interior penalty function method
Objective and constraint approximation	Exterior penalty function method
Sequential linear programming	Augmented Lagrange multiplier method
Sequential quadratic programming	
Methods of feasible directions	
Zoutendijk's method	
Rosen's gradient projection method	
Generalised reduced gradient method	

Sequential Linear Programming

The Sequential Linear Programming (SLP) method, also known as the *cutting plane method* involves constructing an approximate linear optimisation problem about the current design point and solving this to generate a new updated design. Through a series of linear programming problems, the solution to the original nonlinear programming problem is found. Each Linear Programming (LP) problem is generated by approximating the nonlinear objective and constraint functions which is based on the Taylor series expansions about the current design vector, $\mathbf{X}_i$. The resulting LP problem is solved using the simplex method to find the new design vector $\mathbf{X}_{i+1}$. If the new vector $\mathbf{X}_{i+1}$ does not fulfil a specified tolerance of convergence criteria, the problem is relinearised about the point $\mathbf{X}_{i+1}$ and the procedure is continued until the optimum solution $\mathbf{X}^*$ is found. The SLP algorithm, as implemented in this thesis through incorporating VisualDOC API proceeds in the following steps (VR&D 2002):

For the current values of the design variables, the constraints are sorted; the most critical constraints for use during this cycle are retained. The reason that not all constraints are retained because there may be a large number of constraints, and most can be far from critical,

- A first order Taylor series expansion of the objective and retained constraints with respect to design variables is created,
- Move limits on the design variables are defined. Typically during one cycle, the design variables are allowed to change by 20-40%, which is adjusted during later cycles,
- Linear approximate optimisation problem is solved, and
- Convergence is checked. If not satisfied the process is repeated from step 1.

Sequential Quadratic Programming

The Sequential Quadratic Programming (SQP), also known as *Recursive Quadratic Programming*, falls under the heading of *Lagrange* or *Newton-Lagrange* methods. It is considered to be an excellent and robust method for medium size nonlinear constrained problems by researchers (Stoer 1985). With the appropriate quadratic subproblem, the method can be viewed as an extension of *Newton* or *quasi-Newton* algorithms to constrained optimisation (Kruk and Wolkowicz 1998). The basic concept is very similar to Sequential Linear Programming. First, a Taylor series approximation to the objective and constraint functions is created. Instead of minimising the linearised objective, a quadratic approximate objective function is created. Then the linearised constraints are used with this approximate objective function to create a direction finding problem of the form (VR&D 2002):

Minimise $$Q(\mathbf{S}) = F^0 + \nabla F^T \mathbf{S} + \tfrac{1}{2}\mathbf{S}^T \mathbf{B}\mathbf{S} \qquad (4.1)$$

Subject to: $$\left(\nabla g_j\right)^T \mathbf{S}^q + g_j^0 \le 0, \qquad j = 1, M \qquad (4.2)$$

The subproblem is solved using the Modified Method of Feasible Directions. The matrix $\mathbf{B}$ is a positive definite matrix, which is initially the identity matrix. On subsequent iterations, $\mathbf{B}$ is updated to approach the Hessian of the Lagrangian function.

Design of Experiments

Design of Experiments (DOE) is used when the designer is not certain about the underlying relationship between the responses and design variables but wants to know how the responses are influenced by the design variables. Such problems of experimental design are encountered in the iterative design processes such as architectural design. The response variables of interest are $y_1, y_2, \ldots, y_m$ (m is the total number of response variables or responses) and there is a set of predictor variables $x_1, x_2, \ldots, x_n$ (n is the total number of predictor variables or design variables or factors). The underlying relationship can be approximated with an empirical model:

$y_i = f(x_1, x_2, ..., x_n)$. The function $f(\mathbf{x})$ is usually a first- or second-order polynomial (VR&D 2002). This empirical model is called a *Response Surface* (RS) model or *curve fit*. The RS model is created from the value of responses for some combinations of design variables. Each combination of design variables is viewed as a point in the n-dimensional design space where n is the total number of design variables. The particular arrangement of points in the design space is known as *Design Of Experiments*.

The quality of RS model depends on the selection of design points. Using a random selection of design points in the design space to create a RS model may result in significant computation and other expenses due to fact that a large number of points are required to estimate all terms in the RS model. If the selection of points is done by changing one factor at a time, some important information about the interaction of responses and factors may be missed. It is generally suggested that if the designer is interested only in the effect that factors have on responses, then a first-order polynomial model can be used as a response surface model. The same can be said for the identification of the most significant factors where the designer is willing to ignore the interactions between the factors. To construct an accurate approximation of a particular response, the use of second-order polynomial model is generally suggested.

Response Surface Approximate optimisation

Where there are relatively few design variables and the cost of analysing a single design is high, which is sometimes the case in building simulation-based optimisation; the *Response Surface Approximate* (RSA) method may prove to be the most efficient approach to optimisation. The idea of response surface approximate optimisation is to create explicit approximation functions to the objective and constraints, and then use these when performing optimisation. The approximation functions are typically in the form of low order polynomials (linear or quadratic) fit by least squares regression analysis. Once a RSA model is constructed, it can be used as cheap function evaluators, replacing the underlying computationally expensive analysis tools. Further information on RSA can be found in (Khuri 1996; Kirsch 2002). To further improve the efficiency, application of the neural networks-based RSA instead of conventional regression-based ones is suggested (Gosavi 2003).

Analysis in engineering and architectural design is often encumbered with noisy behaviour of responses. The presence of noise in numerical simulations may result in incomplete convergence of iterative processes. In direct gradient-based optimisation methods, noisy responses may cause the search algorithm to get caught in a spurious local minimum, which has been reported to be the case in detailed-based HVAC optimisation in buildings by Wetter (Wetter 2005). In contrast, RSA optimisation is much more robust towards noise and thus enjoys higher success rates. Figure *4.1* gives an example of the noise filtration capabilities of response surface approximate algorithms for a problem with a noisy behaviour by responses generated from a finite element stress analysis.

Figure 4.2 gives an overview of an incrementally refined response surface approximate algorithm where a sequence of quadratic approximations is performed. It starts out covering more or less the entire design space and then gradually zooms in on the region that holds the optimum point. This approach requires a considerable number of analyses to construct the set of quadratic approximations.

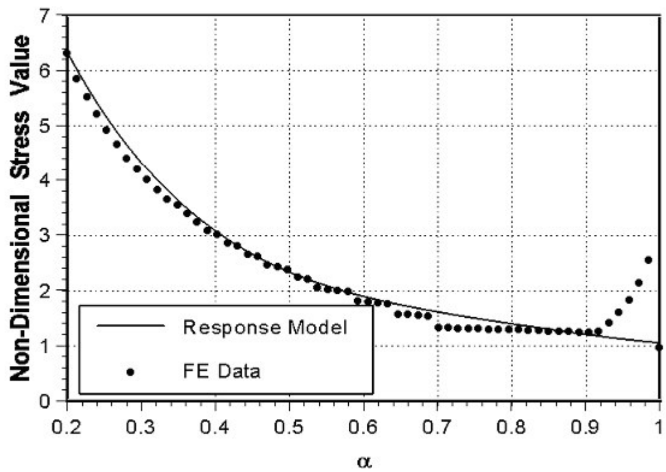


Figure 4.1: An example of a response surface approximate algorithm to a noisy response from a FEA (Finite Element Analysis), after (VR&D 2002).

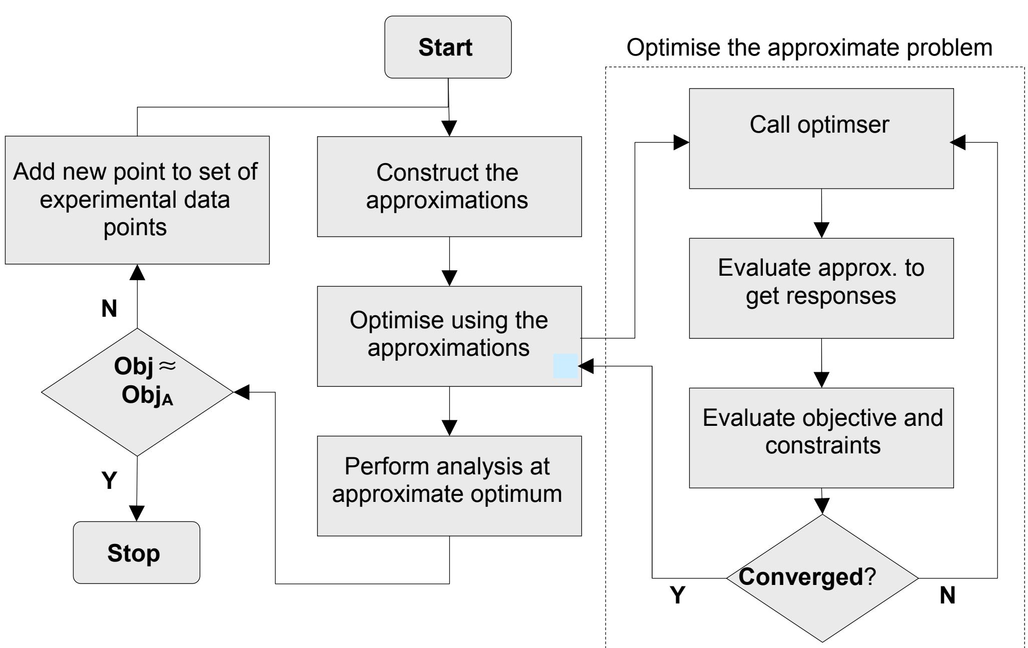


Figure 4.2: Incrementally refined response surface approximate algorithm.

Algorithms based on natural processes

Gradient based methods are very efficient if applied to problems where the design space is convex, with and when there are no severe discontinuities in the objective functions or its constraints (de Sousa *et al.* 2004). Many design problems have complex design spaces - usually non-convex and disjoint. Objective function(s) and its constraints may contain a mix of continuous, discrete and integer design variables; introduce(s) non-linearity in the objective function(s) and characterised by the existence of multiple local minima (Eldred 1998). This phenomenon of multiple local minima in certain cases of environmental design of buildings has been observed in this thesis, discussed in details in Chapter 6; also been reported in (Wetter and Polak 2005). In such cases, a gradient-based method would converge to local optimum unless the designer plays with the convergence and stopping criteria of the algorithm. Implementing global search strategies using deterministic or heuristic approaches may help to avoid the algorithm being trapped in a local minimum. Algorithms based on the observations of natural processes, on the other hand, have been observed to produce simple and efficient solutions; although they are computationally expensive. Algorithms based on the evolution of species (Davis *et al.* 1999), on the immune system (de Castro and Timmis 2002), on the annealing of metals (Kirkpatrick *et al.* 1983) and on the social behaviour of ants (Bonabeau *et al.* 2002) have been used in optimisation problems. The most commonly used algorithms based on natural processes are Simulated Annealing (Kirkpatrick *et al.* 1983), Genetic Algorithm (Goldberg 1989) or variations of them (de Sousa *et al.* 2004).

Genetic Algorithm (GA) is a global search technique capable of dealing with discrete variables with ease and often used in simulation-based optimisation problems (Wang *et al.* 2005a). GAs are essentially search and optimization procedures that are motivated by the principles of natural genetics and natural selection. A variable is usually coded into a string of fixed length of bits consisting of '1's and '0's. The binary codes of all variable values are concatenated to form a binary string. Coding the design variables in a binary string is primarily used to have pseudo-chromosomal representation of a design solution. The phenotypic representations are thus encoded into genotypic representations or chromosomes. The first generation of the population is usually randomly generated while the remaining generations are produced through genetic operations: selection, crossover and mutation.

Application of genetic algorithms to optimisation of complex problems involving many variables, which is usually the case in architectural design, can lead to a substantial computational effort. This is resulted from the repeated evaluation of the objective function(s) and the population-based nature of the search. This is often the case where the objective function evaluation is costly. Usually a large number of generations are required to converge to optimum. As mutation is the strategy to induce optimiser to search the solution space, GAs can face convergence problems. GA efficiency is shown to be enhanced by Hybridising GA with other established methods (Javadi *et al.* 2005). Further details on GAs can be found in (Goldberg 1989; Gen and Cheng 1996).

Pareto optimality

In multi-objective optimisation, the set of solutions for consideration can be reduced to a subset of the attainable set by reducing at least one criterion without increasing any of the others. This subset is known as *Pareto set* that consists of *Pareto optimal points*. A point c_0 in the attainable set A is Pareto optimal if and only if there is not another $c \in A$ such that $c_i \leq c_{0i}$ for all i and $c_i < c_{0i}$ for at least one i. So in multi-objective optimisation a point in the design space is a Pareto (optimal) point if no feasible point exists that would reduce one criterion without increasing the value of one or more of the other criteria (Papalambros and Wilde 2000). Attainable and Pareto sets for a problem with two criteria optimisation can be represented as in Figure 4.3.

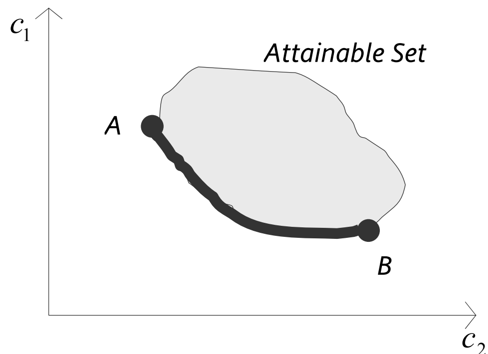


Figure 4.3: Attainable set and Pareto set (line segment AB) for a bi-criterion problem.

Observations on optimisation algorithms

Many types of optimisation methods have been developed to address optimisation problem in science and engineering (Gen and Cheng 1996; Patnaik *et al.* 1996; Vanderplaats 1998; Papalambros and Wilde 2000; Glover and Kochenberg 2003). The efficiency of a given optimisation algorithm is dependent on the kind of problem that is being tackled. Each one has its advantages and disadvantages - a single technique does not exist that can be applied to all problems. Some methods are better suited to a given class of problems. Architectural design problems are characterised by the existence of a large number of design variables and expensive design evaluations; i.e. simulations that are computationally expensive. Applications of GA or other evolutionary based algorithm in architectural design require considerable amount of computing resources to obtain results as opposed to gradient based methods. Sequential Quadratic Programming have been the choice of algorithm in many engineering applications requiring a robust solution (Thanedar *et al.* 1986; Rao 1996).

Application of optimisation in the design of buildings

The application of optimisation methods in the design of buildings ranges from space layout problems to the optimisation of building form and envelope. Liggett, in 1985 (Liggett 1985) and in 2000 (Liggett 2000) provided two comprehensive reviews of automated facility layout starting from the early 1960's. Facility or space layout in architecture is concerned with the allocation of activities to space such that a set of criteria are met and/or some objectives, usually some measure of communication costs are optimised. Liggett (Liggett 2000) reported that there are three major paths space layout solution techniques have followed. The first involves optimisation of a single criterion function - the minimisation of costs associated with communication or flow of materials between activities.

The second path is based on a graph theoretic approach, primarily concerned with generating a layout that meets adjacency requirements between activities. Early works according to this approach can be found in (Grason 1971). Among other approaches notable is the *Systematic Layout Planning Methodology* (Muther 1973) by Richard Muther that results in the generation of a space relationship diagram - a *design skeleton,* from which a layout can be generated. The third path concerns itself with finding an arrangement that satisfies a diverse set of constraints or relations. Early examples following this path are Eastman's *General Space Planner* (Eastman 1973) and Pfefferkorn's *Design Problem Solver* (Pfefferkorn 1975). More recent advances in the third approach can be found in the layout module of SEED (Flemming *et al.* 1994), a software system to support space programming activities during early stages of design. Notable in the SEED is the idea and implementation of user interface (Akin *et al.* 1995) based design exploration during early stages in addition to the methods of optimisation. Figure 4.4 shows SEED-Layout design space.

In terms of solution strategies, QAP (Quadratic Assignment Problems) as well as evolutionary approaches such as genetic algorithms have been applied to solve layout problems. QAPs are shown to produce high quality solutions to realistically sized problems at acceptable cost (Liggett 2000). GAs using mutation and crossover techniques to improve initial starting designs can be found in (Kar Yan Tam 1992; Gero and Kazakov 1998; Jo and Gero 1998). Constrained based approaches can be found in (Medjdoub and Yannou 2000).

Applications of optimisation techniques in environmental design of buildings can be classified based on the selection of design problems and on the number of objective functions. Based on the selection of design problems, either the architectural design of the building or the HVAC system could be optimised. Depending on the types of objective functions, environmental design optimisation can be classified as: *single-objective* and *multi-objective*. Evaluation criteria, i.e. objective functions could be from end-user energy consumption and/or daylighting to life cycle analysis.

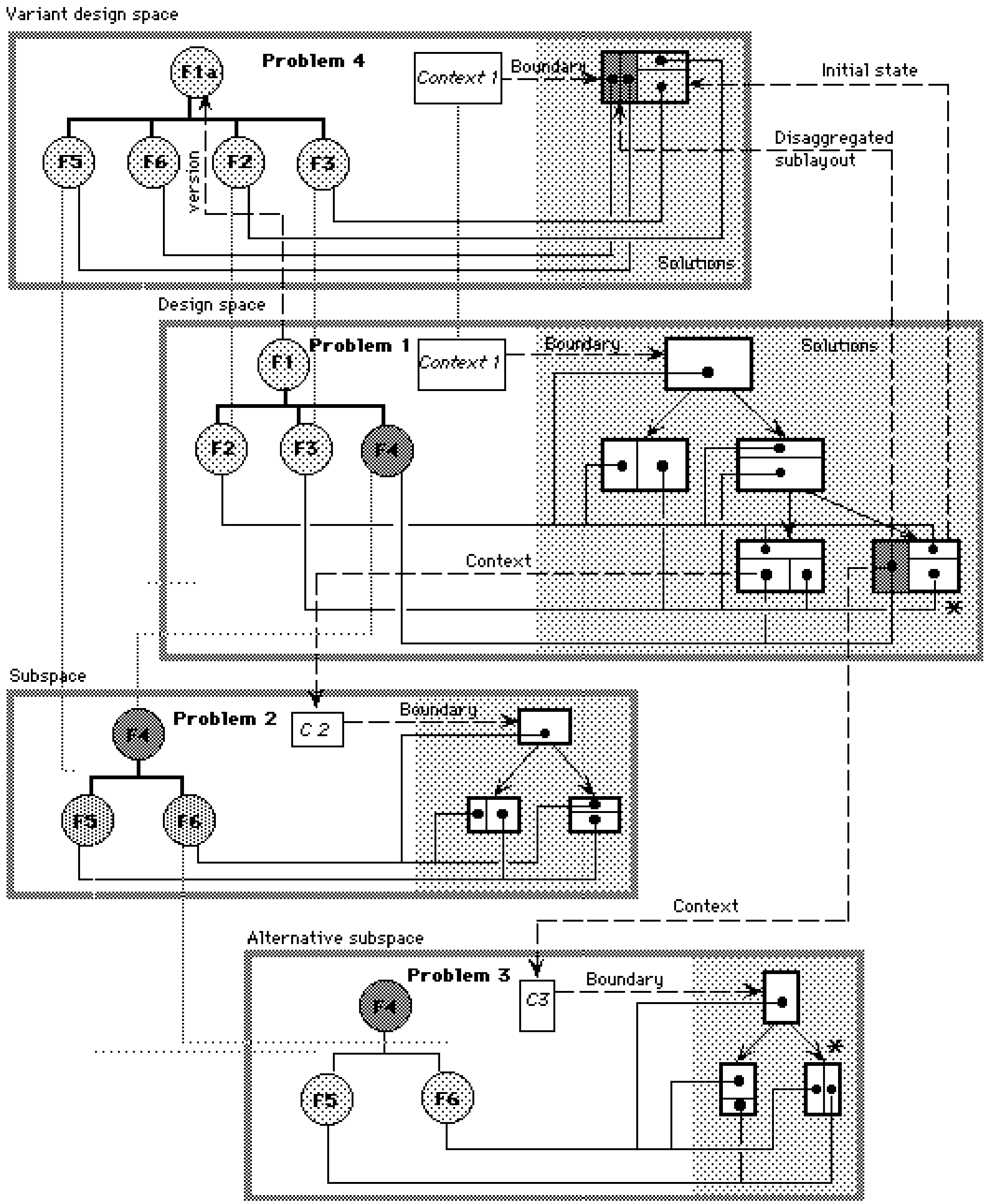


Figure 4.4: The full SEED-Layout design space, after (Flemming 1999).

End user energy consumption has been used as optimisation criteria by many researchers. Heating and cooling energy have been used by Al-Homoud in optimisation of office buildings (Al-Homoud 1997b) and residential buildings (Al-Homoud 1997a). Coley and Schukat have also applied similar optimisation criteria (Coley and Schukat 2002). Simply minimising building energy consumption may lead to compact building forms with high insulations. To provide realistic design solutions lighting energy consumption involving replaced artificial lighting with daylighting has been considered by Caldas (Caldas 2001), Wetter (Wetter and Wright 2003) and Mourshed et al. (Mourshed *et al.* 2003b) in addition to building energy consumptions. All three are based on an integrated environmental performance assessment using whole building energy simulation tools such as DOE or EnergyPlus (Crawley *et al.* 2001).

Peippo et al. (Peippo *et al.* 1999) adopted similar concepts of integrating multiple simulation domains in optimisation but implemented a set of rather idealised models to address the coupled problem of interdependent building energy flows (Figure 4.5). The objective was to reduce computational time for simulation. LCA (Life Cycle Analysis) has been incorporated as an optimisation criteria for design of optimised building form by Nielsen (Nielsen 2002) and for cost optimisation of hybrid HVAC system with composite radiant wall panels by Kilkis (Kilkis 2006). The use of LCA in the optimisation of systems is more commonplace than the design of building forms because of the lack of industry-standard life cycle analysis tools for early design stages, in which details about buildings are not fully known.

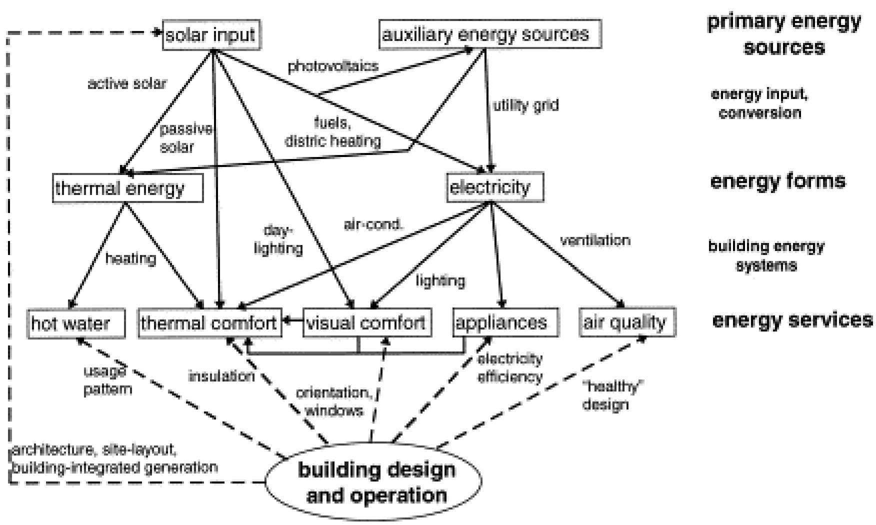


Figure 4.5: The energy flows, conversion processes and end-use in a building as implemented by Peippo et al. (1999).

Since building design is essentially an iterative multi-criteria decision making process (Mourshed *et al.* 2003b), single-objective optimisations are not applicable in practice. Examples of multi-objective optimisations can be found in (Radford and Gero 1988; Hauglustaine and Azar 2001; Wright *et al.* 2002). Radford and Gero (Radford and Gero 1988) used four performance criteria: thermal loads, daylight availability, construction cost and usable area in employing dynamic programming in the multi-criteria design optimisations. Wright et al. (Wright *et al.* 2002) applied a multi-objective genetic algorithm to optimise system and controls. Hauglustaine and Azar (Hauglustaine and Azar

2001) optimised building envelope using genetic algorithms and 10 criteria related to code compliance, energy consumption and cost.

Integrated building simulation considering the domains of thermal and daylighting has been the popular choice as a response generator. Simulation tools, as they are based on iterative solvers, have been reported as producing discontinuous results (Wetter 2005) when detailed-based simulation model is used. However, results from this research suggest that discontinuity in the objective function, if occurs, is related to how the particular problem is modelled for simulation. This is also reported in (Griffith *et al.* 2003). Integrated whole building simulation tools when used in performance evaluation generates responses that can be used for exploration in architectural design, provided that the simulation modelling focuses on heating and cooling loads than system sizing. The development of simplified simulation models specifically for use in optimisation, suggested in (Wetter 2005), would only produce non-reliable results mainly because the impact of environmental parameters on buildings does not occur in isolation. Modelling for simulation thus require special attention. Ideal HVAC systems, also referred to as purchased air based simulation in EnergyPlus has been seen as producing good results for use in optimisation during early stages.

Summary

This chapter summarised optimisation methods and its application in design, in particular in the design of buildings. Comparative efficiencies as to the types of optimisation methods are discussed. It is shown from a review of optimisation literature that not all methods suit every type of design problems. Design activity as multi-criteria optimisation is introduced. Automated space layout and optimisation of environmental design of buildings are discussed as part of the applications of optimisation in architectural design. Gradient-based and non gradient-based algorithms used for environmental design of buildings are described here. Trend towards the use of whole building integrated simulation tools as opposed single domain (i.e. thermal or daylighting) is observed for efficient evaluation of environmental design of buildings. Discussions in this chapter together with discussions from Chapter 2 and 3 form the basis for implementation of the framework in chapter 6.

References

Akin, O., Sen, R., Donia, M. and Zhang, Y. (1995). SEED-Pro: Computer assisted architectural programming in SEED. *Journal of Architectural Engineering*, **1**(4), 153-161.

Al-Homoud, M.S. (1997a). Optimum thermal design of air-conditioned residential buildings. *Building and Environment*, **32**(3), 203-210.

Al-Homoud, M.S. (1997b). Optimum thermal design of office buildings. *International Journal of Energy Research*, **21**(10), 941-957.

Bhatti, M.A. (2000). *Practical Optimization Methods*. New York: Springer.

Bonabeau, E., Dorigou, M. and Theraulaz, G. (2002). Inspiration for optimization from social insect behaviour. *Nature*, **406**, 39-42.

Bonnans, J.F., Gilbert, J.C., Lemaréchal, C. and Sagastizábal, C.A. (2003). *Numerical Optimization: Theoretical and Practical Aspects*. New York: Springer.

Caldas, L.G. (2001). *An Evolution-Based Generative Design System: Using Adaptation to Shape Architectural Form*. PhD Thesis. Cambridge, MA: Massachusetts Institute of Technology.

Coley, D.A. and Schukat, S. (2002). Low-energy design: combining computer-based optimisation and human judgement. *Building and Environment*, **37**(12), 1241-1247.

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl, W.F., Huang, Y.J., Pedersen, C.O., Strand, R.K., Liesen, R.J., Fisher, D.E., Witte, M.J. and Glazer, J. (2001). EnergyPlus: creating a new-generation building energy simulation program. *Energy and Buildings*, **33**(4), 319-331.

Davis, L.D., De Jong, K., Vose, M.D. and Whitley, L.D., Eds. (1999). *Evolutionary algorithms*. The IMA Volumes in Mathematics and its Applications. New York: Springer.

de Castro, L.N. and Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach*. London: Springer.

de Sousa, F.L., Ramos, F.M., Galski, R.L. and Muraoka, I. (2004). Generalized External Optimization: A New Meta-Heuristic Inspired by a Model of Natural Evolution. *Recent Developments in Biologically Inspired Computing*. Zuben, F.J.V. and de Castro, L.N. Hershey, PA: Idea Group Inc**:** 41-60.

Deb, K. (1995). *Optimization for engineering design: Algorithms and examples*. Delhi: Prentice-Hall.

Eastman, C. (1973). Automated space planning. *Artificial Intelligence*, **4**, 41-64.

Eldred, M.S. (1998). *Optimization strategies for complex engineering applications*. SAND98-0340, UC-705. Albuquerque, NM: Sandia National Laboratories.

Flemming, U. (1999). *SEED-Layout Tutorial*. Pittsburgh, PA: Carnegie Mellon University.

Flemming, U., Coyne, R., Fenves, S., Garrett, J. and Woodbury, R. (1994). SEED - software environment to support the early phases in building design. In: *IKM94*, Weimar, Germany, 5-10.

Fox, R.L. (1971). *Optimisation methods for Engineering Design*. Reading, MA: Addison-Wesley.

Gen, M. and Cheng, R. (1996). *Genetic Algorithms and Engineering Design*. New York: Wiley-IEEE.

Gero, J.S. and Kazakov, V.A. (1998). Evolving design genes in space layout planning problems. *Artificial Intelligence in Engineering*, **12**(3), 163-176.

Glover, F.W. and Kochenberg, G.A., Eds. (2003). *Handbook of metaheuristics*. International series in operations research & management science. Boston: Kluwer Academic Publishers.

Goldberg, D.E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Gosavi, A. (2003). *Simulation-Based Optimization: Parametric Optimization Techniques and Reinforcement Learning*. New York: Springer.

Grason, J. (1971). An approach to computerized space planning using graph theory. In: *Design Automation Workshop*, Atlantis City, NJ, June 28-30, IEEE, 170-179.

Griffith, B., Pless, S., Talbert, B., Deru, M. and Torcellini, P. (2003). *Energy Design Analysis and Evaluation of a Proposed Air Rescue and Fire Fighting Administration Building for Teterboro Airport*. NREL/TP-550-33294. Colorado: National Renewable Energy Laboratory.

Haftka, R.T. and Gürdal, Z. (1992). *Elements of Structural Optimization*. New York: Springer.

Hauglustaine, J.M. and Azar, S. (2001). Interactive tool aiding to optimise the building envelope during the sketch design. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 387-394.

Javadi, A.A., Farmani, R. and Tan, T.P. (2005). A hybrid intelligent genetic algorithm. *Advanced Engineering Informatics*, **19**(4), 255-262.

Jo, J.H. and Gero, J.S. (1998). Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering*, **12**(3), 149-162.

Kar Yan Tam (1992). Genetic algorithms, function optimization, and facility layout design. *European Journal of Operational Research*, **63**(2), 322-346.

Khuri, A.I. (1996). *Response Surfaces*. New York: Marcel Dekker.

Kilkis, B.I. (2006). Cost optimization of a hybrid HVAC system with composite radiant wall panels. *Applied Thermal Engineering*, **26**(1), 10-17.

Kirkpatrick, S., Gellat, C.D. and Vecchi, M.P. (1983). Optimization by simulated annealing. *Science*, **220**(4598), 671-680.

Kirsch, U. (2002). *Design-Oriented Analysis of Structures: A Unified Approach*. Dordrecht: Springer.

Kruk, S. and Wolkowicz, H. (1998). SQ^2P, Sequential Quadratic Constrained Quadratic Programming. *Advances in Nonlinear Programming*. Yuan, Y.-X. Dordrecht, the Netherlands: Kluwer Academic Publishers**:** 177-204.

Liggett, R. (1985). Optimal spatial arrangement as a quadratic assignment problem. *Design Optimization*. Gero, J. New York: Academic Press**:** 1-40.

Liggett, R.S. (2000). Automated facilities layout: past, present and future. *Automation in Construction*, **9**(2), 197-215.

Medjdoub, B. and Yannou, B. (2000). Separating topology and geometry in space planning. *Computer-Aided Design*, **32**(1), 39-61.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

Muther, R. (1973). *Systematic Layout Planning*. Boston: Cahners Books.

Nielsen, T.R. (2002). *Optimization of buildings with respect to energy and indoor environment*. PhD Thesis. Lyngby: Technical University of Denmark.

Papalambros, P.Y. and Wilde, D.J. (2000). *Principles of optimal design: modeling and computation*. New York: Cambridge University Press.

Patnaik, S.M., Coroneos, R.M., Guptill, J.D. and Hopkins, D.A. (1996). Comparative evaluation of different optimization algorithms for structural design applications. *International Journal for Numerical Methods in Engineering*, **39**(10), 1761-1774.

Peippo, K., Lund, P.D. and Vartiainen, E. (1999). Multivariate optimization of design trade-offs for solar low energy buildings. *Energy and Buildings*, **29**(2), 189-205.

Pfefferkorn, C. (1975). The design problem solver: a system for desigining equipment or furniture layouts. *Spatial Synthesis in Computer-Aided Building Design*. Eastman, C. New York: Wiley**:** 98-146.

Polak, E. (1997). *Optimization: Algorithms and Consistent Approximations*. New York: Springer.

Radford, A.D. and Gero, J.S. (1988). *Design by Optimization in Architecture, Building and Construction*. New York: Van Nostrand Reinhold Co.

Rao, S.S. (1996). *Engineering Optimization: Theory and Practice*. New York: Wiley-IEEE.

Stoer, J. (1985). Principles of sequential quadratic programming methods for solving nonlinear programs. *Computational Mathematical Programming*. Schittkowski, K. Michigan: The University of Michigan**:** 165-207.

Thanedar, P.B., Arora, J.S., Tseng, C.H., Lim, O.K. and Park, G.J. (1986). Performance of some SQP algorithms on structural design problems. *International Journal of Numerical Methods in Engineering*, **23**, 2187-2203.

Vanderplaats, G.N. (1998). *Numerical optimization techniques for engineering design*. Colorado Springs, CO: Vanderplaats Research & Development.

VR&D (2002). *VisualDOC: Design Optimization Software Version 3.0: Theoretical Manual*. Colorado Springs, CO: Vanderplaats Research & Development.

Wang, W., Rivard, H. and Zmeureanu, R. (2005). An object-oriented framework for simulation-based green building design optimization with genetic algorithms. *Advanced Engineering Informatics*, **19**(1), 5-23.

Wetter, M. (2005). BuildOpt--a new building energy simulation program that is built on smooth models. *Building and Environment*, **40**(8), 1085-1092.

Wetter, M. and Polak, E. (2005). Building design optimization using a convergent pattern search algorithm with adaptive precision simulations. *Energy and Buildings*, **37**(6), 603-612.

Wetter, M. and Wright, J. (2003). Comparison of a generalized pattern search and a genetic algorithm optimization method. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 1401-1408.

Wright, J.A., Loosemore, H.A. and Farmani, R. (2002). Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and Buildings*, **34**(9), 959-972.

Chapter 5
Development of the framework

This chapter describes the methodology of implementation in this thesis. Considering the aspects of building simulation, interoperability and mathematical optimisation, introduced in chapters 2, 3, and 4 respectively, the development of the integrated framework is described here. Optimisation of environmental design of buildings during conceptual stage is introduced as the design activity for implementation. The rationale behind the selection of domain and simulation engines is described.

Development of the framework

The integrated framework developed as part of this thesis builds on the existing knowledge base on information systems development, building simulation, IFC/EXPRES, AEC applications and distributed computing. The framework is implemented in environmental design of buildings integrating architectural and HVAC domain, although it can be applied in other AEC domains and processes. An object oriented three-tier information system has been chosen as the architecture of the framework. To enable collaboration between geographically dispersed project teams, client layer applications may take the form of desktop or web based distributed application. For implementation of the framework, two standalone applications using Java[1] and C++ have been developed as well as a dynamic web application using PHP[2] and MySQL[3]. The web application has been designed independent of software platforms.

System architecture

The implemented framework is designed as a distributed three tier information system (Figure 5.1) to provide flexibility, maintainability, reusability and scalability. The difference between three-tier and two-tier system is the additional layer containing integration logic which enhances interoperability. The resulting performance loss is generally compensated for by the flexibility achieved through this additional tier and the support it provides to the application logic (Alonso *et al.* 2003). More information on three tier system architecture can be found in (Langer 2000; Weijia

[1] Java is an object oriented high level programming language intended for machine-independent software development. http://java.sun.com

[2] PHP is a general purpose scripting language. http://www.php.net

[3] MySQL is an open source Database Management System. http://www.mysql.org

and Zhou 2004). The three tiers, otherwise known as layers are: data management, process management and user interface. The core or the bottom layer is the data representation where all project data are stored. Several other databases, e.g. product database, standards database, etc. may reside in this layer. IFC (Adachi *et al.* 2004), as reviewed in Chapter 3, is the chosen standard for database implementation. The aim is to integrate multiple domains with the AEC processes and enhance collaboration and interoperability among applications and participants. Examples of three-tier system architecture based on IFCs can be found in (Faraj *et al.* 1999; Keane and Kelliher 2001; Owolabi *et al.* 2003; Yang 2003; Lam *et al.* 2004).

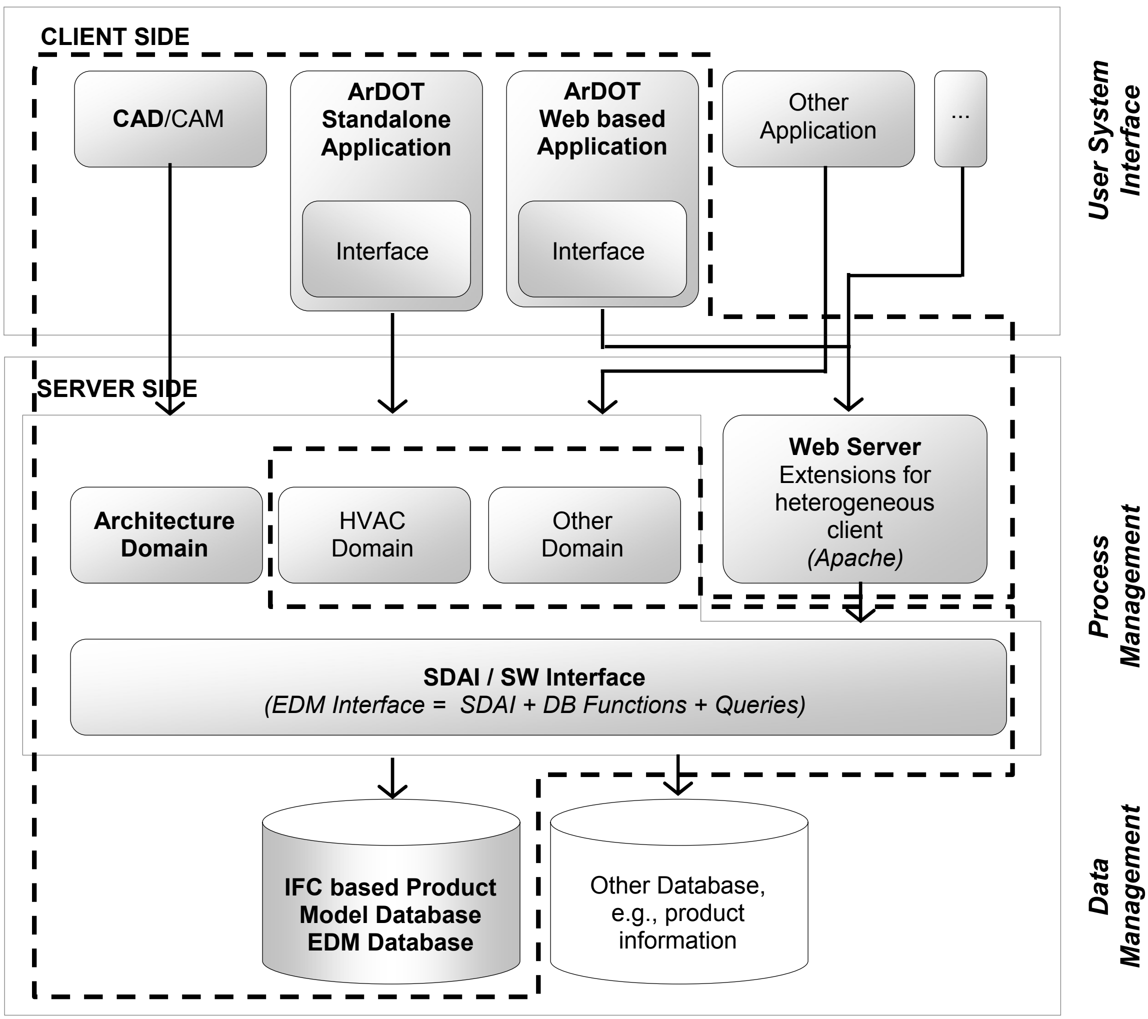


Figure 5.1: System architecture of the implemented framework.

Server-side process management tier contains SDAI (ISO 1998b) interface to connect to IFC based product model database(s). The topological logic resides in this layer which extracts necessary information from the database and serves the requests from client side applications, otherwise known as domain logic. To serve XML/HTTP requests, a web server is connected to the SDAI sub-layer via extensions. Web services can also be implemented via SOAP[4] interfaces to web server. User system interfaces or GUI can either be standalone or web applications.

Representation of domains

The Product Model Database (PMD) stores information about a building project in IFC format. IFC provides access to the domain topology through a layer based architecture. Entity definitions for the concepts specific to individual architecture and HVAC domains can be accessed from the domain layer as discussed in Chapter 3. AEC concepts as implemented in IFC are not yet complete which calls for ad-hoc approaches in mapping domain objects for an integrated AEC activity such as environmental performance simulation or optimisation of environmental design. Not all the objects defined in IFC captures semantics of diverse interpretations by the professionals involved. An example is space, which is the basis of all space-based software environments both for the building design-construction process and the management of facilities (Svensson 1998). The concept of space as implemented in the contemporary product models are shown to be inadequate to cater for the spectrum of applications at different lifecycle stages (Ekholm and Fridqvist 2000), in particular concept development stages (Mourshed *et al.* 2001).

It is envisaged that with time, as IFC matures, some of these hindrances will disappear. It will make seamless integration a reality without the troubles of ad-hoc mapping of objects. Domain applications are dynamic and subject to frequent upgrades because of the changes in the process they support. On the other hand IFC standard needs to be stable enough to encourage development of tools, hence it is rather static. This combination of relatively static standards and dynamic application will require some sort of mapping in the application level necessary, even with the implementation of SW interface between data repository and software applications.

The need to change is driving the AEC industry forward, albeit slowly (Latham 1994; Egan 1998; Dawood *et al.* 2002). New innovations in design computing are constantly changing the ways buildings are designed, constructed and maintained (Mourshed *et al.* 2003a). Mitchell (Mitchell 1994) reported there is a growing consensus that the process of design is treated as a *social activity* - a matter of multiple, autonomous but interconnected intelligences in complex interaction. Examples of such integrated activities in environmental design of buildings are described in (Papamichael *et al.* 1997; Lam *et al.* 2002; Mourshed *et al.* 2006), where decisions are made based on multi-domain performance assessments of proposed designs. Drawing on the experiences gained from implementing an IFC based framework in this thesis, it can be said that for integrated and multiple-domain design assessments, data exchange between an application and an IFC repository need to be on a multi-domain level as opposed to single-domain. For example, in optimisation of

[4] Simple Object Access Protocol is a w3 standard. http://www.w3.org/TR/soap/

building form for energy efficiency at early design stages, data from architectural, HVAC and cost estimating domains is required.

Mapping between IFC and application logic

IFC is originally intended as an overarching specification covering the whole spectrum of concepts related to products or objects in the AEC industry. Exclusion of a process context in the specification has been debated (Augenbroe *et al.* 2004; Bazjanac 2004) and referred to as shortfall of the specification (Augenbroe *et al.* 2004). From the experiences in implementing IFC in a process oriented design task such as optimisation, it can be said that the idea of including process concepts within the specification will only exacerbate the barriers of integration. The aim to achieve interoperability will be hindered by the added complexity of the process aspect, which changes constantly within organisations, partnerships and geographical boundaries. It is, therefore, suggested that the fundamental concepts are implemented using IFC specifications. The process aspects; i.e. the application logic and the tasks involved in a process are left outside the specification.

The framework for this thesis is developed considering the aspects of mapping of IFC objects to application objects and IFC relationships to application logic. Task oriented processes are defined within the application logic. The mapping is explained with the concept of space. From the concepts defined in (Adachi *et al.* 2003), A spatial structure element (`IfcSpatialStructureElement`) is the generalization of all spatial elements that might be used to define a spatial structure in an IFC model. This spatial structure (`IfcSpatialStructureElement`) is often used to provide a project structure to organize a building project. A spatial project structure might define as many levels of decomposition as necessary for the building project. Elements within the spatial project structure are: `IfcSite`, `IfcBuilding`, `IfcBuildingStorey` and `IfcSpace`. Aggregations or parts can as well be within the spatial structure. The composition type declares an element to be either an element itself, or an aggregation or a decomposition. The `IfcRelAggregates` is defined as *1-to-many relationship* and used to establish the relationship between exactly two levels within the spatial project structure. The use of `IfcRelAggregates` to establish a spatial structure including site, building, building section and storey is shown in Figure 5.2.

`IfcSpace` is represented in IFC model as an area or volume bounded actually or theoretically which are areas or volumes that provide for certain functions within a building. A space is associated to a building storey or a site in the case of exterior spaces. A space group can provide for a collection of spaces included in a storey. Partial spaces are defined by decomposing spaces into parts. The following is the EXPRESS specification of `IfcSpatialStructureElement`:

```
ENTITY IfcSpatialStructureElement
ABSTRACT SUPERTYPE OF(ONEOF(IfcBuilding, IfcBuildingStorey, IfcSpace,
                     IfcSite))
SUBTYPE OF (IfcProduct);
 LongName            : OPTIONAL IfcLabel;
 CompositionType     : IfcElementCompositionEnum;
INVERSE
 ServicedBySystems :  SET OF IfcRelServicesBuildings FOR
                      RelatedBuildings;
 ContainsElements  : SET OF IfcRelContainedInSpatialStructure FOR
                     RelatingStructure;
WHERE
 WR41   : (HIINDEX(SELF\IfcObject.Decomposes) = 1) AND
          ('IFCKERNEL.IFCRELAGGREGATES' IN
          TYPEOF(SELF\IfcObject.Decomposes[1])) AND
          (('IFCKERNEL.IFCPROJECT' IN TYPEOF
          (SELF\IfcObject.Decomposes[1].RelatingObject)) OR
          ('IFCPRODUCTEXTENSION.IFCSPATIALSTRUCTUREELEMENT' IN TYPEOF
          (SELF\IfcObject.Decomposes[1].RelatingObject)) );
```

The geometric representation of `IfcSpace` is given by the `IfcProductDefinitionShape` and `IfcLocalPlacement` allowing multiple geometric representations. Further details on `IfcSpace` can be obtained from (Adachi *et al.* 2003; 2004). EXPRESS definition of `IfcSpace`:

```
ENTITY IfcSpace
 SUBTYPE OF (IfcSpatialStructureElement);
  InteriorOrExteriorSpace  : IfcInternalOrExternalEnum;
  ElevationWithFlooring  : OPTIONAL IfcLengthMeasure;
 INVERSE
  BoundedBy     : SET OF IfcRelSpaceBoundary FOR RelatingSpace;
END_ENTITY;
```

Geometric representations are based upon part 42 of STEP specification (ISO 2003b) which specifies the resource constructs for the explicit geometric and topological representation of the shape of a product. Specifications for solid models are the most used methods of representation in IFC. A solid model is defined as the shape of the *IfcProduct*, in which all interior parts are connected. Facetted boundary representation, extruded area solid and constructive solid geometry are the types of solid models used in defining *IfcProduct*. EXPRESS definition of IfcProduct:

```
ENTITY IfcProduct
 ABSTRACT SUPERTYPE OF(IfcProxy)
 SUBTYPE OF (IfcObject);
  ObjectPlacement   : OPTIONAL IfcObjectPlacement;
  Representation   : OPTIONAL IfcProductRepresentation;
 INVERSE
  ReferencedBy   : SET OF IfcRelAssignsToProduct FOR RelatingProduct;
 WHERE
  WR1    : (EXISTS(Representation) AND EXISTS(ObjectPlacement)) OR
           (NOT(EXISTS(Representation))) ;
END_ENTITY;
```

As there are different ways to represent space boundaries and the fact that different vendors do implement IFC differently, IFC export from ArchiCAD has been chosen for implementation in this thesis. IfcSpace is mapped to application logic for which an XML schema has been designed. Documentation of ardML schema is provided in Appendix C.

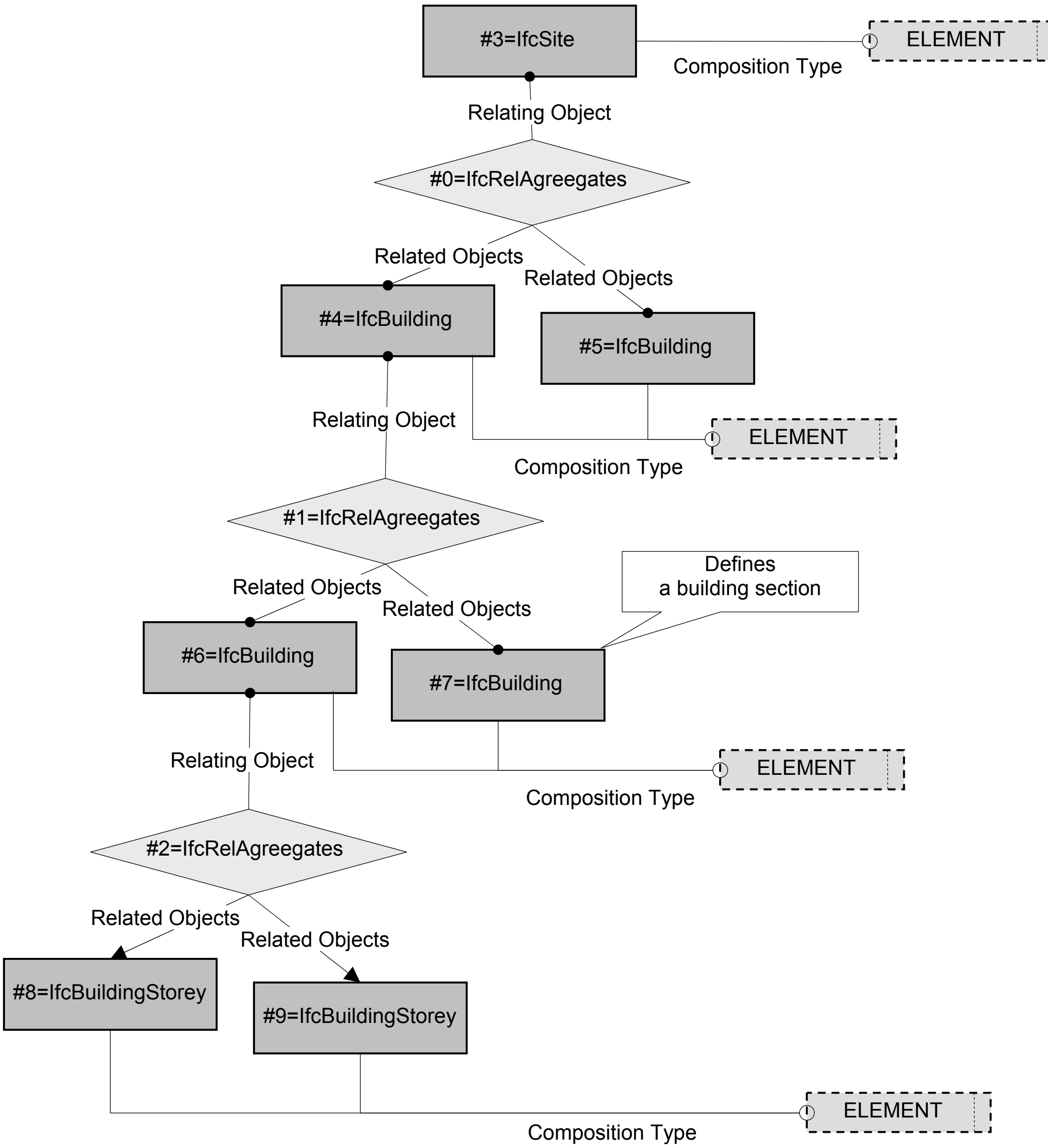


Figure 5.2: The use of IfcRelAggregates to establish a spatial structure, after Adachi et al. (2003).

Development of standalone ArDOT

ardML (**ard**ot X**ML** Shcema) specifies concepts required for optimisation tasks implemented in this thesis. ardML objects come from both the Architectural and HVAC domain of IFC specification as well as other parameters necessary for these objects, not implemented in IFC. Schema diagram of an ardML element - *Material Window Gas* is shown in Figure 5.3.

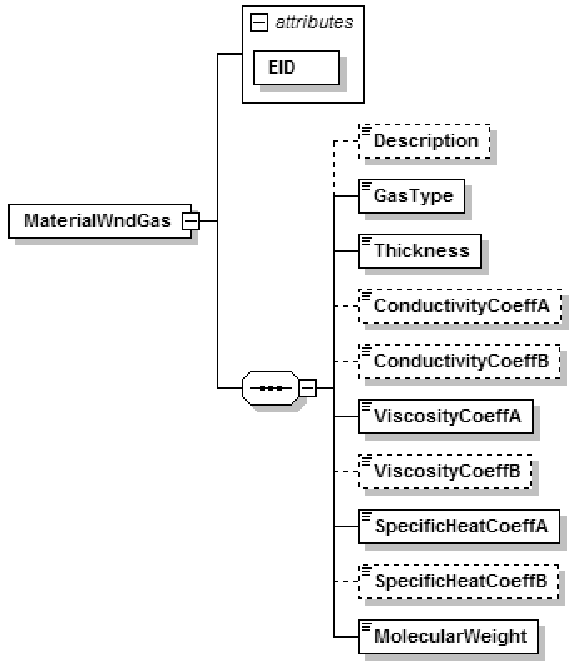


Figure 5.3: Schema diagram for ardML element MaterialWndGas.

Early bound Java class files of ardML - produced using JAXB[5] (Java Architecture for XML Binding), is the skeleton of the Java based standalone ArDOT. C++ version of standalone ArDOT is designed to test versatility of the framework which is named cppArDOT. It includes an optimisation module. cppArDOT does not have a visual interface like Java version of ArDOT. Application logic and workflow for both the versions are identical; hence only Java version is described here. Population of PMD usually starts with the initial sketches done by the architect as shown in Figure 5.5. ArDOT provides a framework for further manipulation of architectural form by combining building simulation and optimisation techniques for informed decision making. Figure 5.6 shows the simulation interface of ArDOT including a dialog box for IFC to IDF conversion. j3d[6], a Java3D API provides 3D graphics rendering in real time for ArDOT. 3D visualisation module of ArDOT is shown

[5] JAXB (Java Architecture for XML Binding), http://java.sun.com/webservices/jaxb/index.jsp

[6] j3D. http://java3d.j3d.org/

in Figure 5.7 where building form can be viewed in either parallel or perspective mode. The UI also allows zooming, rotation and translation of views. System and components of implemented ArDOT is shown in Figure 5.4.

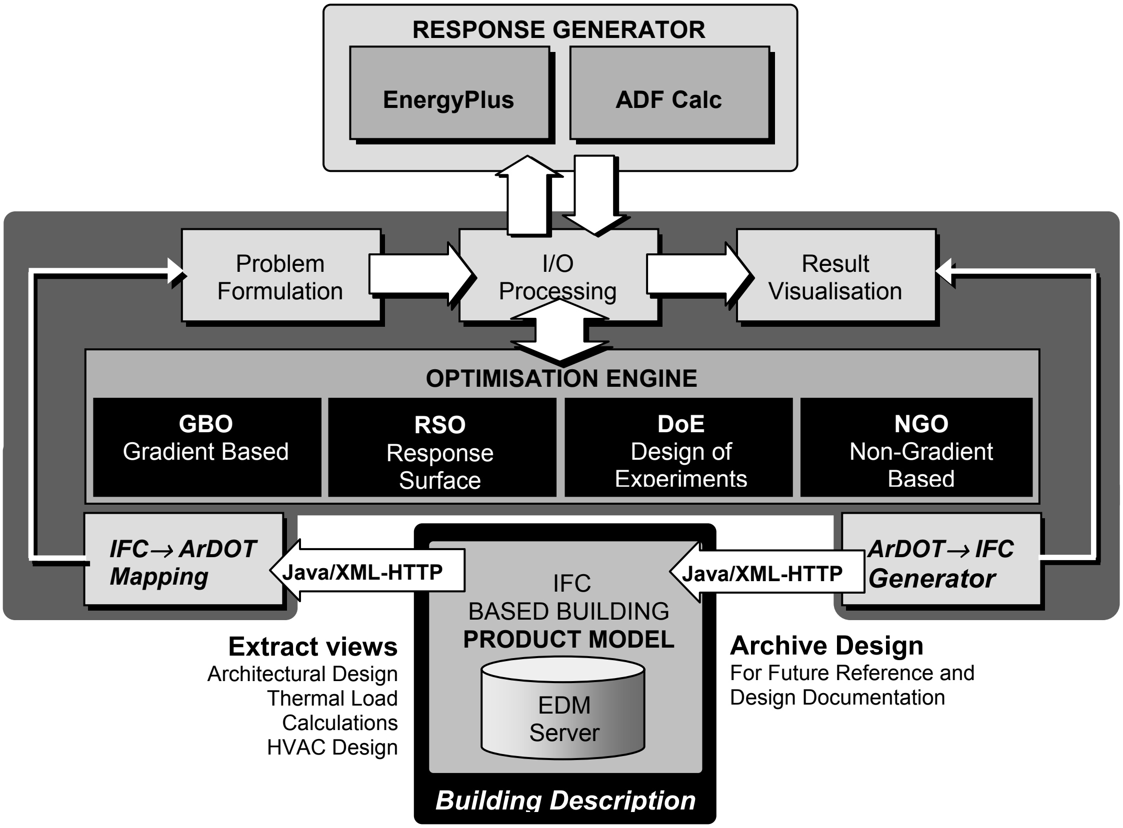


Figure 5.4: ArDOT system and components, as implemented.

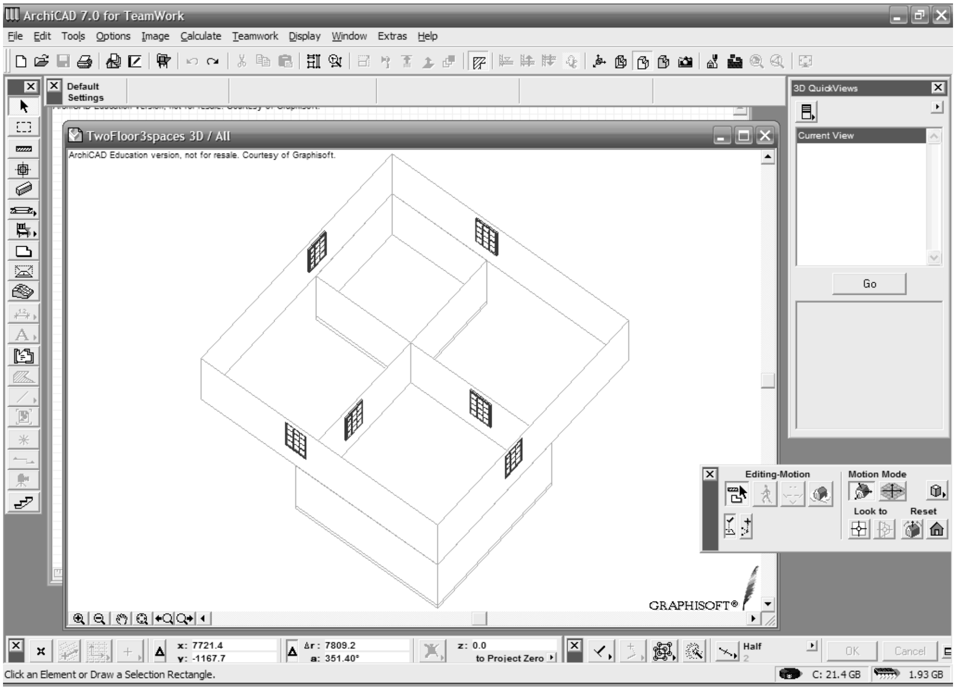


Figure 5.5: Design sketches in ArchiCAD.

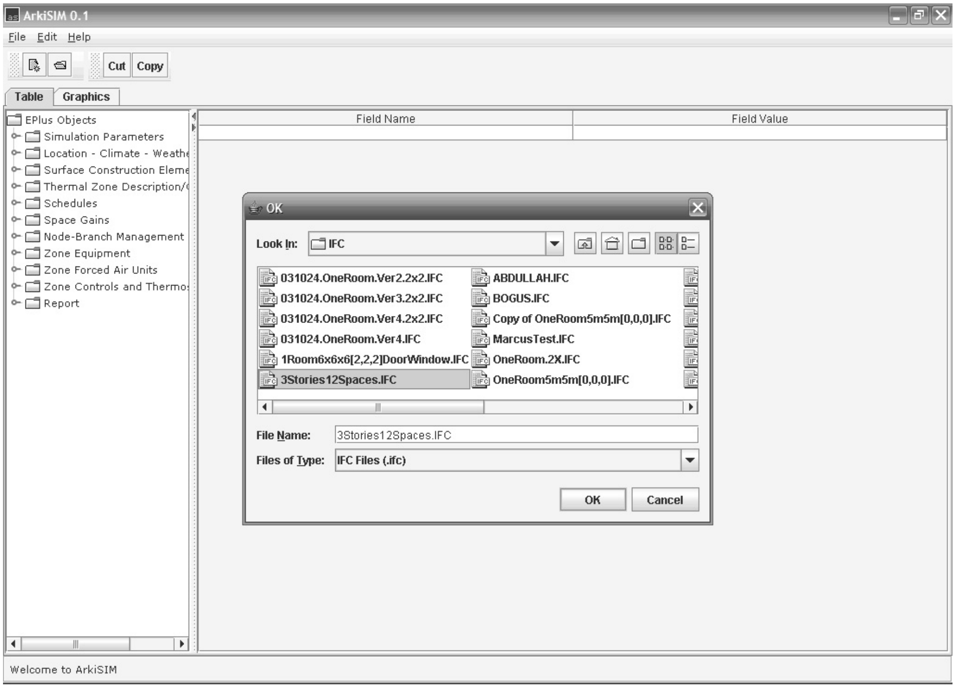


Figure 5.6: Simulation interface of ArDOT with IFC to IDF converter dialog box.

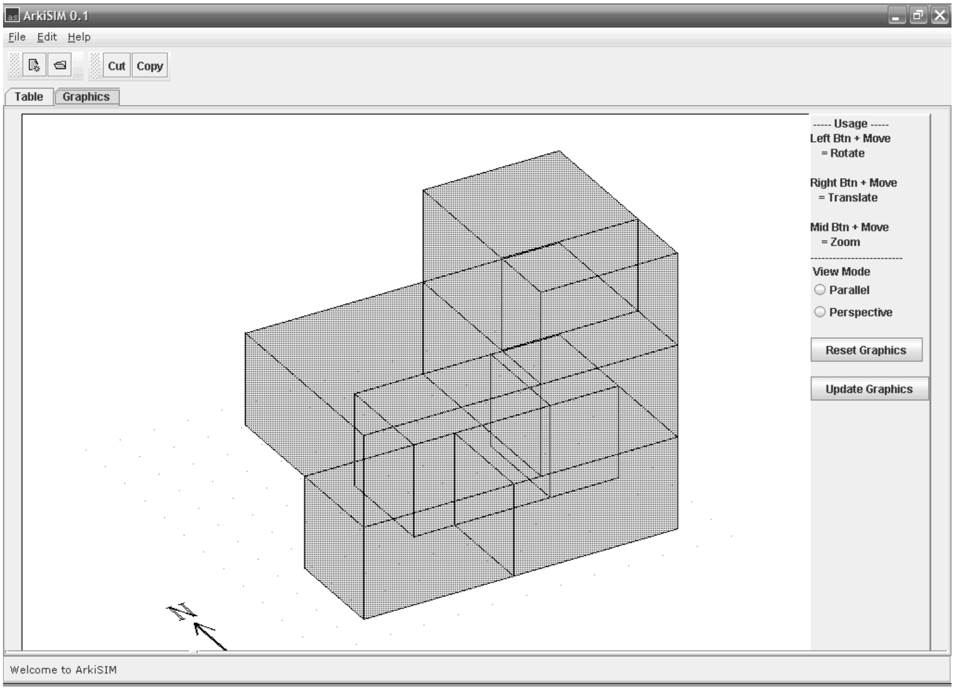


Figure 5.7: 3D visualisation window of ArkiSIM (simulation interface of ArDOT).

Development of web-based ArDOT

For web based client application development Apache HTTP Server[7], an open source web server was added to the process management layer of the framework (see Figure 5.1). A fully web based thin application was developed using PHP as a general purpose scripting language and MySQL as a backend database. Object oriented Javascript[8] and DHTML provide user interaction which allowed development of advanced GUI without the overload of a distributed application usually based on Java. Javascript, developed by Netscape is a superset of ECMA-262 Edition 3 (ECMA Script) standard scripting language. Dynamic HTML (DHTML) is essentially a marketing term applied to a mixture of standards including HTML, style sheets, Document Object Model (DOM) and scripting. GUI of ArDOT as shown in Figure 5.8 has the usual Human Computer Interaction (HCI) features such as menus, tree based browser, toolbars and a status bar - usually found in the mainstream applications. The implementation is based on the java based simulation module to edit and manipulate simulation programmes. All the files (input and output) and the application logic resides in the server. User is only presented with the results through a web browser window.

[7] Apache HTTP Server project. http://httpd.apache.org

[8] Javascript 1.1 specification available from: http://wp.netscape.com/eng/javascript/

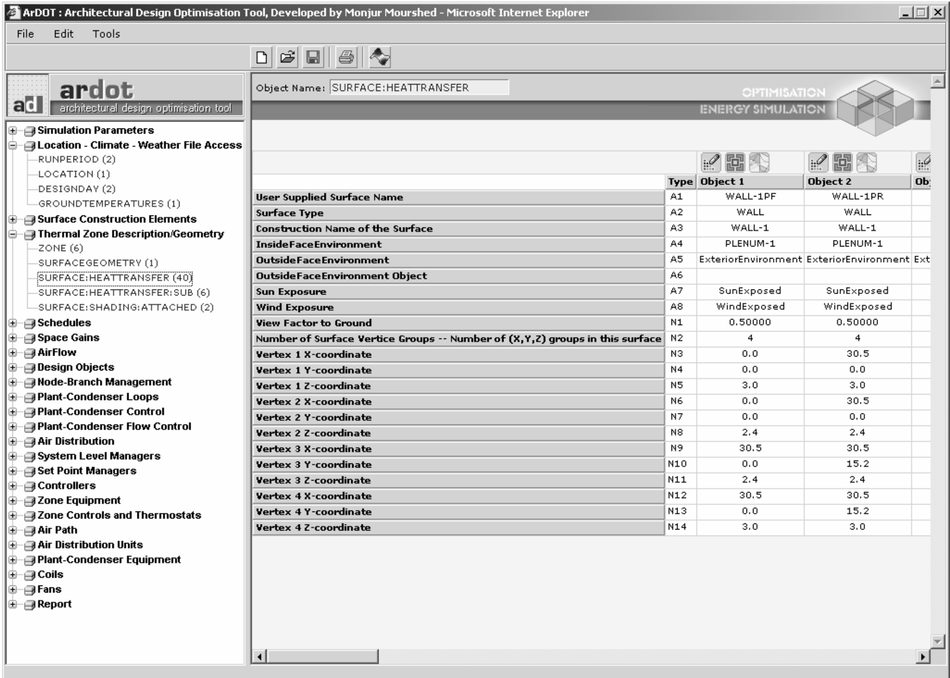


Figure 5.8: GUI of web based ArDOT with object information window on the right.

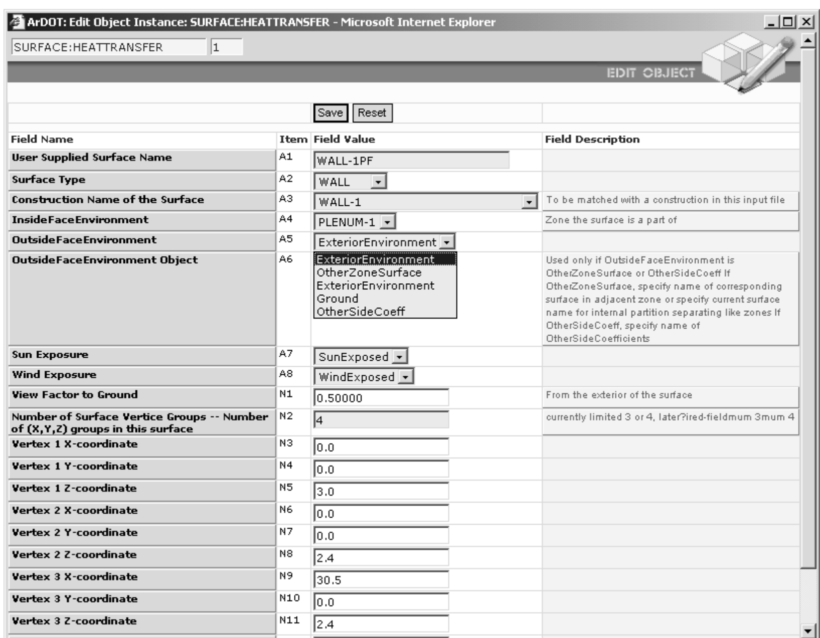


Figure 5.9: Editing an object in web based ArDOT.

EnergyPlus is implemented in this application. Figure 5.8 shows EnergyPlus IDF objects on the left including their numbers and object parameters on the right as a table view. Editing of an object is done in the edit window (see Figure 5.9). Setting up of optimisation problems are envisaged to be done from optimisation window (Figure 5.10), invoked from the menu or the process toolbar.

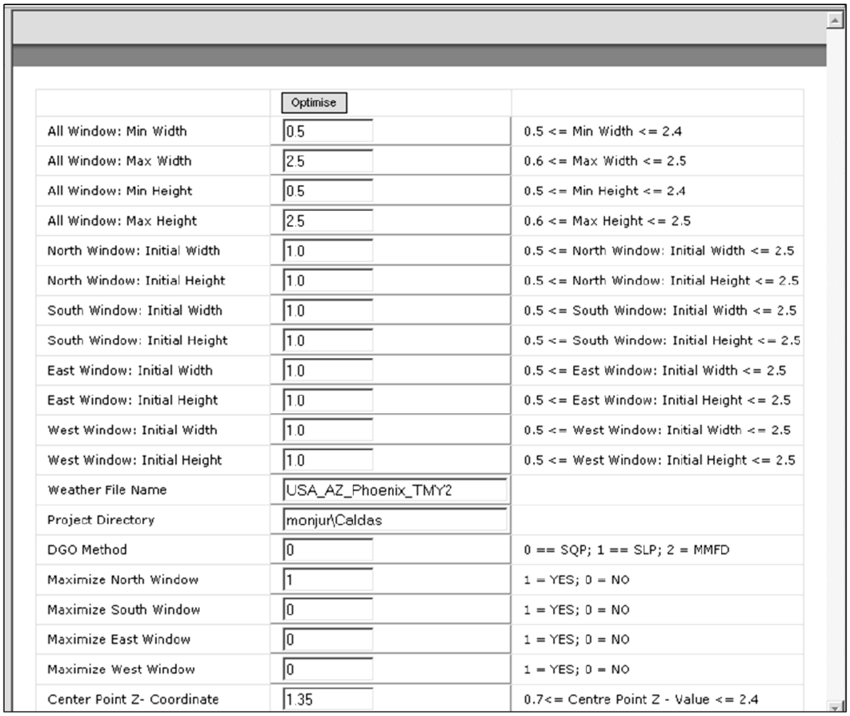


Figure 5.10: Optimisation window in web based ArDOT.

Methodology for implementation

In order to ascertain the effectiveness of the proposed framework in integrating multi-domain tasks and decision making process, environmental design of buildings has been chosen as the domain for implementation. Early design period is the selected life-cycle stage. Selected tasks involve finding the form, orientation and indicative envelope characteristics of a proposed building based on the concepts of energy efficiency in buildings. The specialist user in this case is an architect engaged in the qualitative exploration of design solutions. Optimisation algorithms assist the user by executing a mathematically directed search of the solution space. Simulation tools from within the framework provide the architect with multi-domain knowledge in the form of simulation results which drive the optimisation process.

Environmental design as the domain of discourse

Environmental design of buildings involves multi-disciplinary decision making at each life-cycle stages. Buildings are not usually, and never specifically, designed solely for the purposes of reduction of fossil fuel consumption; rather they seek to provide an environment in which occupants can perform a loosely defined set of activities (Norton *et al.* 1996). Team of professionals involved in making decisions regarding environmental design consider a set of interdependent environmental factors in an integrated fashion. Buildings can thus incorporate any combination of passive and active mechanisms to render energy efficiency. Substantial work has been undertaken in developing analytical and numerical models for the diverse range of environmental aspects in energy efficient building design. Software implementations of these models are known as simulation engines. Simulation tools, as incorporated in the design process, are ad hoc and offer only a piecemeal solution to the problem which undoubtedly requires an integrated approach - as described in (Papamichael *et al.* 1997; Lam *et al.* 2002; Mourshed *et al.* 2003a; Augenbroe *et al.* 2004).

As an inter-disciplinary activity, environmental design of buildings has been chosen as the domain of discourse for implementation in this thesis. The decision for selection of domain was governed by the availability of industry standard simulation tools which had undergone decades of development (Hensen *et al.* 2002; Malkawi 2004). Based on whole-building simulation concepts, these matured tools predict building performance to a greater level of accuracy. The implementation demonstrates vertical integration among applications from a single domain as well as provides mechanisms for cross-domain horizontal integration, as in Figure 5.11.

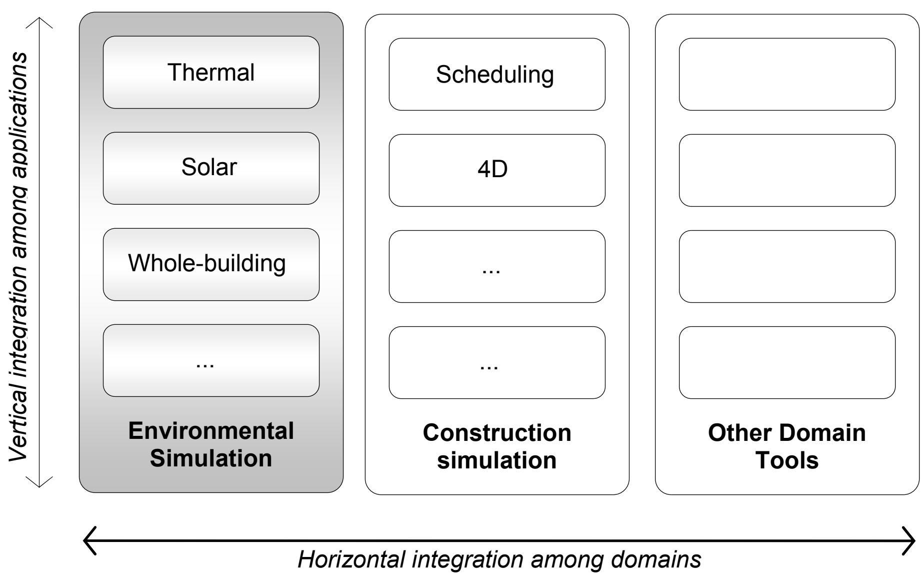


Figure 5.11: Integration of domains and applications.

Optimisation as design activity

Optimisation methods, as introduced in chapter 1, reviewed and described in Chapter 4 can assist in decision making provided that the problem domain is modelled to a reasonable extent of accuracy in simulation/analysis tools. Whole-building energy simulation tools; e.g. EnergyPlus use adaptive solvers such as Newton solvers or variable time step integration routines to compute an approximate numerical solution to a complex system of equations including implicit equations, ordinary differential equations and partial differential equations (Wetter and Wright 2004). Success of the deployment of such tools in simulation based optimisation of architectural design, particularly during concept development stages depends on the formulation of the problem.

Environmental design of buildings, involves 'finding the optimum' solution satisfying predefined objective(s); e.g. reduction in operating/capital cost, maximisation of daylighting etc. Deployment of mathematical optimisation techniques is thus a natural progression for such sets of design activities. Unlike aesthetics, problems in environmental design domain can be modelled and simulated using analytical/numerical methods. Optimisation methods are employed in this thesis taking full advantage of building simulation tools in predicting environmental performance of buildings. The main benefit obtained is the mathematical exploration of the solution space as opposed to the conventional practice of occasional validation of the design proposals.

Design activities chosen for optimisation of environmental design are the selection of form, orientation and percentage glazing on cardinal sides of a rectangular building. Design goals translated into optimisation objectives are: reduction of annual energy consumption and average daylight factor of a specific quantity. Optimisation activities relate to that of design exploration during concept development stage of a building. Motivation behind this was to ascertain the effectiveness of the framework in making early architectural design decisions concerning energy efficiency. The application of gradient based optimisation algorithms in searching concept designs was investigated.

Description of the system for implementation

Environmental design in architecture requires an integrated approach considering the domains of thermal, visual, and acoustic, which have effects on human comfort. EnergyPlus (Crawley *et al.* 2001) has been chosen as the simulation engine to model environmental design problems for testing the framework. EnergyPlus is a versatile simulation engine capable of modelling loads and annual energy use for entire building. The accuracy of EnergyPlus has been validated against other building energy programmes using the BESTEST method (Witte *et al.* 2001). To demonstrate the versatility of the framework in combining multiple response generators, a separate hard-coded module by the author has been added to compute average daylight factor (DETR 2002) of the proposed designs. The tool to compute average daylight factor is hereafter called ADF Calc. The system as implemented in this thesis is shown in Figure 5.12.

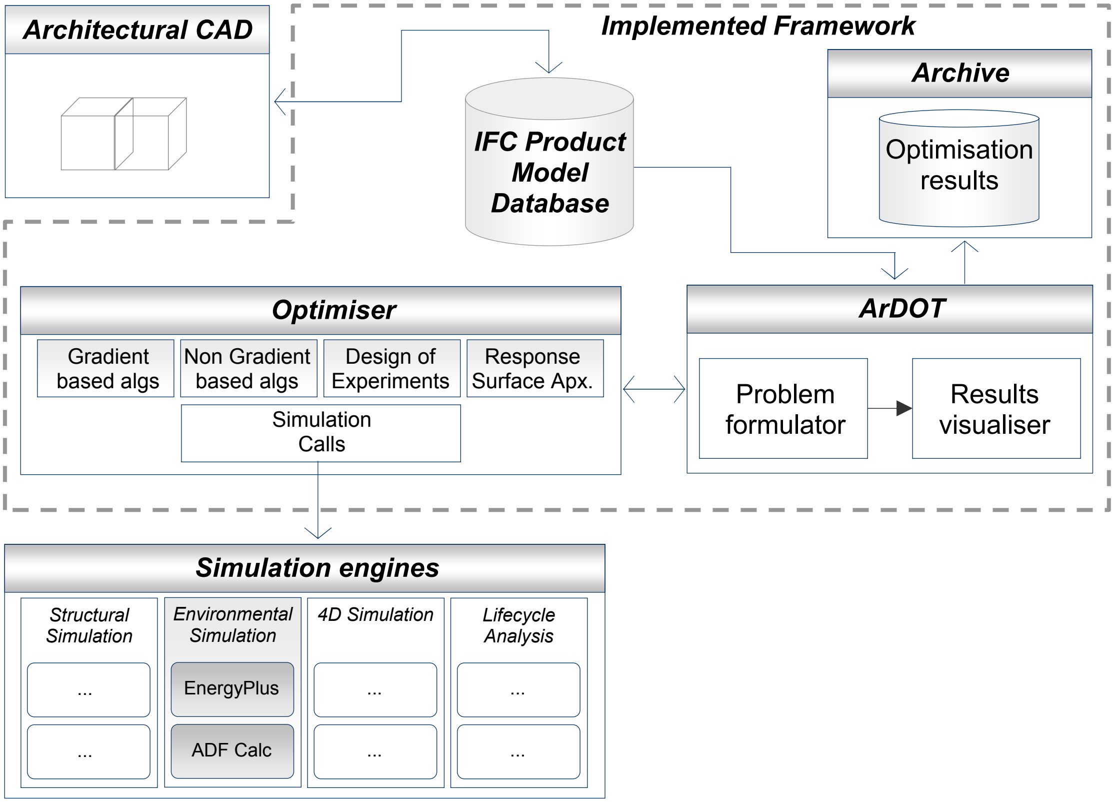


Figure 5.12: The framework, as implemented in this thesis.

Workflow for implementation begins with the architect sketching ideas in an IFC enabled architectural CAD programme such as ArchiCAD or Revit. ArchiCAD[9] v7.0 is used in this thesis. The drawing is then exported as SPF. ArDOT (**Ar**chitectural **D**esign **O**ptimisation **T**ool), developed as part of the integrated framework in this thesis imports the IFC file into predefined PMD (Product Model Database). The PMD is implemented by integrating EDM Database (EPM 2000) through Java API. IFC2x Edition 2 (Adachi *et al.* 2004), released in May 2003 has been implemented and remains constant throughout the thesis. ArDOT extracts relevant information from the PMD and lets the user choose optimisation variables, objectives and constraints in VisualDOC (VR&D 2001) which is integrated within the framework. The connection between PMD and ArDOT is via Java based SDAI interface (ISO 1998a), developed as part of this thesis. SQP (Sequential Quadratic Programming), discussed in Chapter 4, has been chosen as the optimisation algorithm for implementation in this thesis.

After the optimisation problems are defined, VisualDOC takes precedence. Communication between the optimiser and ArDOT is accomplished via ASCII based transfer files. The optimiser makes

[9] http://www.graphisoft.com

subsequent calls to ArDOT to translate optimisation variables into input files suitable for simulation engines required for the chosen problem domains. Input file formats for simulation include IDF and input for ADF Calc. Mapping of IFC definitions into energy simulation input files is implemented on an ad hoc basis which would not have been required if energy simulation programmes were modelled using IFCs as their data model. Results from simulation programmes, in this case, EnergyPlus and ADF Calc, guide the optimisation process. As gradient based algorithms are implemented in this thesis, gradients are calculated by VisualDOC from successive analysis calls made to EnergyPlus and ADF Calc. Results obtained from VisualDOC are visualised and analysed in ArDOT and archived in a task oriented file based system.

Simulation method: EnergyPlus as response generator

After the design task has been defined as an optimisation activity, the optimiser takes care of the initiation of the simulation of the proposed scheme. Year-round energy simulation of the proposed scheme, performed by EnergyPlus determines objective values and gradients. Simulations are performed in 15 minutes time-step for greater accuracy of results and are driven by location specific hourly weather data in EPW format. As weather data is based on statistical interpolation and different simulation programmes use different types of data - simulation results from two different engines may not produce the same result (Enshen 2005). Optimisation results should therefore be considered as dependent on the accuracy and efficiency of the simulation engines.

Typical weather data for energy simulation contains dry-bulb temperature, air humidity, wind velocity, total solar radiation and direct solar radiation (Enshen 2005). Considering the random variation of weather year after year, weather data contains typical year meteorological conditions, not the actual recording data (Zhang *et al.* 2002). TRY and TMY are two kinds of weather data used predominantly in Europe and the United States respectively. EnergyPlus weather file, EPW - embodies other information from the location and weather data, e.g. design conditions, calculated ground temperatures, typical and extreme weather periods in addition to TMY2[10] data (DOE 2004b). The TMY2 dataset is made up of hourly values of measured or modelled solar radiation and meteorological data for a particular location for the 30-year period from 1961-1990. The typical months selected from individual years are concatenated to form a complete year. A TMY provides a standard for hourly data for solar radiation and other meteorological elements that permit performance comparisons of system types and configurations for one or more locations (Marion and Urban 1995). EnergyPlus, therefore, produces responses of reasonable resolution for use in mathematical optimisation.

[10] The new TMY data, compiled in 1994 is known as TMY2 and derived from the 1961-1990 National Solar Radiation Data Base (NSRDB) in the United States.

Interpolation of simulation parameters

EnergyPlus, a detailed based simulation programme requires detailed parameters to be known for environmental performance assessment. Materials and construction of building elements and related attributes used in this implementation are based on standardised components such as standard cavity wall, insulations etc. For optimisation of building form, orientation and envelope, the adoption of standards-based approach is quite common, even when building simulation is not used. Progressive detailing is evident in architects' works, right from inception as parameters for detailed analysis are not known until later. Other types of loads such as lights, people and electric equipment are based on CIBSE (The Chartered Institution of Building Services Engineers) and ASHRAE (American Society of Heating, Refrigeration and Air-Conditioning Engineers) standards (CIBSE 2003; ASHRAE 2005).

Modelling daylighting in EnergyPlus

Daylighting systems, i.e. windows, skylights, roof monitors and atria display a complex environmental behaviour due to different requirements often being in conflict with each other, making the evaluation of their overall performance a complex one (Caldas 2001). With the increase in window sizes during the heating season, there is a corresponding increase in daylighting availability and a decrease in the need for artificial lighting, leading to reduced electricity consumption. The same window may increase cooling loads during summer months through window heat gain resulting in increased energy consumption. Depending on the climate of the location, bigger window sizes for some orientation may provide useful solar gains through the window which tends to reduce heating expenditure. There may also be heat loss through the glazing, which in turn increases heating loads. Less use of artificial lighting tends to lower internal heat gains in the building, leading to an increased need of heating.

The complex interactions between daylighting systems and the corresponding increase/decrease in energy consumption are simulated using the detailed daylighting model provided by EnergyPlus. Daylighting level at the centre of each space is dynamically calculated at a reference point of the architect's choosing, for each hour of the year. For implementation purposes, the reference point is located at the centre of the space. The Illuminance level achieved by using natural light only is then compared to the required levels in lux specified by the architect as a parameter in the detailed daylighting specifications. The continuously dimmable artificial lighting system is set to provide just enough light to make up for the difference between available daylighting and the required light level. Electrical energy consumed by the lights is then calculated. Heat gains due to electrical lighting are also accounted for while calculating space heat gain.

Illuminance setpoint at the reference point is set to be 550 lx which controls the whole zone for replacing artificial lighting. Continuous dimmable artificial lights are used. Maximum discomfort glare index for window shade control is 22.0 where as minimum input power and light output fractions for continuous dimming control are 0.3 and 0.2, respectively. Design lighting level of 15 Wm^{-2} for artificial lighting is used, based on ASHRAE guidelines. Radiant and visible fractions of artificial lighting are set at 0.2. 80% of the artificial lighting can be replaced with daylighting when adequate daylight levels are reached at the reference points.

Modelling the HVAC system

All the case study design problems use the ideal air system model, called *purchased air* in EnergyPlus. The *purchased air* model in EnergyPlus can be thought of as an ideal unit that mixes air at the zone return condition with the specified amount of outside air and then adds or removes heat and moisture at 100% efficiency in order to produce a supply air stream at the specified conditions (DOE 2004c). Examples of the use of purchased air as HVAC system can be found in (Sankaranarayanan 2001; Griffith *et al.* 2003; Mourshed *et al.* 2003b). The main reason for the selection of an ideal HVAC air system; i.e. *purchased air* model in EnergyPlus is that the model is suitable for extracting heating and cooling load without specifying a detailed HVAC system. Design tasks at early architectural design stage concerns with finding solutions for building form, function and fabric that minimise loads and energy use (Griffith *et al.* 2003) than fine tuning HVAC systems. Specification of the system and selection of parameter for HVAC is usually performed at later stages. Moreover, using an HVAC system for preliminary exploration of solution space may introduce noise, which in effect can make optimisation algorithms getting trapped at local minima. This is because the simulation programmes; i.e. EnergyPlus use iterative solvers that iterate until convergence criterion for comfort conditions within a space are met; also known as system sizing.

Practical difficulties arise when using whole building energy simulation programmes like EnergyPlus because of the complexity of HVAC system models and the time required to create input for them. Also at early stages, the required parameters for full HVAC simulations may not be known to the user. Models with detailed HVAC systems may also show significant additional energy use because of non-ideal control situations where cooling and heating components work against each other, as in terminal reheat units (Griffith *et al.* 2003). It is therefore, useful to use ideal HVAC air system models to better normalise the effect of energy efficiency improvements that are not part of the HVAC system during the optimisation process.

Purchased Air with the outside air option is used in this thesis. Autosize option for outside air flow has been selected to allow EnergyPlus to calculate the rate based on zone sizing inputs. Outside air is mixed with the zone return air to make up the heating or cooling supply air stream. The *purchased air* component is operated with infinite heating and cooling capacity. Heating and cooling supply air temperatures are 50ºC and 13 ºC respectively. A supply air humidity ratio of 0.009 kg-H_2O/kg-air is used for both heating and cooling.

Calculating the Average Daylight Factor

To demonstrate horizontal integration among simulation tools from a single domain, a separate simulation programme called ADF Calc (Average Daylight Factor Calculator) has been developed. The purpose of this tool is to compute Average Daylight Factor according to the rule of thumb proposed by DETR (2002). The tool takes in geometry of the building including glazing area and produces ADF for the design solution. ADF as a guiding principle for design exploration has been modelled as constraint on the optimisation problem in this thesis.

Daylighting factor

Daylight is the diffuse light of the overcast sky which is similar in all orientations and is soft and cool in both temperature and colour. On the other hand sunlight is the direct rays of the sun. It is directional, warmer in colour and temperature, piercing and very strong. It is suggested that a significant part of lighting inside a building should come from daylighting even though increasing glazed areas may contribute to heating or cooling loads. Daylit spaces are preferred by users to live and work in - the reasons behind this are uncertain. It is reported to be a likely combination of aesthetic, psychological and practical factors which together bring an unquantifiable, additional dimension to an environment (Loe and Mansfield 1998). Windows and rooflights offer contact with the outside world, whether internally within a building as in an atrium - or externally. There are also anecdotal evidences of benefits for spaces with views. In some climates, daylighting may significantly reduce energy consumption predominantly through either supplementing or replacing electrical lighting. Lighting and its associated cooling energy use constitute 30-40% of a commercial building's energy use (O'Connor *et al.* 1997). Daylighting is sometimes the most cost effective strategy for reducing annual operating and mechanical system first costs.

Daylighting even to a small extent can make an interior appear daylit even if the task lighting is predominantly electric. It is reported to occur when the main room surfaces receive enough daylight for the room to appear 'light'. DETR (2002) states that "a room can have a daylit appearance if the area of glazing is at least $1/25^{th}$ of the of the total room surface area (floor, ceiling and walls; including the windows). This is based on the assumption that the room is approximately rectangular in plan and that there are no factors that significantly reduce the amount of light in the space such as dark room surfaces, low transmittance glazing or high external obstructions. This rule of thumb is based on achieving an average daylight factor of at least 2% at table-top level in the room".

The *daylight factor* is the illuminance received at a point from a sky of known or assumed luminance distribution expressed as a percentage of the horizontal illuminance outdoors from an unobstructed hemisphere of the same sky, excluding direct sunlight. The *daylight factor* at a specific point within a zone consist of two components - the direct component entering directly through the window and the indirect component reflected from internal surfaces. The direct component can be considered to consist of a further two components - a sky component and a reflective component from outdoor sources. *Daylight factor*s are used in this thesis in the EnergyPlus simulation runs to determine the displaced artificial illuminance within a zone, which in turn controls the reactive lighting system within the simulations.

Average Daylight Factor

Daylight factor can be a valuable design tool, particularly at the early design stages, as mentioned in (Loe and Mansfield 1998; DETR 2002) through the use of a simplified design method developed by the Building Research Establishment which is implemented in ADF Calc. The method relates an Average Daylight Factor, DF_{avg}, to the glazed area within the zone. This factor is defined as the ration of a room's average internal illuminance at the working plane to that of the external global horizontal illuminance, expressed as a percentage of standard overcast sky conditions:

$$DF_{avg} = \frac{\sum (W\tau\theta M)}{A(1-R^2)} \quad \textbf{(5.1)}$$

where:

DF_{avg} is the Average Daylight Factor,

W is the are of each window (m²),

τ is the transmittance of each glazing,

θ is the vertical angle of visible sky, measured from the centre of the window opening in the plane of the inside window wall,

M is the maintenance factor based on angle of glazing and cleanliness,

A is the total internal surface area of the space (m²), and

R is the area weighted average reflectance of all surfaces making up A.

Summary

Development of the framework based on precedent analysis has been presented in this chapter. Three tier information system has been chosen as the architecture for the framework to provide flexibility, maintainability, reusability and scalability. IFC based product model database is the central repository for the project data allowing disparate systems to collaborate. AEC processes are dynamic and vary among organisations and geographical boundaries. The advantage of product centric PMD over a process approach is shown. From an application development context, it is shown that accomplishing AEC tasks also varies within applications from the same domain. Incorporation of integrated performance assessment and new techniques such as multi-domain optimisations requires data from several domains. This makes mapping between application logic and domain layer of IFC mandatory. ardML, an XML schema has been developed for mapping of IFC objects into ArDOT objects.

To demonstrate the versatility of the framework, two groups of software have been developed keeping the application logic the same. The first one is a Java based standalone application, called ArDOT allows architects to perform optimisation of environmental design of buildings. ArDOT contains Java GUI as well as 3D visualisation module. There is a C++ version of ArDOT without a GUI which has hardcoded optimisation algorithm based on VisualDOC's C API. The second application is developed as a thin web based dynamic application using PHP and MySQL. Both the data and the application logic resides in the server demonstrating the possibility of platform independent lightweight software development.

Methodologies for implementation of the framework in solving some environmental design tasks have been discussed here. Environmental design of buildings has been chosen as the domain of discourse because of the widespread availability of robust industry standard solutions. Optimisation of the environmental design of buildings is the chosen task to demonstrate the adaptability of the framework in implementing new processes. This framework goes a step further from the contemporary integrated framework by adopting optimisation, which is still in the theoretical domain, as a design task. A method of interpolation has been used to use detailed-based simulation programmes during early stages of design in which detailed parameters are not known until later.

The chapter also discussed modelling for simulation, in particular HVAC and daylighting. Development of ADFCalc, a rule of thumb based simulation program for computing Average Daylight Factor has been discussed. Methodologies of implementation discussed in this chapter are applied and tested in Chapter 6.

References

Adachi, Y., Forester, J., Hyvarinen, J., Karstila, K., Liebich, T. and Wix, J. (2003). *IFC2x Edition 2*. Washington: International Alliance for Interoperability.

Adachi, Y., Forester, J., Hyvarinen, J., Karstila, K., Liebich, T. and Wix, J. (2004). *IFC2x Edition 2 Addendum 1*. Washington: International Alliance for Interoperability.

Alonso, G., Kuno, H., Casati, F. and Machiraju, V. (2003). *Web Services: Concepts, Architectures and Applications*. Berlin.

ASHRAE (2005). *ASHRAE Handbook: Fundamentals*. Atlanta, GA: American Society of Heating Refrigeration and Air-Conditioning Engineers.

Augenbroe, G., Wilde, P.d., Moon, H.J. and Malkawi, A. (2004). An interoperability workbench for design analysis integration. *Energy and Buildings*, **36**(8), 737-748.

Bazjanac, V. (2004). Building energy performance simulation as part of interoperable software environments. *Building and Environment*, **39**(8), 879-883.

Caldas, L.G. (2001). *An Evolution-Based Generative Design System: Using Adaptation to Shape Architectural Form*. PhD Thesis. Cambridge, MA: Massachusetts Institute of Technology.

CIBSE (2003). *CIBSE Concise Handbook*. London: The Chartered Institution of Building Services Engineers.

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl, W.F., Huang, Y.J., Pedersen, C.O., Strand, R.K., Liesen, R.J., Fisher, D.E., Witte, M.J. and Glazer, J. (2001). EnergyPlus: creating a new-generation building energy simulation program. *Energy and Buildings*, **33**(4), 319-331.

Dawood, N., Akinsola, A. and Hobbs, B. (2002). Development of automated communication of system for managing site information using internet technology. *Automation in Construction*, **11**(5), 557-572.

DETR (2002). *Desktop Guide to Daylighting - for Architects*. 245. Department of Environment, Transport and Regions, United Kingdom.

DOE (2004a). *Getting started with EnergyPlus*. Berkeley: Department of Energy.

DOE (2004b). *Input Output Reference: The Encyclopedic Reference to EnergyPlus Input and Output*. Berkeley: Department of Energy.

Egan, J. (1998). *Rethinking Construction*. London: DETR.

Ekholm, A. and Fridqvist, S. (2000). A concept of space for building classification, product modelling, and design. *Automation in Construction*, **9**(3), 315-328.

Enshen, L. (2005). General descriptions of research method, buildings, analysis object, assumptions and weather conditions of background cities. *Building and Environment*, **40**(4), 443-450.

EPM (2000). *EDMassist Volume I*. Oslo: EPM Technology.

Faraj, I., Alshawi, M., Aouad, G., Child, T. and Underwood, J. (1999). Distributed Object Environment: Using International Standards for Data Exchange in the Construction Industry. *Computer-Aided Civil and Infrastructure Engineering*, **14**(6), 395-405.

Griffith, B., Pless, S., Talbert, B., Deru, M. and Torcellini, P. (2003). *Energy Design Analysis and Evaluation of a Proposed Air Rescue and Fire Fighting Administration Building for Teterboro Airport*. NREL/TP-550-33294. Colorado: National Renewable Energy Laboratory.

Hensen, J.L.M., Lamberts, R. and Negrao, C.O.R. (2002). A view of energy and building performance simulation at the start of the third millennium. *Energy and Buildings*, **34**(9), 853-855.

ISO (1998a). *Industrial automation systems and integration - Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure*. ISO10303-21:1998. Geneva: International Organisation for Standardization.

ISO (1998b). *Industrial automation systems and integration - Product data representation and exchange - Part 22: Implementation methods: Standard data access interface*. ISO 10303-22:1998. Geneva, Switzerland: ISO.

ISO (2003). *Industrial automation systems and integration - Product data representation and exchange - Part 42: Integrated generic resource: Geometric and topological representation*. ISO 10303-42:2003. Geneva, Switzerland: ISO.

Keane, M. and Kelliher, D. (2001). Green building design management. *International Journal of Architectural Management, Practice and Research*, **16**, 263-271.

Lam, K.P., Mahdavi, A., Gupta, S., Wong, N.H., Brahme, R. and Kang, Z. (2002). Integrated and distributed computational support for building performance evaluation. *Advances in Engineering Software*, **33**(4), 199-206.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004). SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. *Automation in Construction*, **13**(5), 651-663.

Langer, A.M. (2000). *Analysis and Design of Information Systems*. New York: Springer.

Latham, M. (1994). *Constructing the Team: Final Report to the Government/Industry Review of Procurement and Contractual Arrangements in the United Kingdom Construction Industry*. London: HMSO.

Loe, D. and Mansfield, K.P. (1998). *Daylighting Design in Architecture: Making the Most of a Natural Resource*. Garston, Watford: Building Research Establishment.

Malkawi, A.M. (2004). Developments in environmental performance simulation. *Automation in Construction*, **13**(4), 437-445.

Marion, W. and Urban, K. (1995). *User's Manual for TMY2s: Typical Meteorological Years*. Golden, Colorado: National Renewable Energy Laboratory.

Mitchell, W.J. (1994). Three paradigms for computer-aided design. *Automation in Construction*, **3**(2-3), 239-245.

Mourshed, M.M., Kelliher, D. and Keane, M. (2001). Spatial representation in product modelling. In: *IV2001: IEEE Conference on Information Visualisation*, London, UK, IEEE Computer Society, 651-656.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003a). Integrating Simulation in Design. *IBPSA News*, **13**(1), 21-26.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003b). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

Mourshed, M.M., Kelliher, D. and Keane, M. (2006). Green architecture: the need for climate analysis and thermal simulation during early stages of design. *Global Built Environment Review*, **5**(2), XX-XX.

Norton, B., Lo, S.N.G., Eames, P.C. and Griffiths, P.W. (1996). Optimising solar energy in buildings via data sharing during design. *Renewable Energy*, **9**(1-4), 708-713.

O'Connor, J., Lee, E., Rubinstein, F. and Selkowitz, S. (1997). *Tips for Daylighting with Windows*. Berkeley: Lawrence Berkeley National Laboratory.

Owolabi, A., Anumba, C.J. and El-Hamalawi, A. (2003). Architecture for implementing IFC-based online construction product libraries. *Electronic Journal of Information Technology in Construction*, **8**, 201-218.

Papamichael, K., LaPorta, J. and Chauvet, H. (1997). Building Design Advisor: automated integration of multiple simulation tools. *Automation in Construction*, **6**(4), 341-352.

Sankaranarayanan, K.P. (2001). *Modeling, Verification and Optimization of Hybrid Ground Source Heat Pump Systems in EnergyPlus*. MS Thesis. Stillwater: Oklahoma State University.

Svensson, K. (1998). *Integrating facilities management information*. PhD Thesis. Stockholm: Ronal Institute of Technology, Stockholm.

VR&D (2001). *VisualDOC: Getting Started Guide*. Colorado Springs: Vanderplaats Research & Development.

Weijia, J. and Zhou, W. (2004). *Distributed Network Systems: From Concepts to Implementations*. New York: Springer.

Wetter, M. and Wright, J. (2004). A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment*, **39**(8), 989-999.

Witte, M.J., Henninger, R.H. and Glazer, J. (2001). Testing and Validation of a New Building Energy Simulation Program. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 353-360.

Yang, Q.Z. (2003). IFC-compliant design information modelling and sharing. *Electronic Journal of Information Technology in Construction*, **8**, 1-14.

Zhang, Q., Huang, J. and Siwei, L. (2002). Development of typical year weather data for Chinese locations. *ASHRAE Transactions*, **108**(2), 1063-1075.

Chapter 6
Results and discussions

Results from the implementation of the framework in optimisation of environmental design of buildings are described and analysed in this chapter. The role of pre-design climate analysis in the selection of building form and envelope is discussed. Some selected design problems related to energy efficiency at early architectural design stage are tested with the implemented framework in two locations with distinct climate characteristics. Rationale behind the selection of locations is discussed along with their climate characteristics. Climate analysis is preceded by the methodology of analysis of climates. Discussions of optimisation results focus on the effectiveness of the optimisation process in reaching optimum. Convergence to optimum from different starting points is analysed and discussed. It is shown that the use of rules of thumb in constraint based problem formulation offers advantage in design exploration at early stages. Study case involving multi-criteria optimisation using Pareto optimality is shown.

Selection of study locations

Climatic conditions can vary quite significantly at different locations on Earth. There are also daily and seasonal variations in wind direction, rain, solar availability, etc. Buildings are designed to deal with adverse climatic situations and to provide comfortable and usable activity spaces. To determine the effectiveness of the proposed framework, two locations with significant climatic variations have been chosen from the climatic classification suggested by Köppen (1884), later modified by Trewartha (1968). Major climatic regions in Köppen classification are shown in Figure 6.1. Monthly Dry-Bulb and Dew-Point temperatures in Köppen climates in the Northern Hemisphere are shown in adsf Figure 6.2 and Figure 6.3 respectively. The locations chosen for this thesis are Phoenix, USA and Kilkenny, Ireland.

Phoenix, USA is classified as part of type *BWh: arid climate* and characterised by deficient and irregular precipitation. Climate is hot and dry with a mean annual temperature of above 18°C; maximum daytime temperature during the summer can reach above 50°C. On the other hand, Kilkenny, Ireland falls under group C: moist subtropical mid-latitude climates of type *Cfb: marine*. The climate is characterized by low annual temperature in high latitude region, winds from the oceans moderate the climate and summers are quite cool. Places are usually humid with mild winters; average temperature of the coldest month ranges between -3°C to 18°C. Location maps of both locations are given in Figure 6.4 and Figure 6.5.

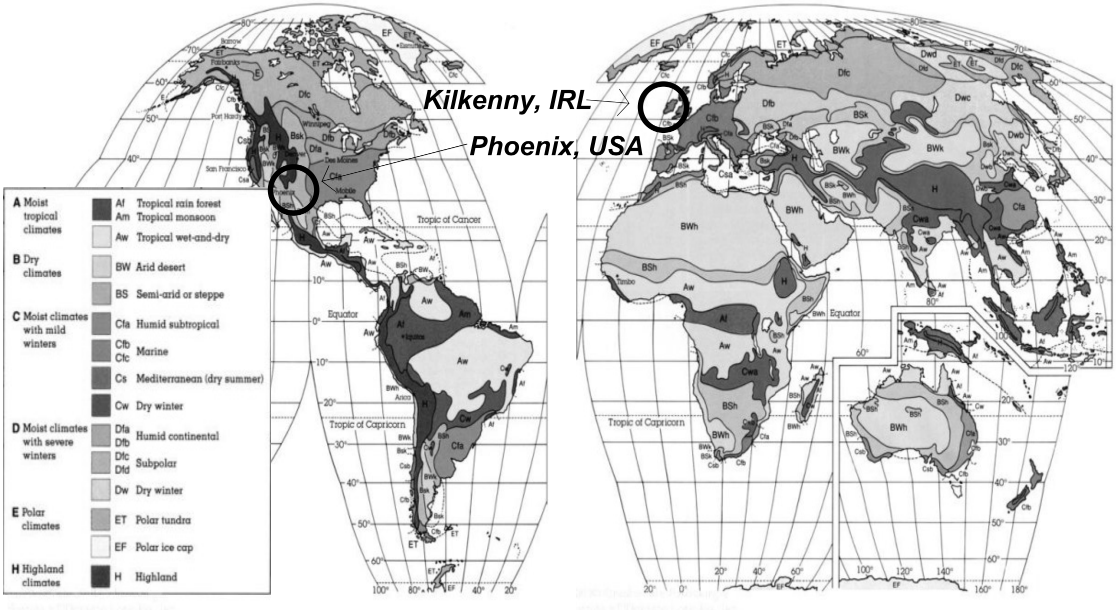


Figure 6.1: Köppen climatic regions, after (Leemans and Cramer 1991).

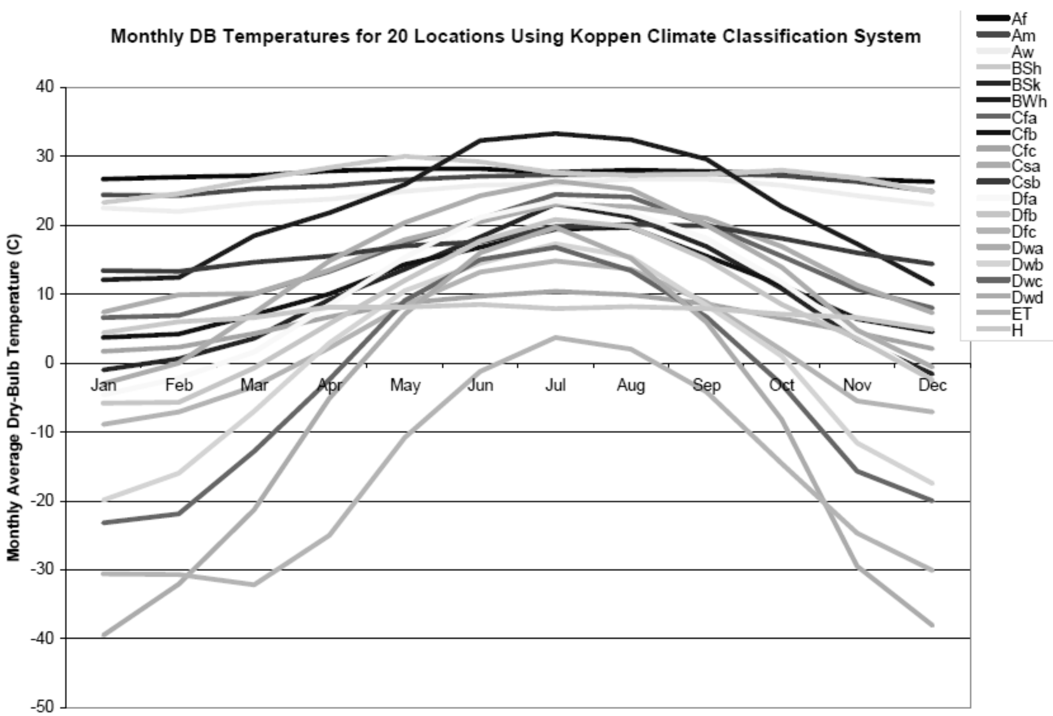


Figure 6.2: Monthly Dry-Bulb temperatures in Köppen climates (Northern Hemisphere), after (DOE 2004a).

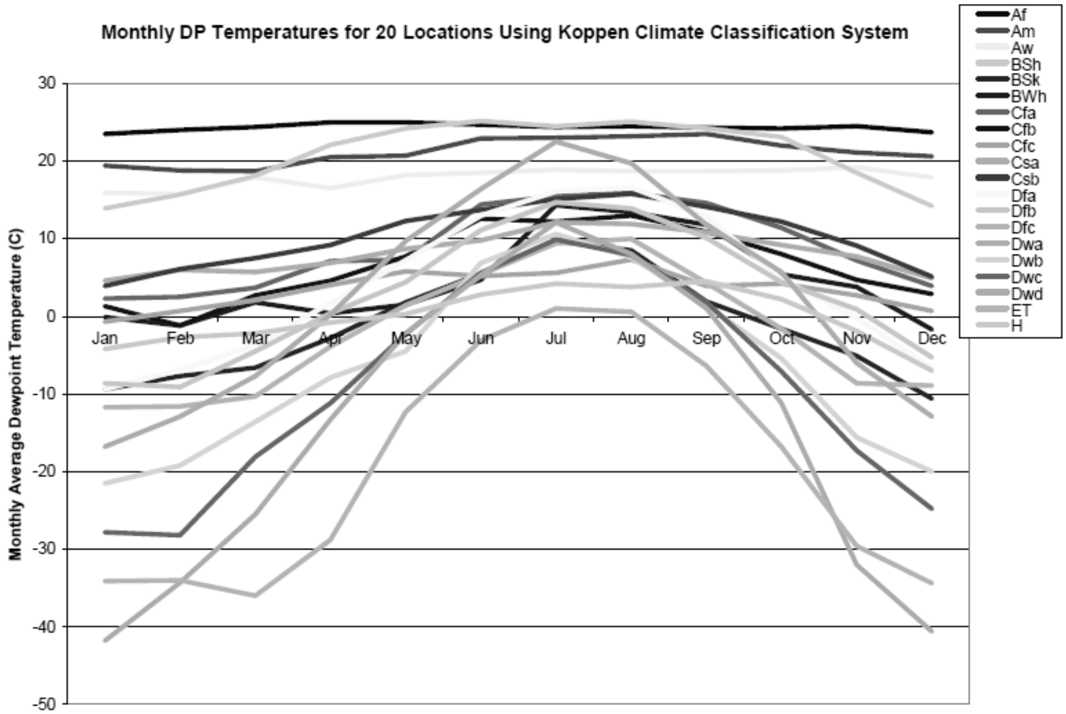


Figure 6.3: Monthly Dew Points in Köppen climates (Northern Hemisphere), after (DOE 2004a).

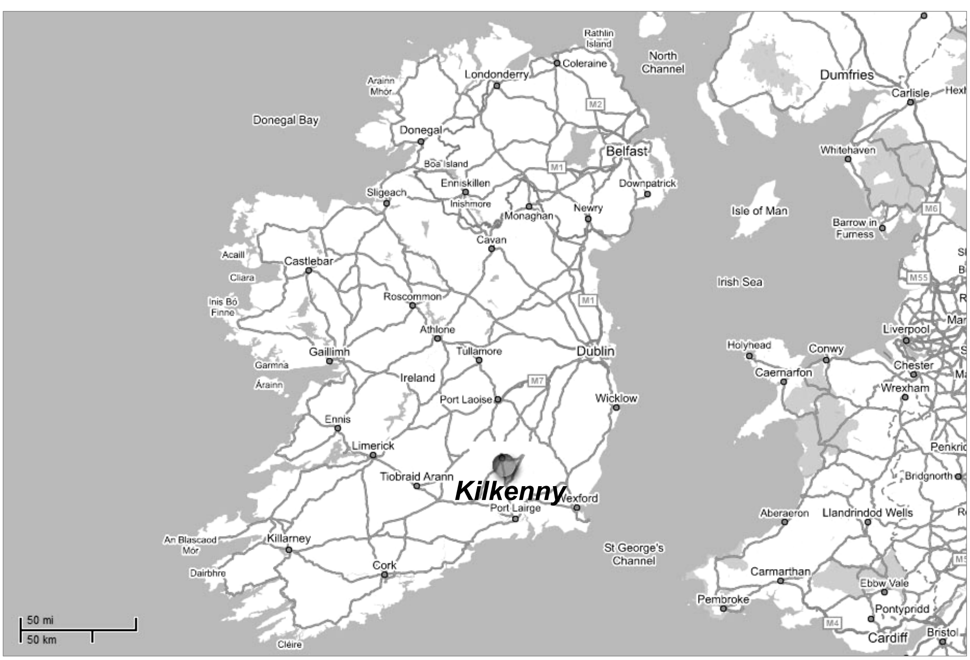


Figure 6.4: Location map of Kilkenny, Ireland. Source: http://maps.google.com/

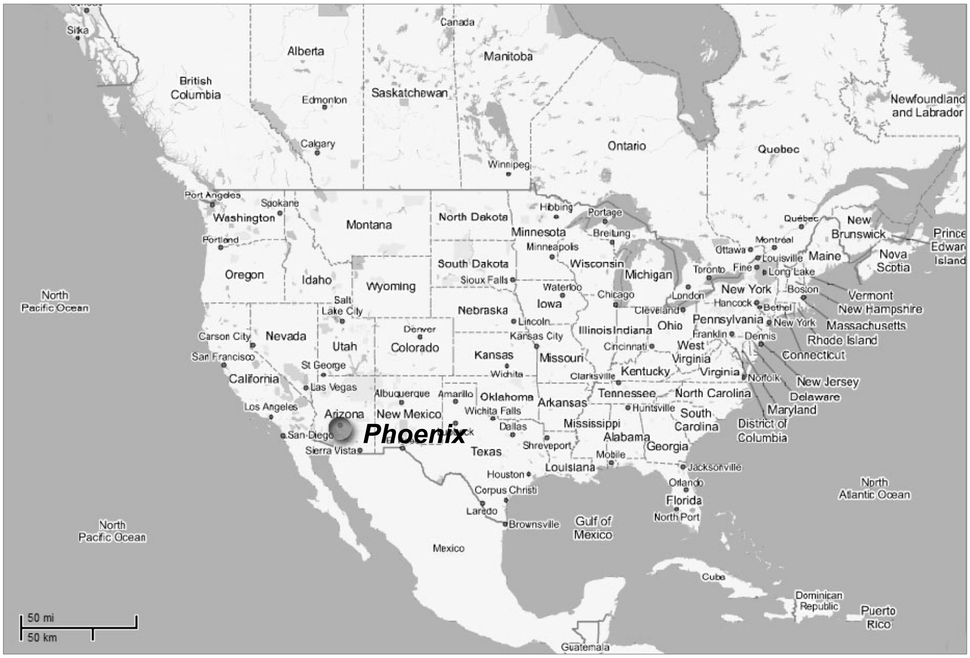


Figure 6.5: Location map of Phoenix, USA. Source: http://maps.google.com/

Climate analysis of locations

The ability to design green buildings depends on the building professionals' skills to identify variations of climatic parameters within a site, develop awareness of possible future modifications produced by introduction of new built form and use this potential during the design process at different scales of application (De Schiller and Evans 1996). The use of climate charts and thermal indices in the selection of bioclimatic design strategies are described in (Zain-Ahmed *et al.* 1998; Ogunsote and Prucnal-Ogunsote 2003; Rabah 2005). Widespread availability of meteorological data for myriad locations and stochastic modelling of climatological databases have made it possible to process climate for practically any location (Robinson 2003).

To interpret the results of optimisation runs, an understanding of the dynamics of climate at the two locations is necessary. Statistically averaged weather data obtained from US Department of Energy[1] are visualised using the application and methods described in (Marsh 2003; Robinson 2003). Hourly data: global and diffuse horizontal solar irradiance (Whm^{-2}), air temperature (°C), relative humidity (%), wind speed (ms^{-1}) and wind direction (° from N) along with the latitude and longitude difference for locations are processed to produce the graphs relating to solar, daylight, synoptic and

[1] Weather data in EPW (EnergyPlus Weather file) format, produced from TMY or ASHRAE IWEC (International Weather for Energy Calculations) datasets are available from: http://www.eere .energy.gov/buildings/energyplus /cfm/weather_data.cfm

ground parameters. Appendix G describes the methodology and analyses the climate characteristics of the chosen locations.

Study 1: one-zone commercial building

A one-zone, single-storied commercial building having an aggregate floor area of 500sqm has been selected for study 1. There are four windows on four cardinal sides of the building expressed as a fraction of the cardinal wall areas. The purpose of this study is to build confidence in the system and test convergence efficiencies of the implemented algorithms for optimisation from different starting points. The building is modelled according the simulation methods described in Chapter 5. Optimisation problem is run for both the locations: Kilkenny, Ireland and Phoenix, USA. The building is let to operate in regular schedules, details of which are given in Appendix B. Other than the fraction glazing on the four sides, building azimuth and building aspect ratio (width/length) are allowed to change. The objective of the study is to minimise total energy consumption while keeping the ADF within a specified value, 5% in this case - modelled as a constraint to the optimisation problem.

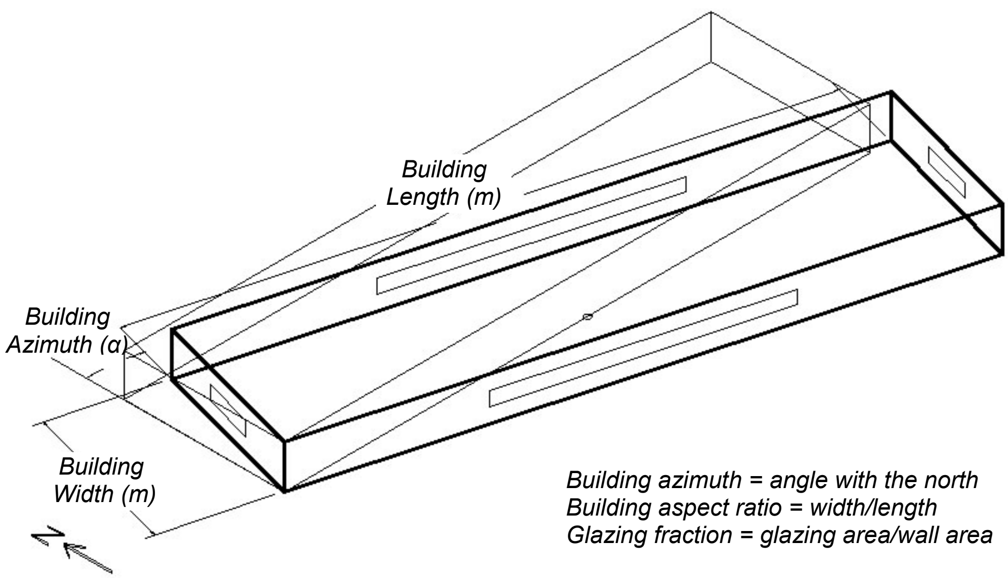


Figure 6.6: One-zone, single-storied building for study 1.

Starting points for six design variables for the convergence study are shown in Figure 6.7 and Table 6.1. Information on design control, design response, design constraints, design objectives and optimiser parameters are given in Table 6.2.

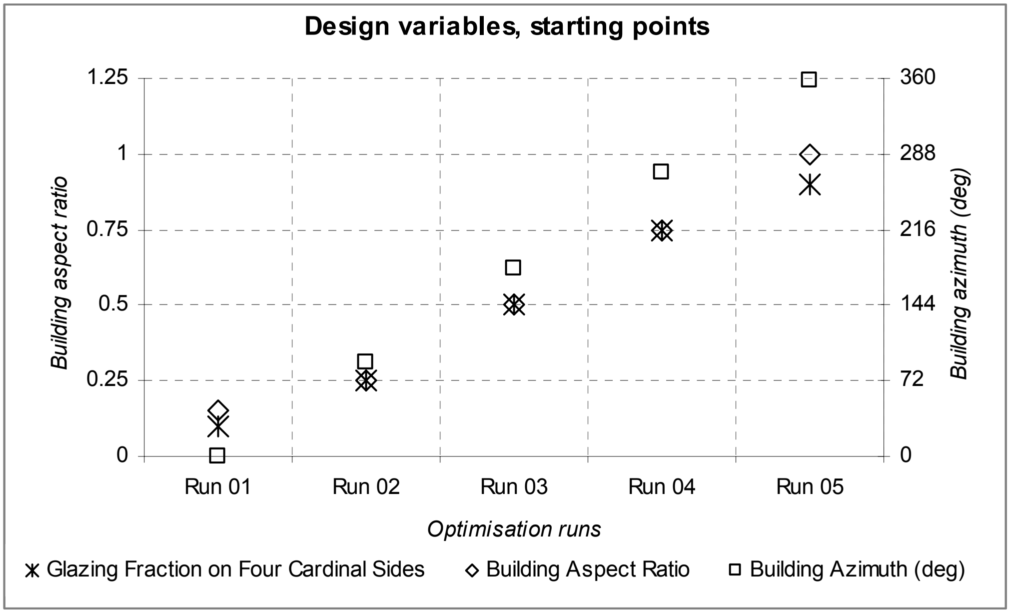


Figure 6.7: Design variables for convergence study, starting points.

Table 6.1: Design variable information and initial responses for study 1.

Variables	***Run 1***	***Run 2***	***Run 3***	***Run 4***	***Run 5***
Building azimuth (deg)	0.0	90.0	180.0	270.0	359.0
Building aspect ratio	0.15	0.25	0.5	0.75	1.0
North glazing fraction	0.1	0.25	0.5	0.75	0.9
South glazing fraction	0.1	0.25	0.5	0.75	0.9
East glazing fraction	0.1	0.25	0.5	0.75	0.9
West glazing fraction	0.1	0.25	0.5	0.75	0.9
Responses (Kilkenny)					
Energy Consumption (kWh)	49431	51981	53994	59354	62225
Average Daylight Factor	1.32	2.9	5.1	7.35	8.75
Responses (Phoenix)					
Energy Consumption (kWh)	93224	106538	108334	123679	127009
Average Daylight Factor	1.32	2.9	5.1	7.36	8.76

Table 6.2: Parameters for implemented optimisation algorithm.

Design control information			
Optimisation method	SQP (Sequential Quadratic Programming)		
Objective	Minimise		
Constraint tolerance	-0.03		
Violated constraint tolerance	0.003		
Gradients calculated by	First forward difference		
Relative finite difference step	0.001		
Minimum finite difference step	0.0001		
Optimiser parameters			
Relative hard convergence criteria	0.001		
Consecutive iteration for convergence	2		
Maximum number of iteration	100		
Design response information			
	Objective	***Constraint***	
Total Energy Consumption	Yes	No	
Daylight Factor	No	Yes	
Design constraints information			
	Bound	***Limit***	***Scale Factor***
Daylight Factor	Lower	Variable	1.0
Design objectives information			
	Worst value	***Weight factor***	***Target***
Total Energy Consumption	Undefined	1.0	Minimise

Results for Kilkenny

Results from optimisation runs for Kilkenny are given in tabular format in Table 6.3. Values of objective function in best design points having different starting values of variables vary between 52440 and 53337 kWh with a variation of 1.67% (see Figure 6.8). Indicative savings in energy consumption is highest in run 5, a saving of 14.7%, partly due to the fact that ADF for starting point is 8.75%, above the specified constraint value of 5%. Results for run 1 and 2 can be explained with the lower ADF value for starting points which is less (1.32% and 2.9% respectively) than the specified constraint value for ADF in this optimisation problem.

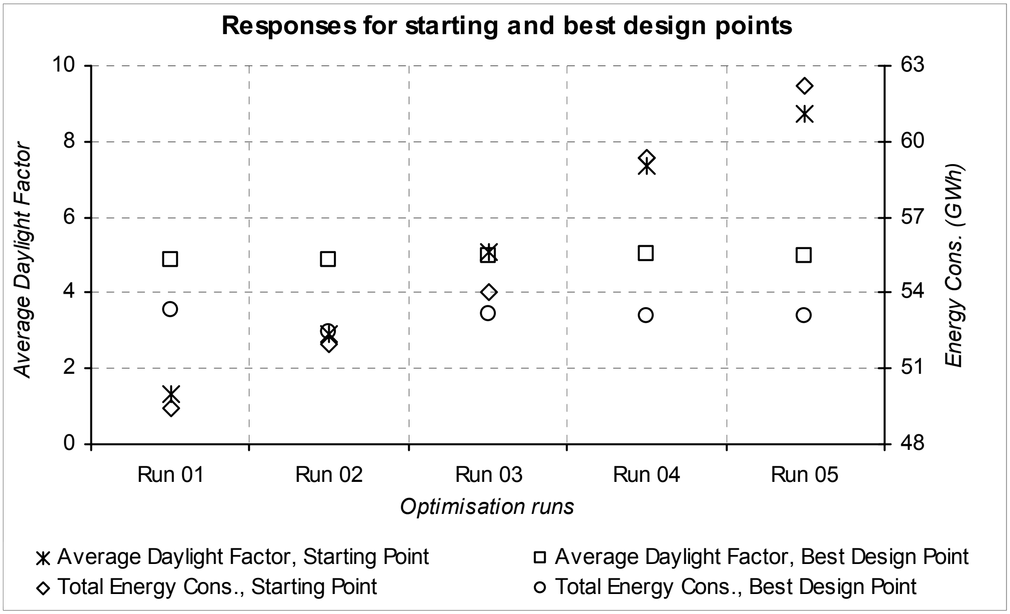


Figure 6.8: Responses for starting and best design points for study1, Kilkenny.

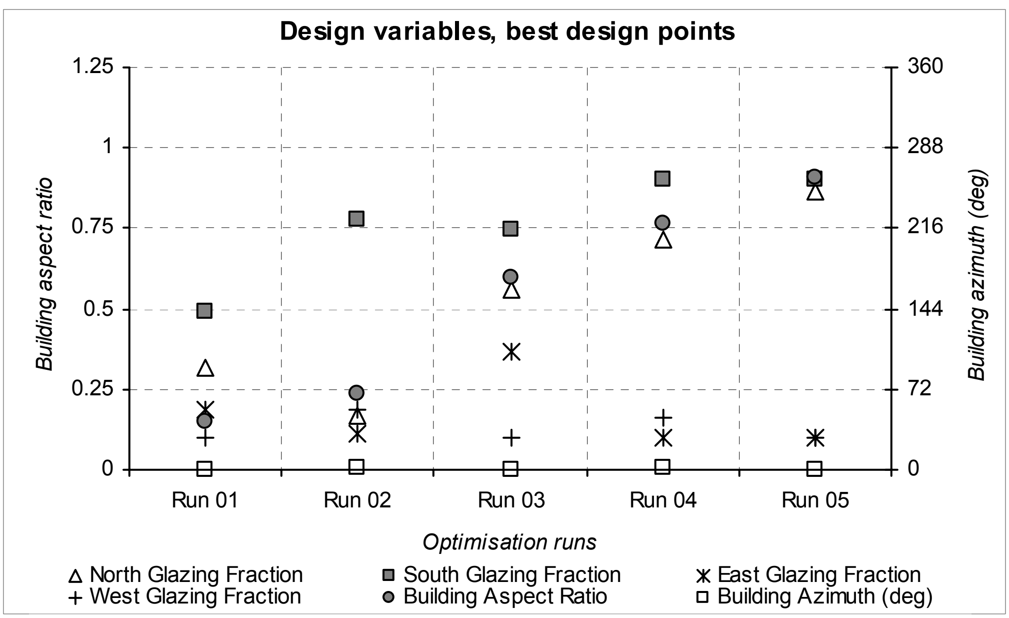


Figure 6.9: Design variables, best design points for study1, Kilkenny.

Design variables, if translated into building parameters, reveal strong climatic bias in converging to optimum. Glazing fraction in west and east tend to be closer to the lower bounds for the variables. Glazing on the cardinal south tends to be the maximum of all four glazing variables, which is representative of the climate of Kilkenny as analysed in Appendix G. Solar gain is highest in south during regular office hours and contributes to the heating gain, thus resulting in reduced energy consumption. Building azimuth varies between 0° to 1° from the North conforming to the directional advantage on building orientation in the Northern Hemisphere. Building aspect ratio, i.e. form of the building tend to be closer to the initial value; i.e. starting points.

Table 6.3: Optimisation results, best design points for study1, Kilkenny.

Variables	***Run 1***	***Run 2***	***Run 3***	***Run 4***	***Run 5***
Building azimuth (deg)	0	0.97	0.52	1.0	0.0
Building aspect ratio	0.15	0.23	0.6	0.76	0.91
North glazing fraction	0.32	0.17	0.56	0.71	0.86
South glazing fraction	0.49	0.78	0.75	0.9	0.9
East glazing fraction	0.19	0.1	0.37	0.1	0.1
West glazing fraction	0.1	0.19	0.1	0.16	0.1
Responses					
Energy Consumption (kWh)	53337	52440	53119	53075	53048
Average Daylight Factor (%)	4.89	4.86	4.96	5.03	4.96
Building parameters					
Length of building (m)	57.74	46.3	28.97	25.6	23.45
Width of building (m)	8.66	10.8	17.26	19.53	21.32
North glazing area (m^2)	49.75	20.94	43.92	49.23	54.72
South glazing area (m^2)	76.35	97.52	58.34	62.13	56.99
East glazing area (m^2)	4.33	3.2	17.14	5.27	5.76
West glazing area (m^2)	2.34	5.47	4.66	8.55	5.76
Total glazing area (m^2)	132.78	127.13	124.06	125.18	123.22
Total vertical surface area (m^2)	358.53	308.34	249.63	243.7	241.77

Design variables for best design points are shown in Figure 6.9. Glazing fractions for four cardinal sides are translated into building parameters in Figure 6.10 and compared with corresponding building aspect ratios. Figure 6.12 compares total glazing area with total vertical surface area. Total glazing area for best design points in five optimisation runs vary between 123.22 and 132.78 m^2; referring to the need for certain sizes of glazing to ensure the specified ADF, modelled as constraint in this optimisation problem. The use of ADF, a rule of thumb for determining daylight availability in interior spaces can be said as having significant influence in convergence. Resulting building plans and 3D models from optimisation runs are given in Figure 6.11 and Figure 6.13 respectively. Total *analysis calls, gradient requests and the total number of points in the design space are given in*

Table 6.4.

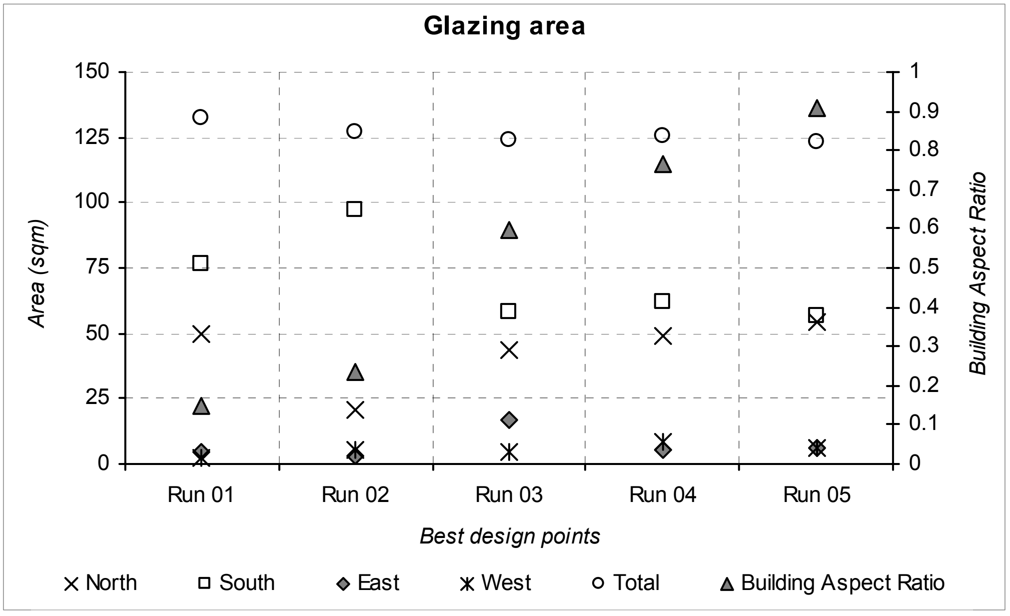


Figure 6.10: Glazing area vs. building aspect ratio for study1, Kilkenny.

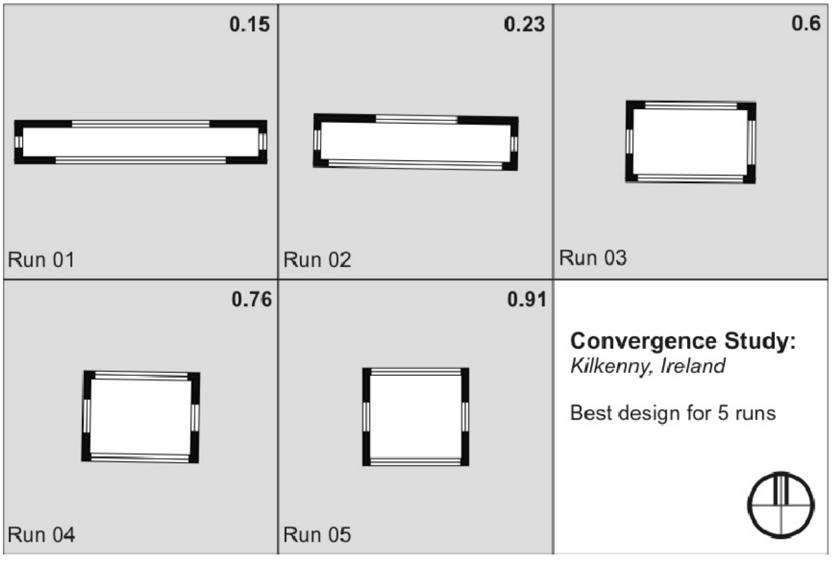


Figure 6.11: Building plans for study 1 results for Kilkenny, Ireland.

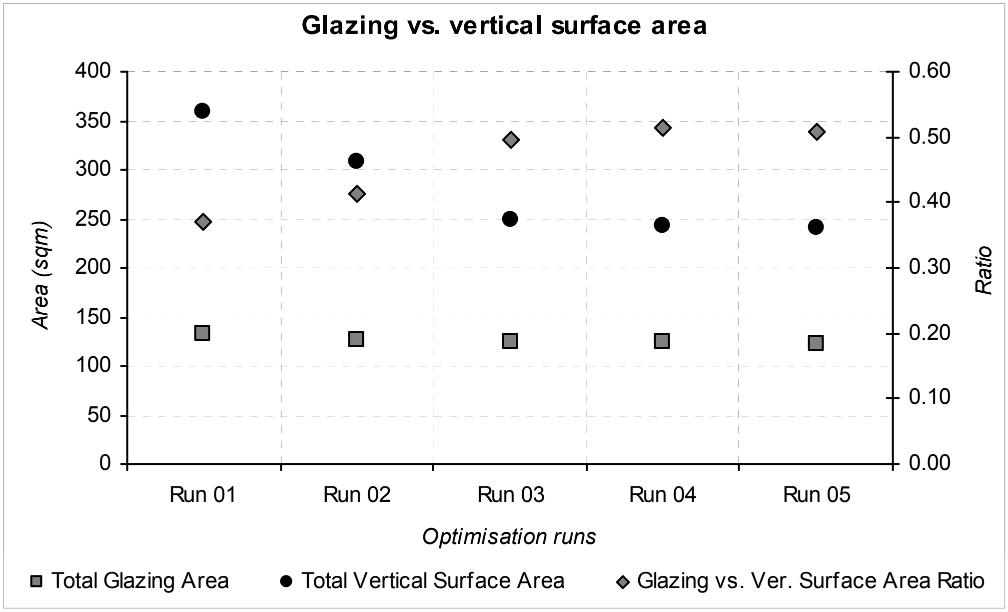


Figure 6.12: Glazing vs. vertical surface area for study1, Kilkenny.

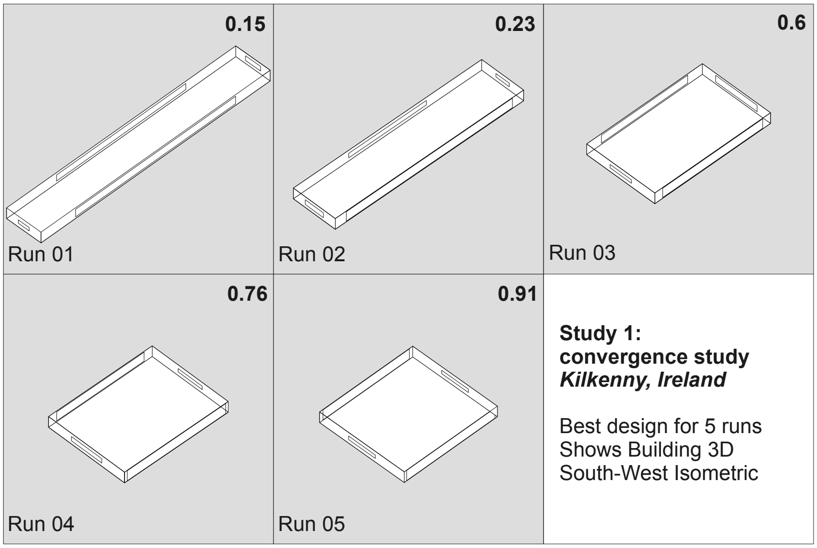


Figure 6.13: Building 3D models for study 1 results for Kilkenny, Ireland.

Table 6.4: Simulation calls and design space parameters for study1, Kilkenny.

	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**
Total analysis calls	34	118	64	36	36
Total gradient requests	4	13	7	4	4
Total no. of points in the design space	12	43	26	17	16

Results for Phoenix

Results from optimisation runs for Phoenix are given in tabular format in Table 6.5. Values of objective function for different starting points varied between 104072 and 106326 kWh with a variation of 2.11% (see Figure 6.14). Indicative savings in energy consumption is highest in run 5 with a saving of 19.54%, similar to that of results for Kilkenny; again due to the fact that ADF for starting point is 8.76%, above the specified constraint value of 5%. Results for run 1 and 2 can be explained with the lower ADF value for starting points which is less (1.32% and 2.90% respectively) than the specified constraint value for ADF in this optimisation problem.

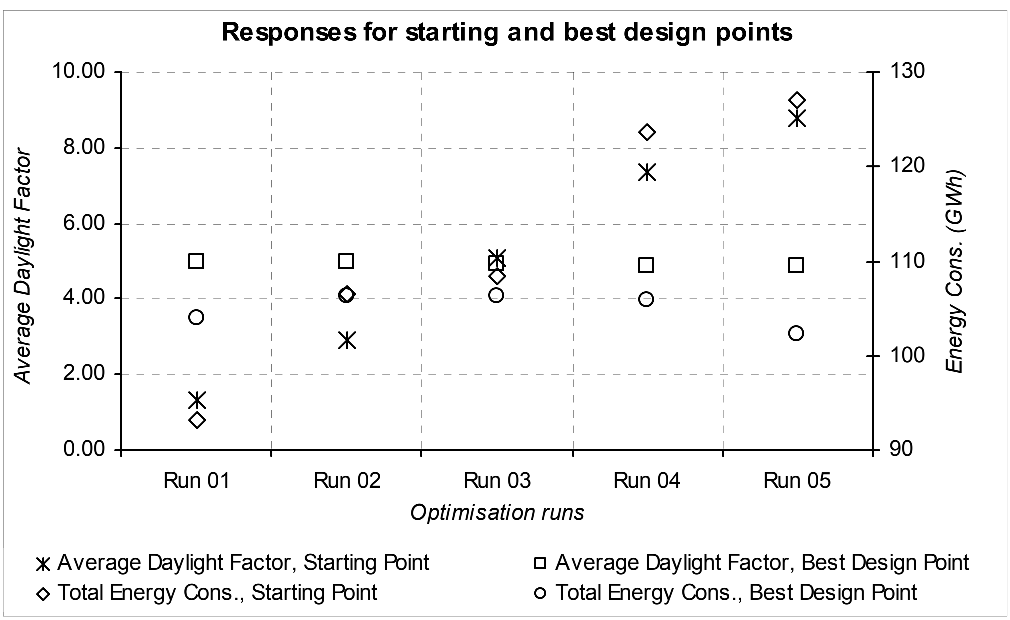


Figure 6.14: Responses for starting and best design points for study1, Phoenix.

Like Kilkenny, design variables correspond strongly with the climate conditions of the site as analysed in Appendix G. North glazing area tends to be greater than that of the other cardinal sides, in line with the cooling dominated nature of Phoenix climate. South glazing area seems to be the second choice of variable for the optimiser to bring daylighting in the interior space to the adequate level. Being a cooling dominated climate, southern façade needs to be designed with adequate shading to reduce heat gains for most parts of the year. More pronounced climate impact in the articulation of vertical surfaces can be seen in west façade where values for west glazing fraction and corresponding glazing area tended to be closer to the lower bounds. Design variables, when translated into building parameters in Figure 6.16 reveal strong bias for increased glazing in north and south façades. Building azimuth of 4.45° and 15° from the north in run 3 and run 4 respectively corresponds with some increase in the glazing area in east and west façades. This essentially minimises the effect of solar heat gain during daytime in Phoenix, which can be explained with the help of solar availability contours in Appendix G. Solar availability is highest on south façade and less pronounced in other cardinal façades.

Total glazing area for best design points in five optimisation runs vary between 120.66 to 135.86 m^2. The variations are more pronounced than that of Kilkenny, simply because of the greater differences in solar availability between different cardinal façades.

Table 6.5: Optimisation results, best design points for study1, Phoenix.

Variables	***Run 1***	***Run 2***	***Run 3***	***Run 4***	***Run 5***
Building azimuth (deg)	0.0	2.77	4.45	14.94	0.0
Building aspect ratio	0.15	0.15	0.7	1.0	0.27
North glazing fraction	0.72	0.33	0.66	0.72	0.9
South glazing fraction	0.12	0.5	0.57	0.9	0.13
East glazing fraction	0.1	0.1	0.34	0.27	0.1
West glazing fraction	0.1	0.1	0.34	0.1	0.1
Responses					
Energy Consumption (kWh)	104072	106227	106326	105819	102194
Average Daylight Factor (%)	5.0	5.0	4.93	4.86	4.86
Building parameters					
Length of building (m)	57.74	57.74	26.66	22.36	43.17
Width of building (m)	8.66	8.66	18.76	22.36	11.58
North glazing area (m^2)	111.7	52.12	47.71	43.57	104.89
South glazing area (m^2)	19.38	78.93	40.94	54.26	14.88
East glazing area (m^2)	2.39	2.34	17.07	16.79	3.13
West glazing area (m^2)	2.39	2.34	17.07	6.04	3.13
Total glazing area (m^2)	135.86	135.73	122.8	120.66	126.03
Total vertical surface area (m^2)	358.53	358.53	245.24	241.5	295.64

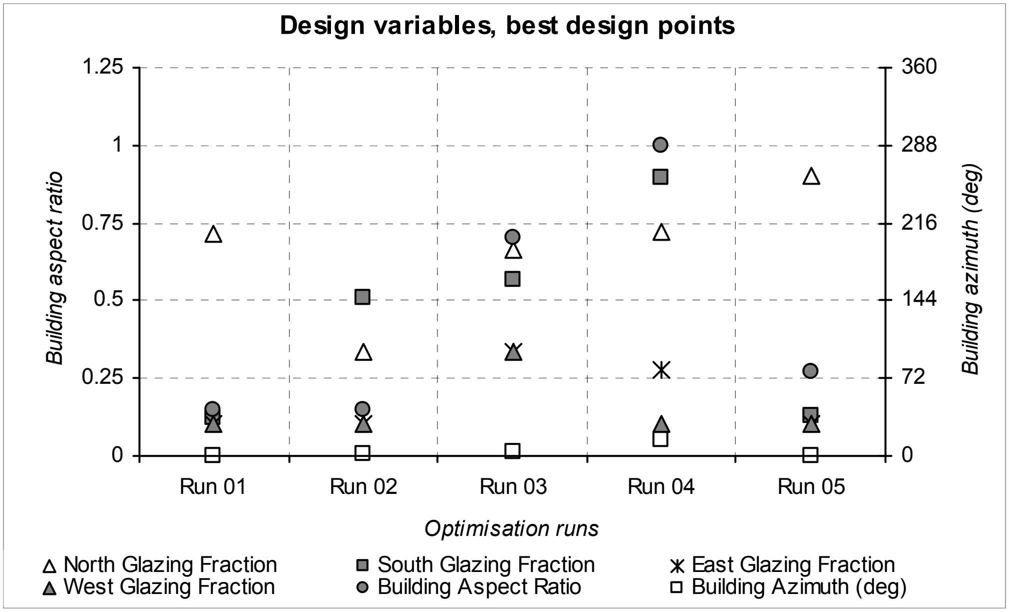


Figure 6.15: Design variables, best design points for study1, Phoenix.

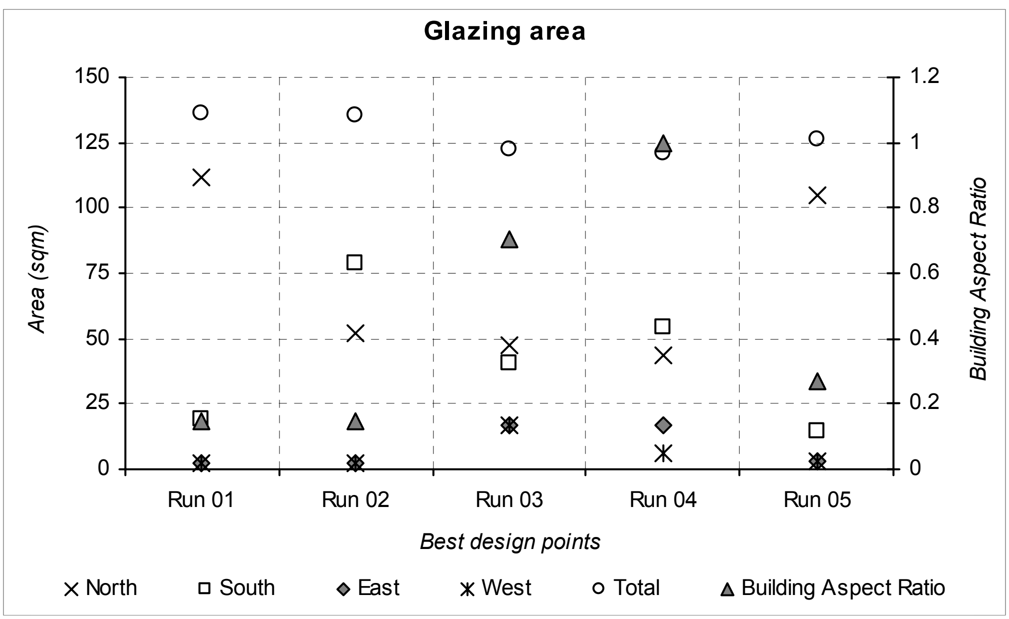


Figure 6.16: Glazing area vs. building aspect ratio for study1, Phoenix.

Total analysis calls, gradient requests and the total number of points in the design space are given in

Table 6.6. Resulting building plans and 3D models from optimisation runs are given in Figure 6.18 and Figure 6.19 respectively.

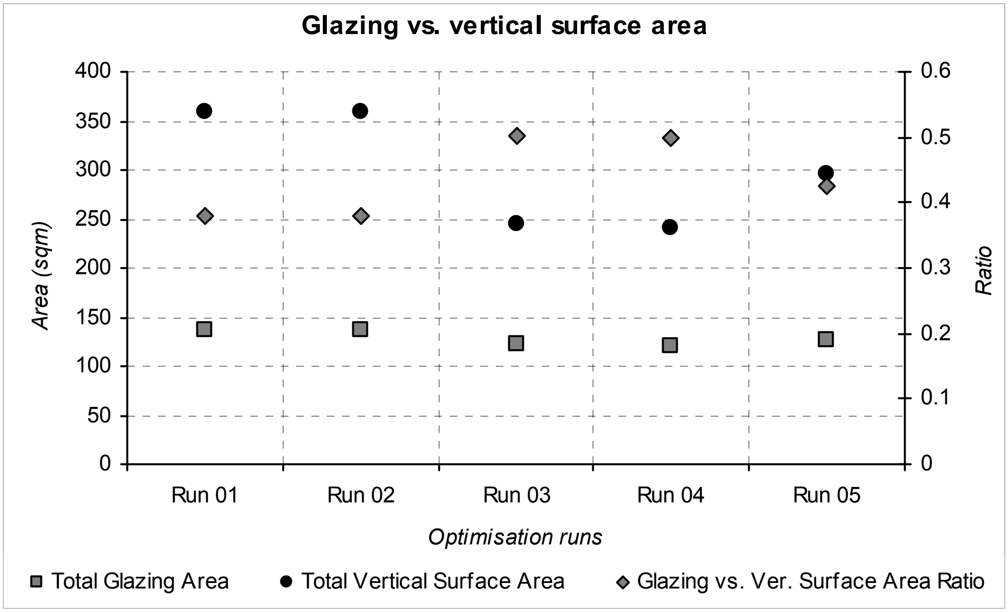


Figure 6.17: Glazing vs. vertical surface area on cardinal sides for study1, Phoenix.

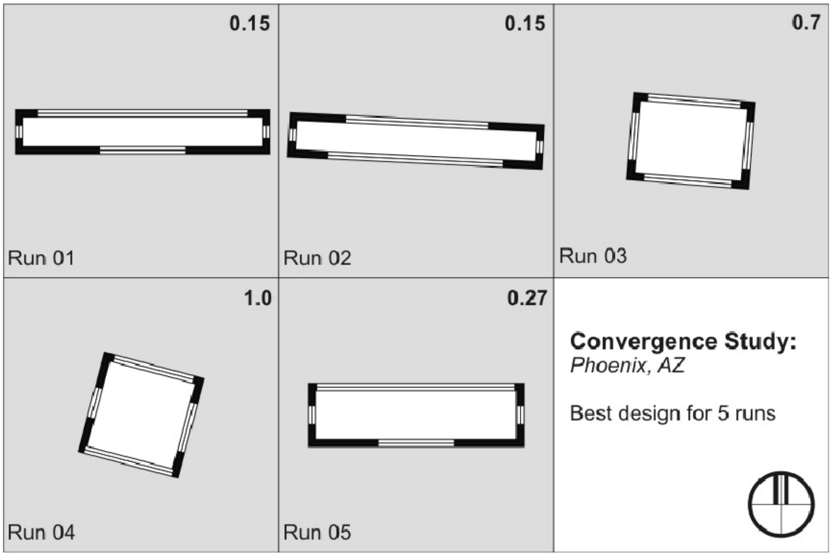


Figure 6.18: Building plans for study 1 results for Phoenix, USA.

Table 6.6: Simulation calls and design space parameters for Phoenix.

	Run 1	***Run 2***	***Run 3***	***Run 4***	***Run 5***
Total analysis calls	45	96	37	104	55
Total gradient requests	5	11	4	12	6
Total no. of points in the design space	21	39	16	35	25

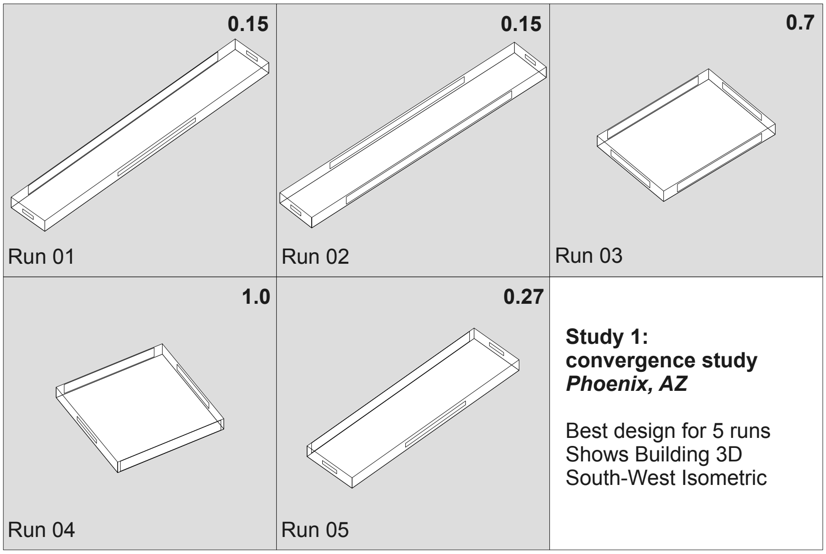


Figure 6.19: Building 3D models for study 1 results for Phoenix, USA.

Convergence efficiency

Optimisation runs in two locations with distinct climatic features show the effectiveness of the algorithm in converging to an acceptable conceptual design solution. From the discussion of results in the preceding sections it is clear that a number of solutions having similar values for objective function exist for *energy consumption minimisation* problem for both Kilkenny and Phoenix; i.e. different values for building aspect ratio, building azimuth and glazing fractions. It can be generalised that conceptual designs aimed at energy efficiency through optimisation of building form and orientation does not restrict the creative aspects of architectural design. Building parameters for subsequent optimisation runs from various starting points, although seems to lack continuity which has also been reported in (Wetter and Wright 2004; Wetter and Polak 2005), can offer significant advantage in exploring design solutions; in particular at concept development stages. Like the vast majority of real engineering problems where optimisation is applied, architectural design of buildings concerns itself more with the exploration of design space than mere minimisation of design variables. Study 1 suggests that deployment of mathematical optimisation in conceptual architectural design opens up opportunities for multi-criteria design exploration.

Study 2: Multi-criteria decision making

To incorporate multi-criteria design exploration at the centre of decision making process, this study has been designed with two interdependent objective functions. Annual cumulative energy consumption and a predefined ADF have been selected as objectives for the optimisation problem. An increase in the ADF results in greater availability of daylight; sometimes offering significant energy savings by offsetting a portion of the electric lighting load. This may also increase cooling or heating loads because of heat gain or loss through increased glazing in façades, respectively. Reduction in internal gains can result in lower cooling demands in hot climates but higher heating demands in cold climates which may be compensated by solar gains during daytime. In addition to energy savings, daylighting generally improves occupant satisfaction and comfort (LANL 2002), which may be one of the design objectives provided that the cumulative energy consumption remains at an acceptable level. This experiment involves the study of two objectives and their interaction in environmental design of buildings. Based on the modelling techniques described in Chapter 5, the one-zone, single-storied commercial building from study 1 has been tested here. Values of design variables for starting points are given in Table 6.7. Values for objective ADF varies from 2% to 8% in 7 different runs. Simulation is performed based on two separate schedules: regular and extended, details of which are given in Appendix. The reasons behind running the building with two schedules are to see the impact of the absence of daylight during winter nights on building form and orientation. Two locations: Kilkenny, Ireland and Phoenix, USA have been chosen as locations. Parameters for implemented optimisation algorithm are given in Table 6.2.

Table 6.7: Values of design variables for starting points, study 2.

Variables	***Starting point value***
Building azimuth (deg)	0.0
Building aspect ratio	0.15
North glazing fraction	0.5
South glazing fraction	0.5
East glazing fraction	0.5
West glazing fraction	0.5

For evaluation of optimisation results, the concept of Pareto optimality (Petrie *et al.* 1995) is employed. The use of Pareto optimality in design can be found in (Jo and Gero 1998; O'Sullivan 1999; Wright and Loosemore 2001; Mourshed *et al.* 2003b). Pareto optimal solutions, which are often called nondominated or non-inferior (Radford and Gero 1988), allows the efficient evaluation of multi-criteria solutions (Jo and Gero 1998). In the multiobjective optimisation, the objective space is of interest to the designers. Because of the contradiction and possible incommensurability of the objective functions, a single solution can not be found satisfying optimal criteria for all the objectives simultaneously. Objective vectors, where none of the components can be improved without deterioration to at least one of the other components (Miettinen 1999). A Pareto optimal set usually has a lot of Pareto optimal solutions which could be nonconvex and nonconnected. It is the decision maker's task to select the optimum value based on the decision matrix or decision plot.

Results for Kilkenny, regular schedule

Results from study 2 for the building in Kilkenny, operating in regular schedule, are given in Table 6.8. Values of objective function (Total Energy Consumption) increases with the corresponding increase in the constraint value (Average Daylight Factor). Design variables for best design points are given in Figure 6.20 which shows relatively benign characteristics of Kilkenny climate, also been observed in study 1.

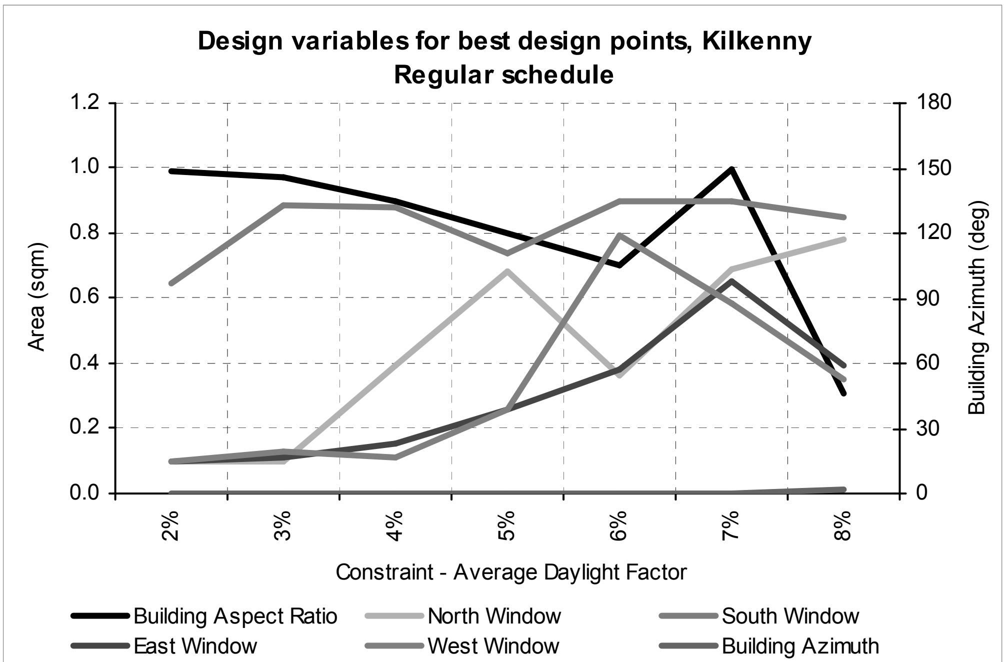


Figure 6.20: Design variables for best design points for study 2, Kilkenny, regular schedule.

Relatively larger value of south glazing fraction and corresponding area of glazing reinforces conventional wisdom of bioclimatic design in Marine climates such as Kilkenny. Relatively larger glazing fractions on east and west walls particularly at 8% ADF can be explained with Figure 6.21, where design variables are interpreted to corresponding glazing areas. As building aspect ratio became smaller, the corresponding areas for east and west glazing are considerably smaller than that of north and south. The dominance of south glazing over other cardinal sides is visible from the resultant graphs. Pareto curve for Kilkenny is given in Figure 6.22 which combines total energy consumption and corresponding ADF in one graph for decision making.

Table 6.8: Optimisation results, best design points for Kilkenny with regular schedule.

	Average Daylight Factor						
Variables	***2%***	***3%***	***4%***	***5%***	***6%***	***7%***	***8%***
Building azimuth (º)	0.00	0.00	0.00	0.00	0.00	0.00	1.62
Building aspect ratio	0.99	0.97	0.90	0.80	0.70	1.00	0.31
North glazing fraction	0.10	0.10	0.40	0.68	0.36	0.69	0.78
South glazing fraction	0.64	0.88	0.88	0.74	0.90	0.90	0.85
East glazing fraction	0.10	0.11	0.15	0.26	0.38	0.65	0.39
West glazing fraction	0.10	0.13	0.11	0.26	0.80	0.58	0.35
Responses							
Energy Consumption (MWh)	49	50	51	53	56	58	59
Average Daylight Factor (%)	2.30	3.01	3.88	4.99	6.03	6.88	7.89
Building parameters							
Length of building (m)	22.49	22.71	23.58	25.00	26.67	22.36	40.43
Width of building (m)	22.24	22.02	21.20	20.00	18.75	22.36	12.37
North glazing area (m^2)	6.07	6.13	25.20	46.00	26.16	41.67	85.30
South glazing area (m^2)	39.12	54.16	55.98	49.95	64.75	54.34	92.54
East glazing area (m^2)	6.00	6.71	8.81	14.04	19.16	39.53	13.11
West glazing area (m^2)	6.00	7.66	6.40	14.04	40.25	35.27	11.77
Total glazing area (m^2)	57.20	74.66	96.38	124.03	150.32	170.82	202.73
Total wall area (m^2)	241.50	241.52	241.84	243.00	245.26	241.50	285.09

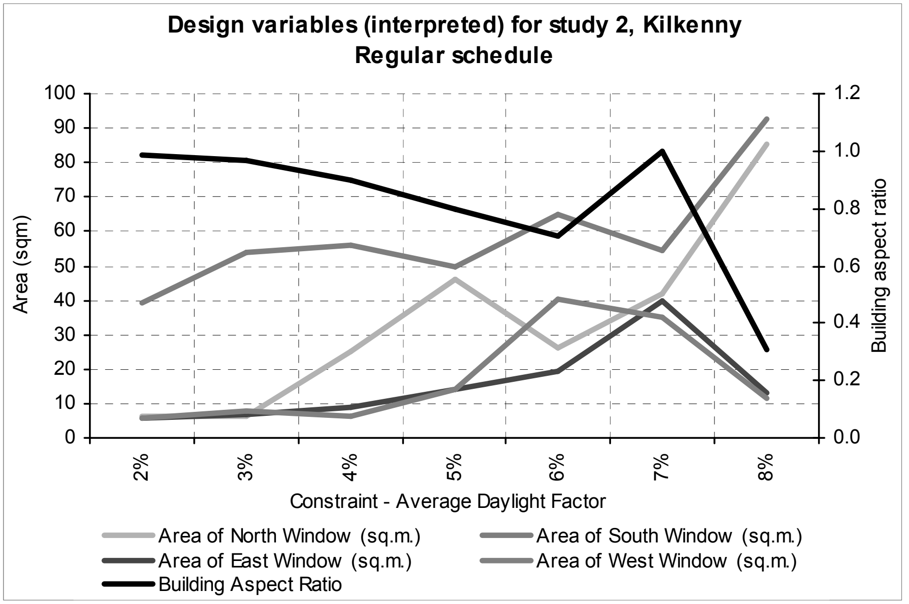


Figure 6.21: Interpreted design variables for best design points, study 2, Kilkenny, regular schedule.

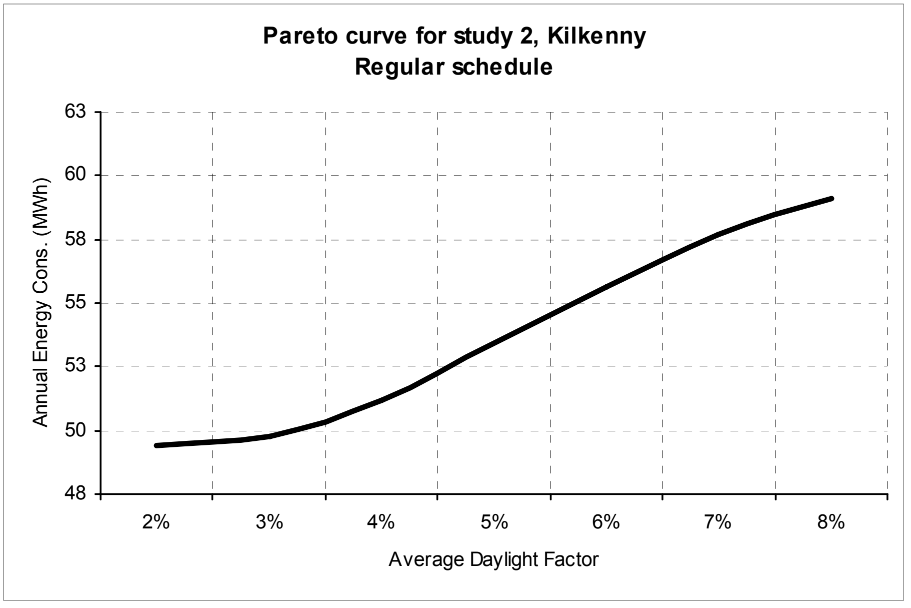


Figure 6.22: Pareto curve for study 2, Kilkenny, regular schedule.

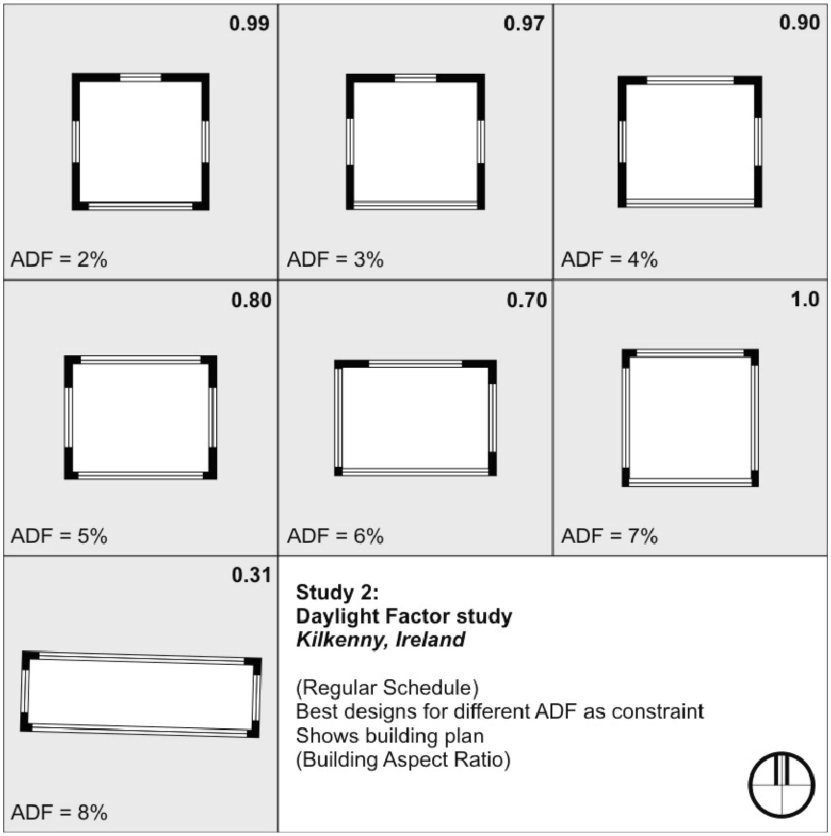


Figure 6.23: Building plans of optimisation results for Study 2, Kilkenny: regular schedule.

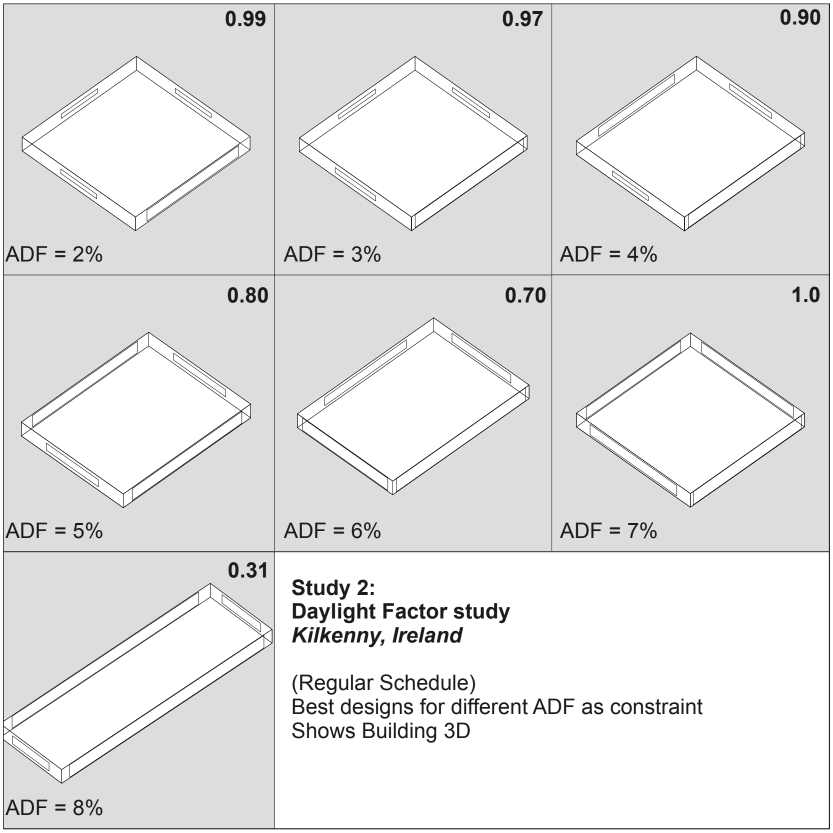


Figure 6.24: Building 3D models of optimisation results for Study 2, Kilkenny: regular schedule.

Results for Kilkenny, extended schedule

Results for study 2 building in Kilkenny with extended operation in Table 6.9 show pronounced difference than that of regular occupancy Table 6.8. Glazing on east and west and to some extent north sides are lower than that of south sides and continuous except for the ADF value of 7%. Figure 6.25 shows the design variables at best design points for the test building with extended schedule, where as Figure 6.26 shows the building parameters as interpreted design variables. Pareto curve for Kilkenny is given in Figure 6.27.

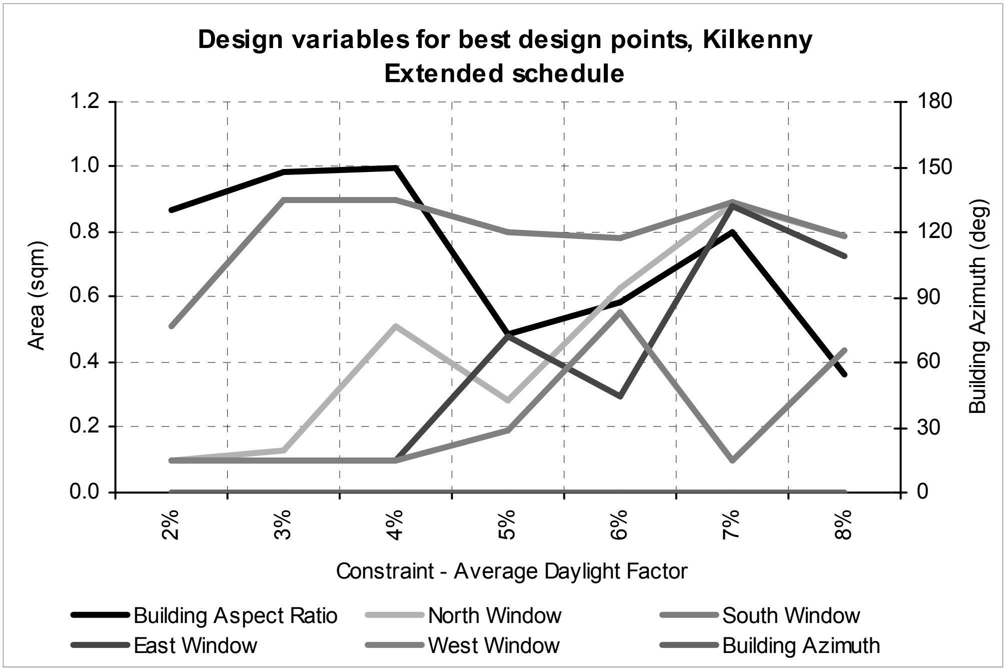


Figure 6.25: Design variables at best design points for study 2, Kilkenny, extended schedule.

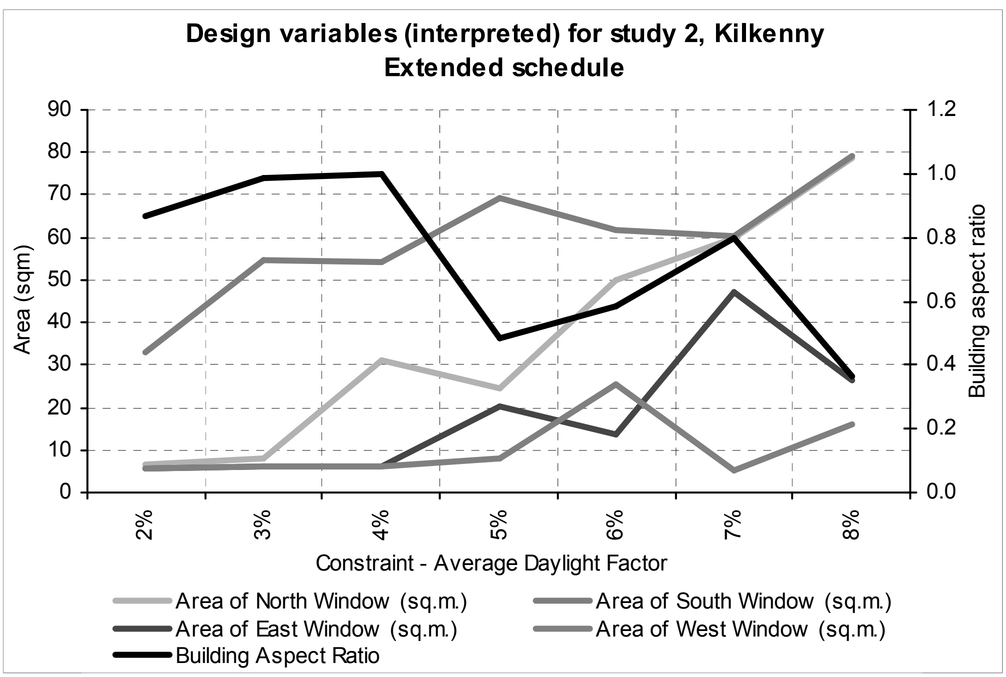


Figure 6.26: Interpreted design variables at best design points for study 2, Kilkenny, extended schedule.

Table 6.9: Optimisation results, best design points for Kilkenny with extended schedule.

	Average Daylight Factor						
Variables	***2%***	***3%***	***4%***	***5%***	***6%***	***7%***	***8%***
Building azimuth (º)	0.00	0.00	0.00	0.00	0.00	0.03	0.00
Building aspect ratio	0.87	0.98	1.00	0.49	0.58	0.80	0.36
North glazing fraction	0.10	0.13	0.51	0.29	0.63	0.89	0.79
South glazing fraction	0.51	0.90	0.90	0.80	0.78	0.89	0.79
East glazing fraction	0.10	0.10	0.10	0.48	0.30	0.88	0.73
West glazing fraction	0.10	0.10	0.10	0.19	0.56	0.10	0.43
Responses							
Energy Consumption (MWh)	50	51	52	54	57	59	61
Average Daylight Factor (%)	2.05	3.01	3.92	4.87	6.03	6.98	7.86
Building parameters							
Length of building (m)	23.98	22.54	22.36	32.10	29.31	25.01	37.08
Width of building (m)	20.85	22.18	22.36	15.57	17.06	19.99	13.48
North glazing area (m^2)	6.47	8.05	30.91	24.71	49.78	59.99	78.84
South glazing area (m^2)	33.13	54.72	54.28	69.44	61.86	60.29	79.13
East glazing area (m^2)	5.63	5.99	6.04	20.18	13.65	47.36	26.46
West glazing area (m^2)	5.63	5.99	6.04	8.05	25.63	5.40	15.81
Total glazing area (m^2)	50.86	74.74	97.27	122.37	150.92	173.04	200.24
Total wall area (m^2)	242.08	241.50	241.50	257.46	250.40	243.01	273.06

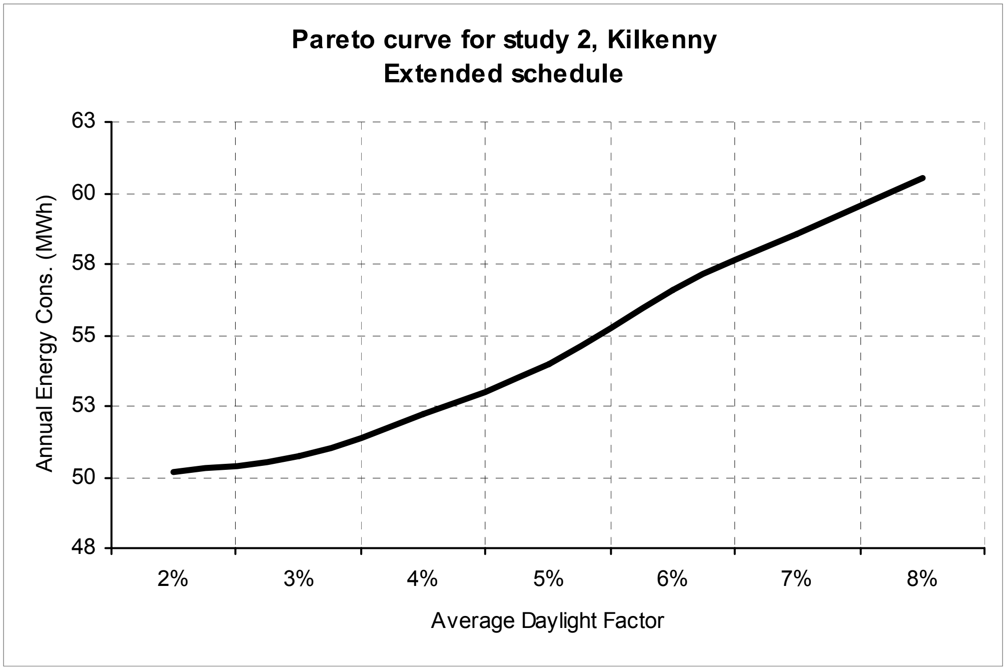


Figure 6.27: Pareto curve for study 2, Kilkenny, extended schedule.

Results for Phoenix, regular schedule

Results from Phoenix in Table 6.10 show a tendency towards maximisation of north glazing as the most feasible strategy for increasing glazing percentage. The reduction of solar heat gain seems to be the most efficient of bioclimatic strategies for Phoenix which is reflected in the values of design variables at best design points in Figure 6.28 and in the interpreted design variables in Figure 6.29. An increase in glazing area in the east and west can be interpreted as the last option to satisfy ADF constraints.

Climatic parameters in Phoenix are more pronounced, hence the gradual increase in the values of design variables. Pareto curve for Phoenix with regular schedule is shown in Figure 6.30. Building plans and 3D models of optimisation results for study 2 for Phoenix is shown in Figure 6.31 and Figure 6.32 respectively.

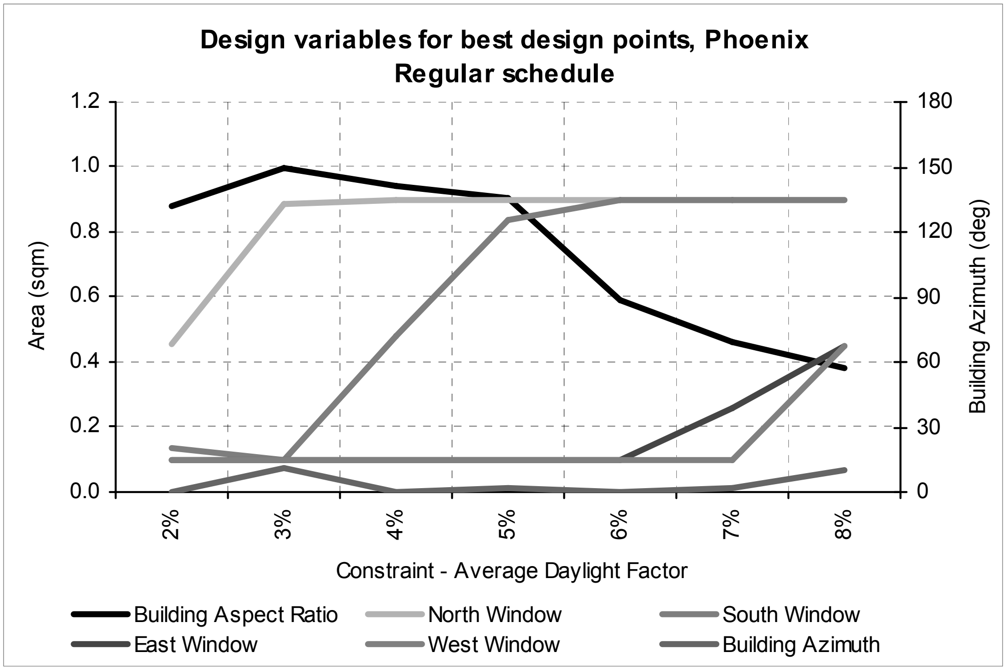


Figure 6.28: Design variables at best design points for study 2, Phoenix, regular schedule.

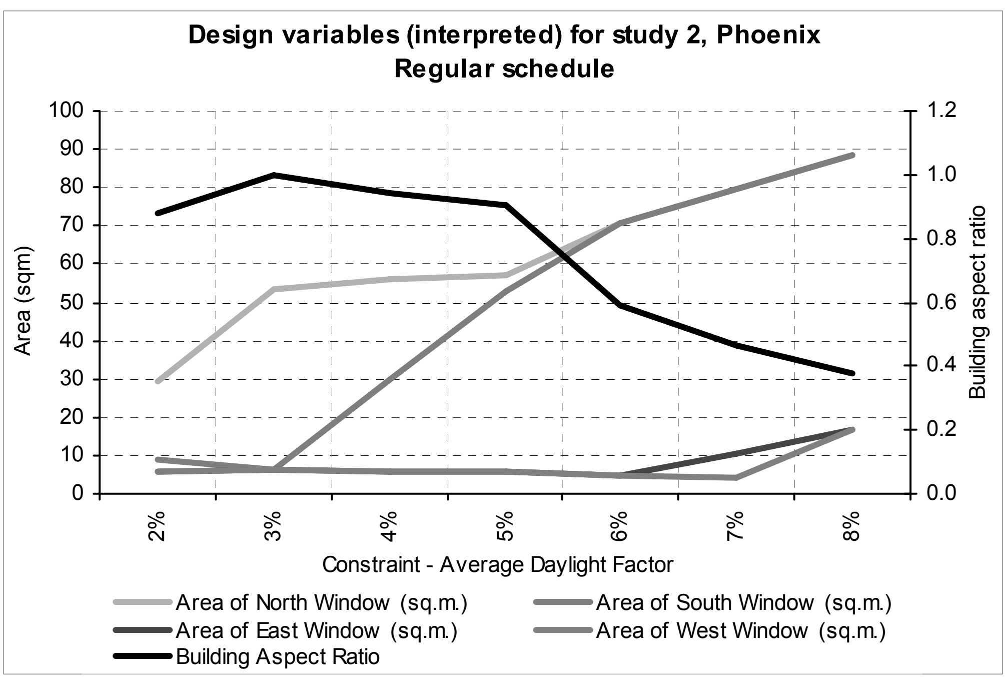


Figure 6.29: Interpreted design variables at best design points for study 2, Phoenix, regular schedule.

Table 6.10: Optimisation results, best design points for Phoenix with regular schedule.

	Average Daylight Factor						
Variables	***2%***	***3%***	***4%***	***5%***	***6%***	***7%***	***8%***
Building azimuth (°)	0.01	11.36	0.02	1.45	0.02	1.92	10.52
Building aspect ratio	0.88	1.00	0.94	0.91	0.59	0.46	0.38
North glazing fraction	0.46	0.89	0.90	0.90	0.90	0.90	0.90
South glazing fraction	0.13	0.10	0.48	0.84	0.90	0.90	0.90
East glazing fraction	0.10	0.10	0.10	0.10	0.10	0.26	0.45
West glazing fraction	0.10	0.10	0.10	0.10	0.10	0.10	0.45
Responses							
Energy Consumption (MWh)	95	97	100	103	106	110	118
Average Daylight Factor (%)	1.99	2.89	3.92	4.90	6.02	6.92	8.26
Building parameters							
Length of building (m)	23.83	22.36	23.01	23.50	29.05	32.84	36.35
Width of building (m)	20.98	22.36	21.73	21.28	17.21	15.23	13.76
North glazing area (m^2)	29.44	53.54	55.91	57.10	70.60	79.79	88.32
South glazing area (m^2)	8.66	6.04	29.66	53.14	70.60	79.79	88.23
East glazing area (m^2)	5.66	6.04	5.87	5.75	4.65	10.64	16.68
West glazing area (m^2)	5.66	6.04	5.87	5.75	4.65	4.11	16.68
Total glazing area (m^2)	49.43	71.65	97.31	121.7	150.5	174.3	209.9
Total wall area (m^2)	241.9	241.5	241.5	241.7	249.8	259.5	270.5

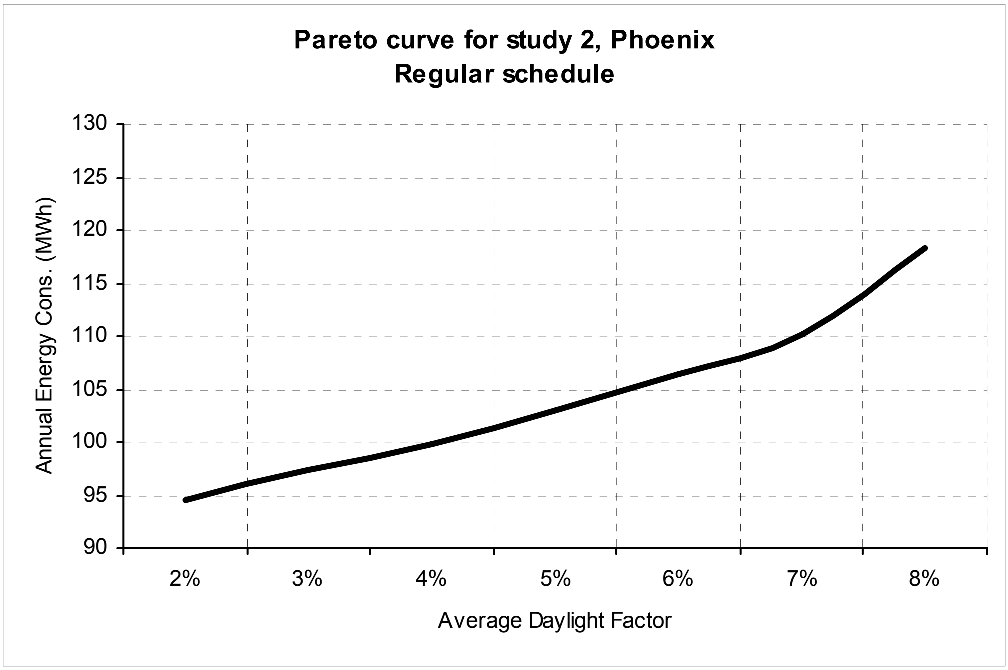


Figure 6.30: Pareto curve for study 2, Phoenix, regular schedule.

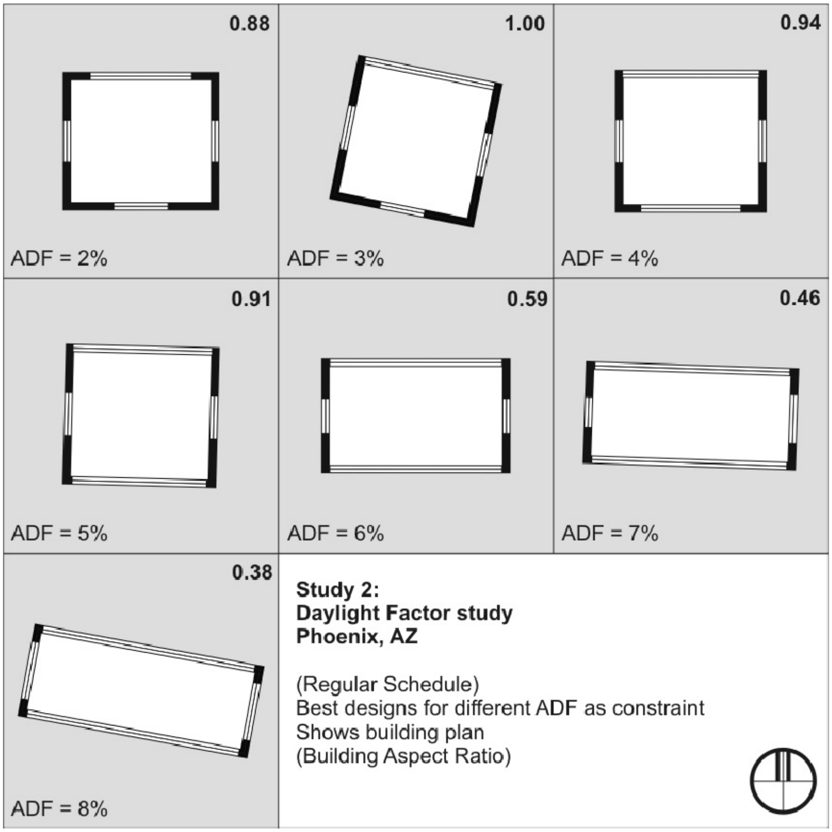


Figure 6.31: Building plans of optimisation results for Study 2, Phoenix: regular schedule.

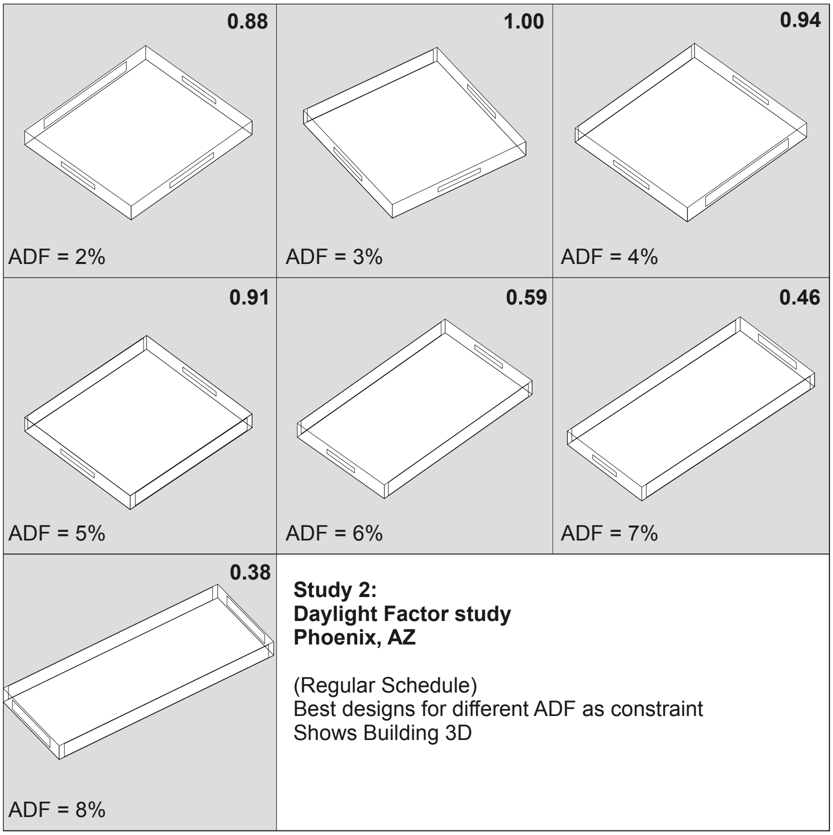


Figure 6.32: Building 3D models of optimisation results for Study 2, Phoenix: regular schedule.

Results for Phoenix, extended schedule

The drop in building aspect ratio for ADF values of 2% and 8% is more pronounced for buildings with extended schedule in Phoenix (see Figure 6.33 and Table 6.11) mainly due to the night-time climate parameters affecting the form. Building aspect ratio of 0.23 at ADF 8% is a sharp decrease from initial 0.98 at ADF 2% which can be said to be resulted from the need to dissipate heat during the extended period. Increased surface will allow the building to cool down quicker than a more compact form. North glazing dominates the composition of the building envelope followed by south glazing.

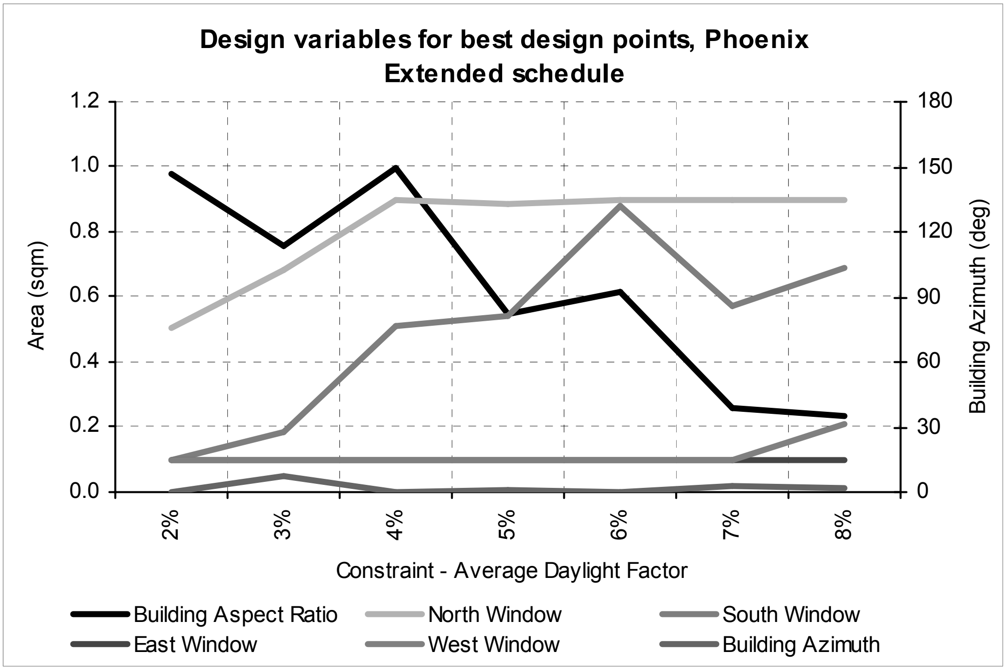


Figure 6.33: Design variables at best design points for study 2, Phoenix, extended schedule.

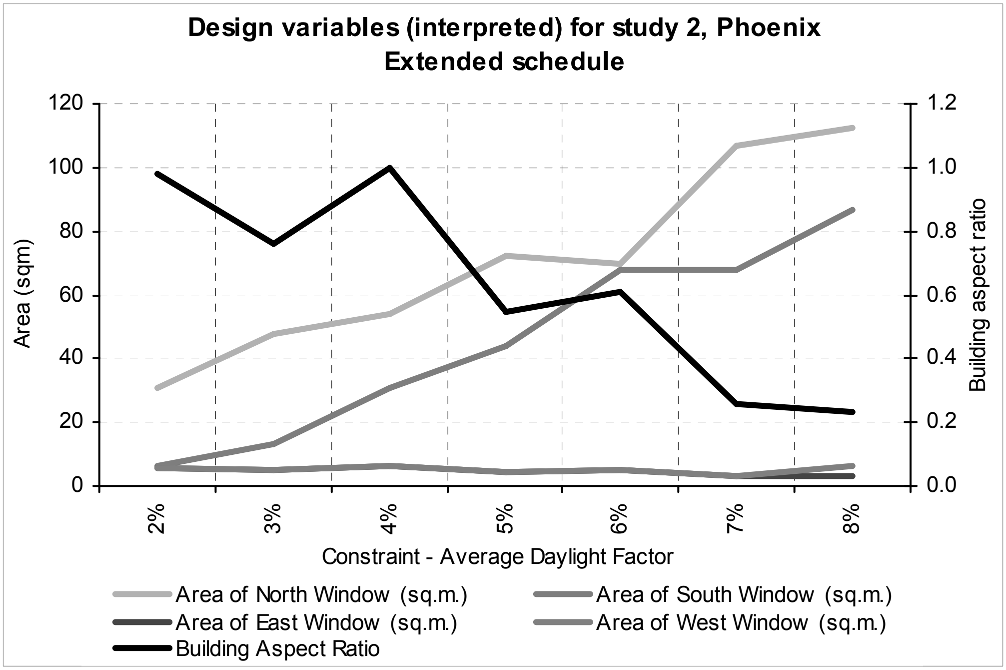


Figure 6.34: Interpreted design variables for study 2, Phoenix, extended schedule.

Table 6.11: Optimisation results, best design points for Phoenix with extended schedule.

	Average Daylight Factor						
Variables	***2%***	***3%***	***4%***	***5%***	***6%***	***7%***	***8%***
Building azimuth (º)	0.01	7.16	0.01	0.70	0.45	2.47	2.09
Building aspect ratio	0.98	0.76	1.00	0.55	0.61	0.26	0.23
North glazing fraction	0.51	0.69	0.90	0.89	0.90	0.90	0.90
South glazing fraction	0.10	0.19	0.51	0.54	0.88	0.57	0.69
East glazing fraction	0.10	0.10	0.10	0.10	0.10	0.10	0.10
West glazing fraction	0.10	0.10	0.10	0.10	0.10	0.10	0.21
Responses							
Energy Consumption (MWh)	110	112	115	118	121	125	129
Average Daylight Factor (%)	1.97	2.85	3.92	5.01	5.89	6.96	7.94
Building parameters							
Length of building (m)	22.62	25.69	22.36	30.24	28.57	43.87	46.26
Width of building (m)	22.10	19.46	22.36	16.53	17.50	11.40	10.81
North glazing area (m^2)	30.90	47.53	54.34	72.49	69.43	106.60	112.40
South glazing area (m^2)	6.11	12.95	30.98	44.17	68.12	68.08	86.40
East glazing area (m^2)	5.97	5.26	6.04	4.46	4.72	3.08	2.92
West glazing area (m^2)	5.97	5.26	6.04	4.46	4.72	3.08	6.15
Total glazing area (m^2)	48.94	70.99	97.39	125.58	147.00	180.83	207.87
Total wall area (m^2)	241.51	243.82	241.50	252.59	248.79	298.44	308.15

Glazing on east and west are always kept to a minimum to reduce heat gain through glazed areas (see Figure 6.34). Pareto curve for Phoenix with extended occupancy is shown in Figure 6.35 which expresses a rather linear relationship between ADF and total energy consumption.

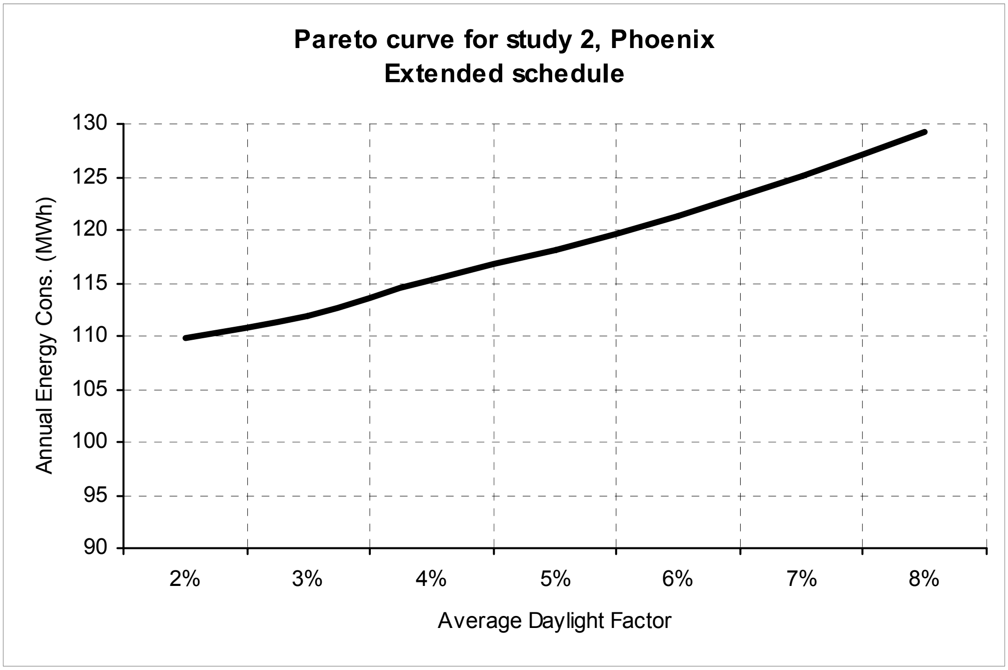


Figure 6.35: Pareto curve for study 2, Phoenix, extended schedule.

Discussions on multi-criteria optimisation, study 2

ADF as constraint to the optimisation problem has significant impact on the determination of building aspect ratio. Resultant forms are more pronounced in the case of Phoenix than in Kilkenny due to extremities in climatic parameters. Generally, lower values of ADF as constraint result in compact forms. On the other hand, higher values of ADF result in elongated form along east-west axis. It can be explained with the need for increased glazing to wall ratio at higher ADFs and to maintain a comfortable interior environment. The result conforms with the suggested guideline of 5% ADF in (DETR 2002). However, if accurate daylighting level, extracted from integrated simulation was implemented as a constraint, would have resulted in linear forms. The depth of building is increased in compact forms which reduces daylight availability on the far side of the room.

Pareto optimal curves considering ADF and annual energy consumption allows designer to make informed decisions. ADF value of 5% usually strikes a balance between the required level of daylighting and optimised annual energy consumption in all four cases in study 2. It further validates the suggestions that ADF is an effective mechanism in ensuring the design of energy efficient forms during early stages of design, found in literature (Loe and Mansfield 1998; DETR 2002). Extended operation of buildings results in significantly different building aspect ratio and glazing parameters. In the case of Phoenix, the resultant forms are less compact with increased wall areas for easy dissipation of heat during extended hours. A combination of glazing on north and south are preferred as opposed to north as found in regular occupancy. Absence of solar heat gain on south side during extended hours can be considered responsible for the effect. In Kilkenny, similar trend can be seen. A combination of north and south glazing are preferred due to the fact glazing on north and south has same effect of heat loss during extended occupancy at night time. An increase in north glazing area can also result from the longer daylight availability during summer which is brought into the scene because of the extended occupancy.

Study 3: multi-zone building

This study involves the investigation of the applicability of ArDOT in a multi-zone environmental design problem. The objective for this study is to find the optimum form of a commercial building comprising of four office zones and one core zone as in Figure 6.36. The building is let to operate on regular office schedule from 09:00 am to 05:00 pm. The floor area of each office zone is allowed to vary up to 250 sqm, while the floor area of the core zone remains constant at 144 sqm. Core zone does not have any external surface, but connects the remaining zones thermally. Daylighting and artificial lighting is modelled as described in Chapter 5. ADF for each of the office zones has a lower bound of 4.5 and upper bound of 5.0. Parameters for optimisation algorithm can be found in Table 6.2. Optimisation is modelled as minimisation of annual energy consumption for the facility that includes heating, cooling and electric lighting. Lower bound, initial value and upper bound of variables for study 3 can be found in Table 6.12. The optimisation problem is run for both Kilkenny and Phoenix.

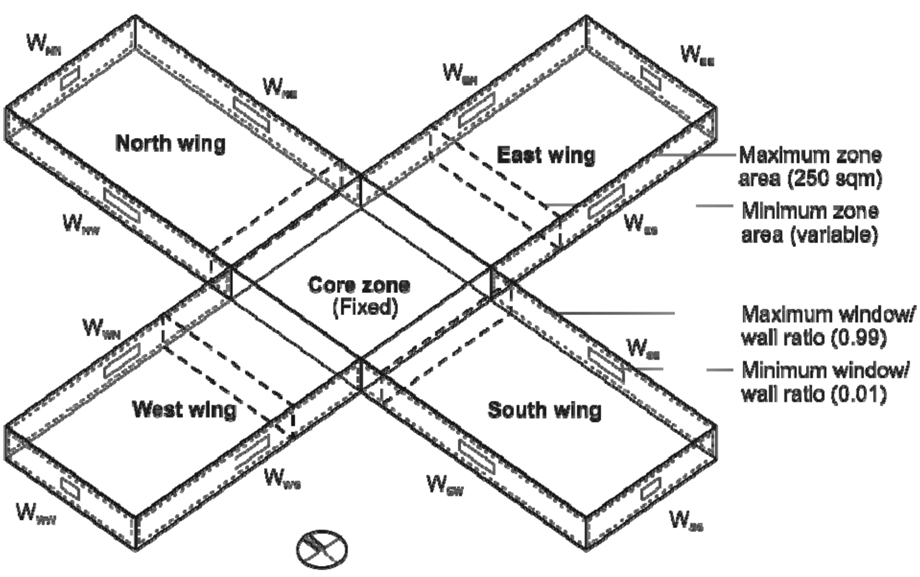


Figure 6.36: Study 3, multi-zone problem definition (not to scale).

Table 6.12: Lower bound, initial value and upper bound of variables for study 3.

Variables	***Lower bound***	***Initial value***	***Upper bound***
Core zone area	144	144	144
North wing zone area (sqm)	5	50	250
South wing zone area (sqm)	5	50	250
East wing zone area (sqm)	60	200	250
West wing zone area (sqm)	60	200	250
North wing north window*	0.01	0.5	0.99
North wing east window*	0.01	0.05	0.99
North wing west window*	0.01	0.05	0.99
South wing south window*	0.01	0.5	0.99
South wing east window*	0.01	0.05	0.99
South wing west window*	0.01	0.05	0.99
East wing east window*	0.01	0.05	0.99
East wing north window*	0.01	0.5	0.99
East wing south window*	0.01	0.5	0.99
West wing west window*	0.01	0.05	0.99
West wing north window*	0.01	0.5	0.99
West wing south window*	0.01	0.5	0.99

Study 3 results for Kilkenny

Study 3 results for Kilkenny suggest a building form elongated in the East-West axis. Areas for the East and the West wings are 189 and 159 sqm respectively compared to 108 for the North wing and 40 sqm for the South wing. Shadow study of the resulting building form for Kilkenny at different times of the year in Figure 6.38 supports the selection of East and West zone area as dominant variables in the optimisation run. Windows on the cardinal North and South walls are considerably larger than on the East and West walls, suggesting the significance of the North and South façades in achieving energy efficient building forms. Values of design variables for optimised result are given in Table 6.13. The 3D model of the resulting form is shown in Figure 6.37.

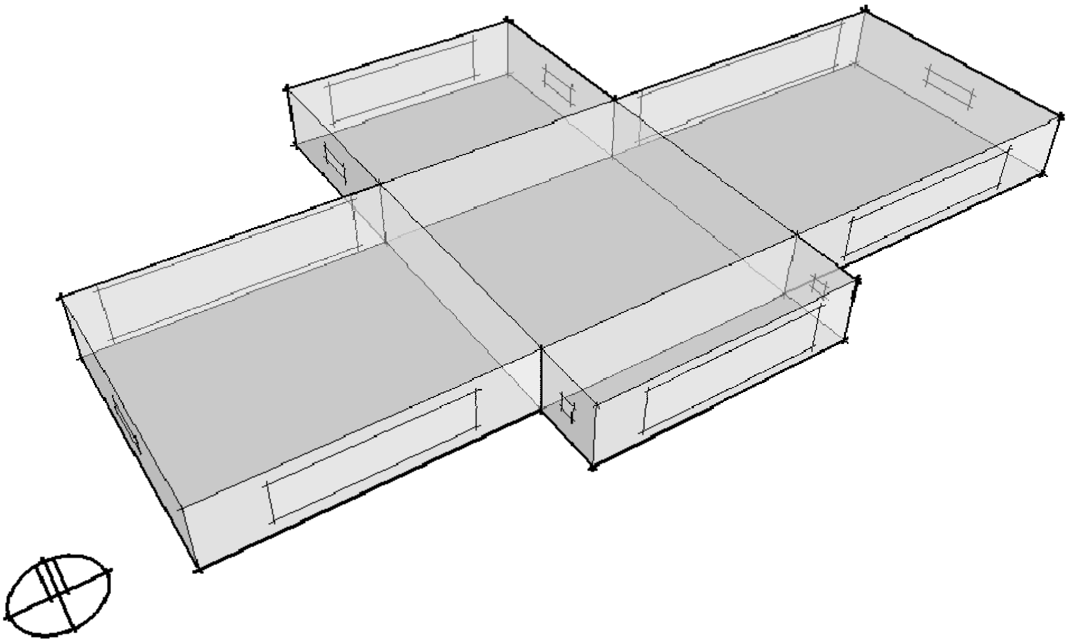


Figure 6.37: 3D model of the building for study 3 optimisation run for Kilkenny.

Table 6.13: Values of Design variables for study 3 optimisation run for Kilkenny.

Variables	***Value***	***Variables***	***Value***
North wing zone area (sqm)	108	South wing east window*	0.030
South wing zone area (sqm)	40	South wing west window*	0.039
East wing zone area (sqm)	189	East wing east window*	0.051
West wing zone area (sqm)	159	East wing north window*	0.642
North wing north window*	0.436	East wing south window*	0.356
North wing east window*	0.064	West wing west window*	0.037
North wing west window*	0.044	West wing north window*	0.639
South wing south window*	0.453	West wing south window*	0.347
* Windows are modelled as a fraction of wall area			

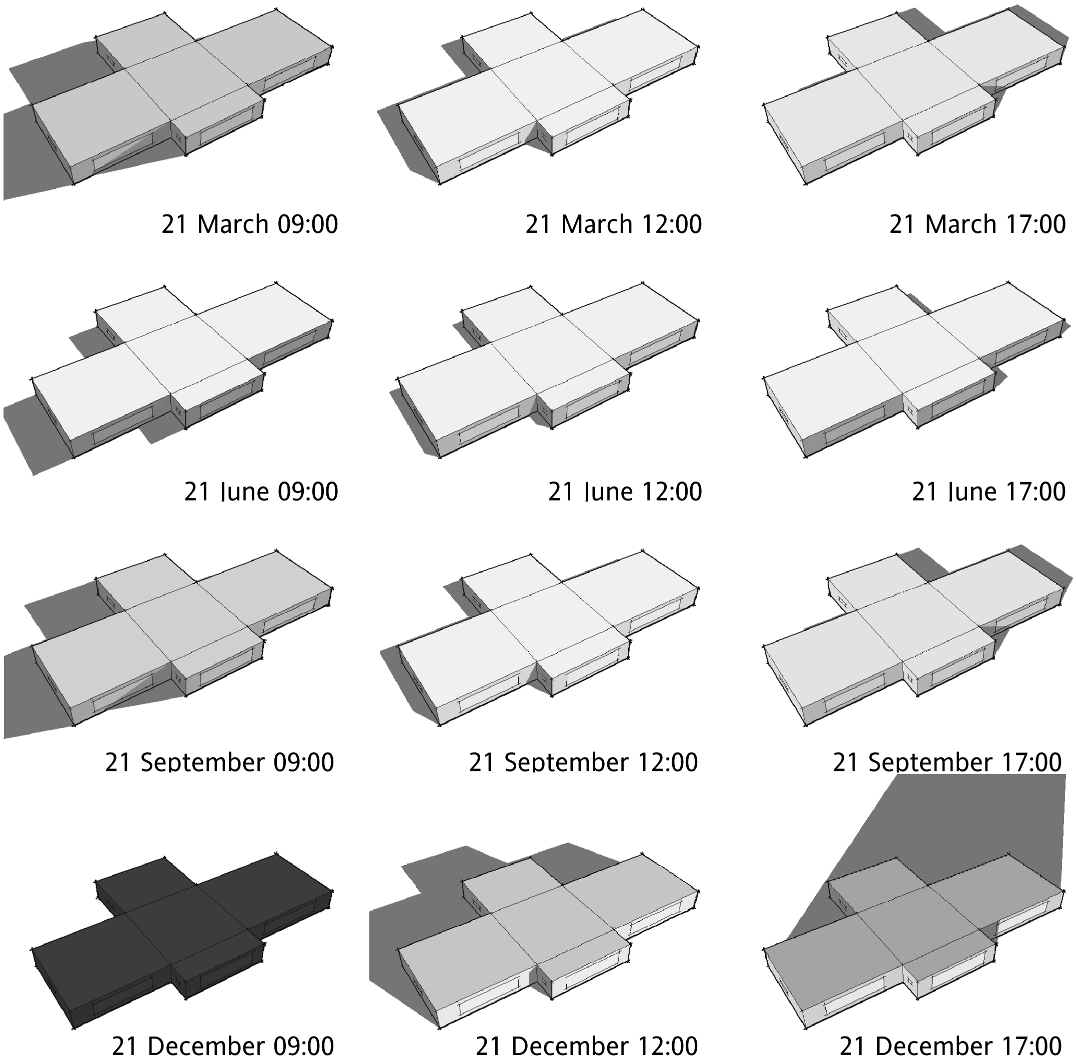


Figure 6.38: Shadow study of resulting building form for Kilkenny at different times of the year.

Study 3 results for Phoenix

Study 3 results for Phoenix suggest a building form elongated in the East-West axis, more pronounced than that of Kilkenny. The form is representative of the climate of Phoenix. Areas for the East and the West wings are 198 and 181 sqm respectively compared to 96 for the North wing and 47 sqm for the South wing. Shadow study of the resulting building form for Phoenix at different times of the year in Figure 6.40 supports the selection of East and West zone area as dominant variables in the optimisation run. As in Kilkenny, the windows on the cardinal North and South walls are considerably larger than on the East and West walls. Values of design variables for optimised result are given in Table 6.14. The 3D model of the resulting form is shown in Figure 6.39.

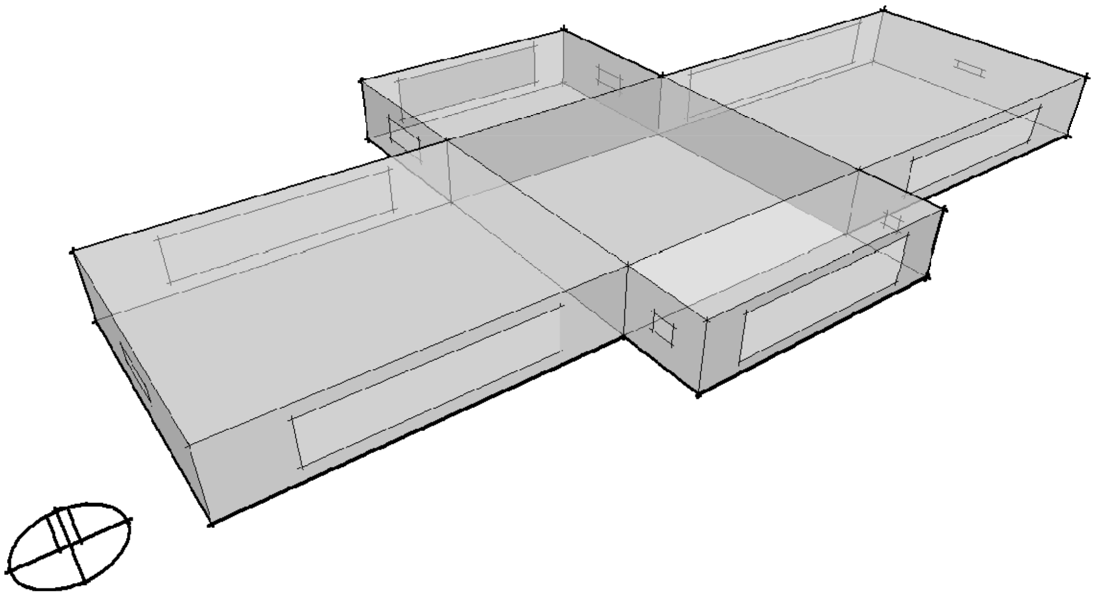


Figure 6.39: 3D model of the building for study 3 optimisation run for Phoenix.

Table 6.14: Values of Design variables for study 3 optimisation run for Phoenix.

Variables	***Value***	***Variables***	***Value***
North wing zone area (sqm)	96	South wing east window*	0.027
South wing zone area (sqm)	47	South wing west window*	0.061
East wing zone area (sqm)	198	East wing east window*	0.014
West wing zone area (sqm)	181	East wing north window*	0.589
North wing north window*	0.456	East wing south window*	0.317
North wing east window*	0.053	West wing west window*	0.045
North wing west window*	0.122	West wing north window*	0.392
South wing south window*	0.416	West wing south window*	0.390
* Windows are modelled as a fraction of wall area			

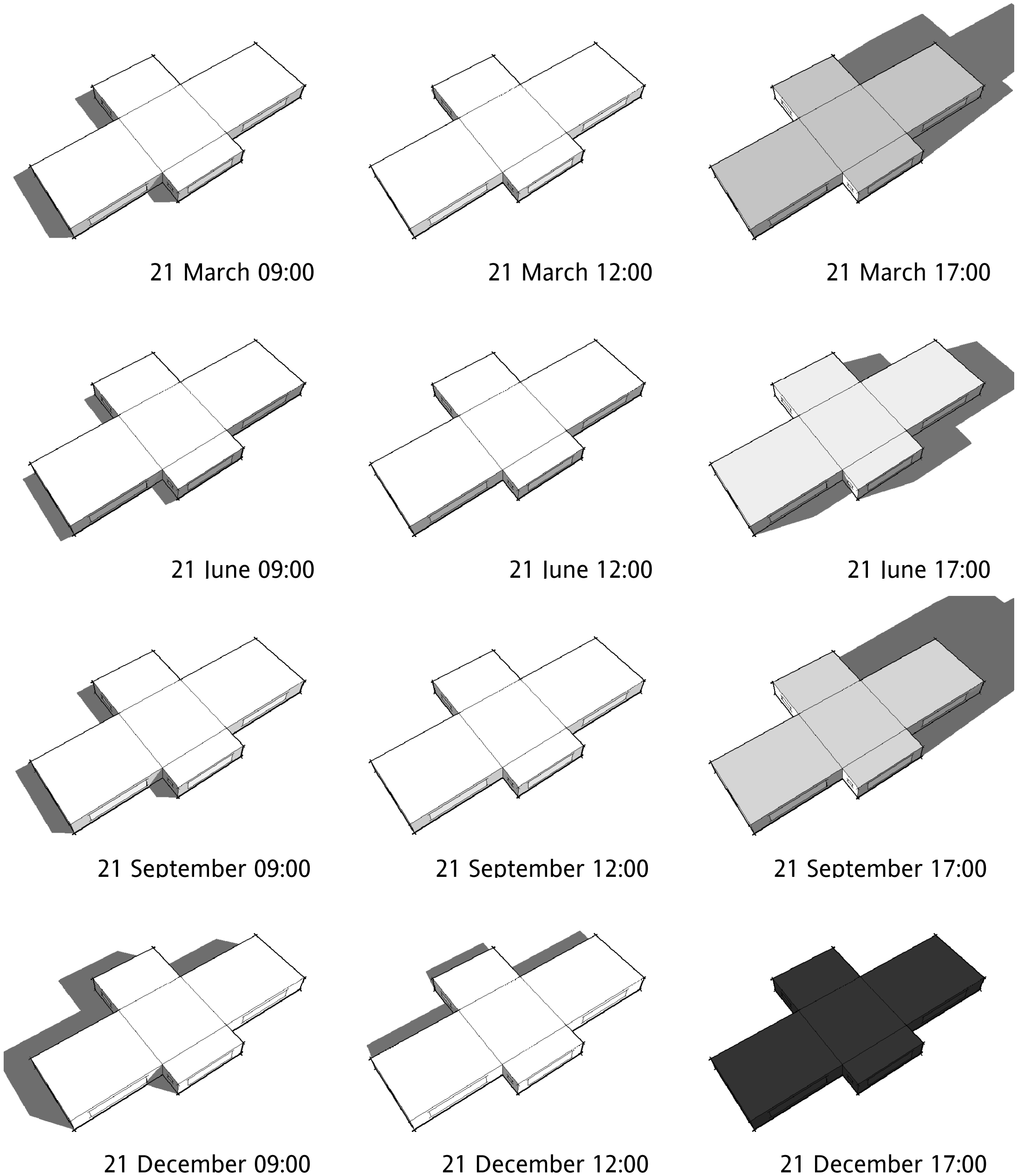


Figure 6.40: Shadow study of resulting building form for Phoenix at different times of the year.

Discussions on multi-zone optimisation, study 3

Results from optimisation runs support conventional wisdom in selection of building form for Northern hemisphere where South façade has higher solar irradiance value than the rest. Window to wall area ratios in different cardinal sides in both climates show strong correlation with the respective climate. With an increase in the number of variables in study 3, computational time increases significantly. Further investigations can be carried out on the decomposition of the optimisation problem to reduce the dimensions of the solution space.

Summary

The importance of pre-design climate analysis and visualisation of climate parameters for environmental design of buildings during early stages are described. The methodology of climate analysis using statistically averaged weather data is elaborated in conjunction with Appendix G. The reasons behind the selection of climatic locations for implementation of the framework are given. Pre-design climate analysis of the selected locations: Kilkenny, Ireland and Phoenix, USA forms the basis for interpretation of optimisation results.

Three studies on single and multi criteria optimisation of environmental design of buildings show the effectiveness of the application of mathematical optimisation in early stages of design. Study 1, convergence study essentially establishes the efficiency of gradient based algorithms in reaching optimum from different starting points. Unlike computationally expensive non gradient based methods (Wetter and Wright 2003), gradient based methods such as SQP (Sequential Quadratic Programming), as tested in this thesis are efficient in converging to optimum. Results from study 2, a multi-criteria optimisation of study 1 problem in different occupancy patterns conforms to the conventional wisdom in environmental design of buildings in respective climates. Application of Pareto optimality in multi-criteria optimisation shows how designers can perform informed decision making during design development. Study 3 investigates the application of the framework in multi-zone environmental design problems.

The diverse studies on environmental design optimisation in this chapter show the effectiveness of ideal HVAC air system in the modelling of simulation problems. It also proves applicability of EnergyPlus as a response generator in optimisation. The reported discontinuity of EnergyPlus responses in (Wetter 2005) is because of the use of detailed HVAC system in simulation modelling. Design tasks at early architectural design stage concerns with finding solutions for building form, function and fabric that minimise energy loads and energy use (Griffith *et al.* 2003) than fine tuning HVAC systems. Hence the use of detailed HVAC systems based on iterative solvers may introduce noise which in effect can make optimisation algorithms getting trapped at local minima during optimisation in early stages of design.

The implementation of the integrated framework developed as part of this thesis, demonstrates the fact that integration and interoperability can be achieved during early stages of design. Interpolation of architectural standards for use as defaults during the translation between IFC and simulation input files show that the detailed building simulation capabilities can be brought to architectural designers at early stages of design; as opposed to the conventional suggestions on either (a) the use of simplified simulation programmes (Augenbroe 2002) or (b) incorporating simulators during early design exploration (McElroy and Clark 1999).

References

Augenbroe, G. (2002). Trends in building simulation. *Building and Environment*, **37**(8-9), 891-902.

De Schiller, S. and Evans, J.M. (1996). Training architects and planners to design with urban microclimates. *Atmospheric Environment*, **30**(3), 449-454.

DETR (2002). *Desktop Guide to Daylighting - for Architects*. 245. Department of Environment, Transport and Regions, United Kingdom.

DOE (2004). *Auxiliary EnergyPlus Programs: To Increase Your Efficiency at Using EnergyPlus*. Berkeley: Department of Energy.

Griffith, B., Pless, S., Talbert, B., Deru, M. and Torcellini, P. (2003). *Energy Design Analysis and Evaluation of a Proposed Air Rescue and Fire Fighting Administration Building for Teterboro Airport*. NREL/TP-550-33294. Colorado: National Renewable Energy Laboratory.

Jo, J.H. and Gero, J.S. (1998). Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering*, **12**(3), 149-162.

Köppen, W.P. (1884). Die Warmezonen der Erde, nach der Dauer der heissen, gemassigten und kaltern Zeit und nach der Wirkung der Warme auf die organische Welt betrachtet. *Meteorologische Zeitschrift*, 215-226.

LANL (2002). *LANL Sustainable Design Guide*. Los Alamos: Los Alamos National Laboratory.

Leemans, R. and Cramer, W.P. (1991). *The IIASA Database for Mean Monthly Values of Temperature, Precipitation and Cloudiness of a Global Terrestrial Grid*. RR-91-18. Laxenburg, Austria: IIASA.

Loe, D. and Mansfield, K.P. (1998). *Daylighting Design in Architecture: Making the Most of a Natural Resource*. Garston, Watford: Building Research Establishment.

Marsh, A. (2003). ECOTECT and EnergyPlus. *Building Energy Simulation User News*, **24**(6), 2-3.

McElroy, L.B. and Clark, J.A. (1999). Embedding simulation within energy sector businesses. In: *BS '99: Sixth International IBPSA Conference*, Kyoto, Japan, 1197-1202.

Miettinen, K.M. (1999). *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic Publishers.

Mourshed, M.M., Kelliher, D. and Keane, M. (2003). A tool to optimise environmental design of buildings. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 919-926.

O'Sullivan, B.A. (1999). *Constraint-Aided Conceptual Design*. PhD Thesis. Cork: University College Cork.

Ogunsote, O.O. and Prucnal-Ogunsote, B. (2003). Choice of a thermal index for architectural design with climate in Nigeria. *Habitat International*, **27**(1), 63-81.

Petrie, C.J., Webster, T.A. and Cutkosky, M.R. (1995). Using Pareto optimality to coordinate distributed agents. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, **9**(4), 269-281.

Rabah, K. (2005). Development of energy-efficient passive solar building design in Nicosia Cyprus. *Renewable Energy*, **30**(6), 937-956.

Radford, A.D. and Gero, J.S. (1988). *Design by Optimization in Architecture, Building and Construction*. New York: Van Nostrand Reinhold Co.

Robinson, D. (2003). Climate as a pre-design tool. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 1109-1116.

Trewartha, G.T. (1968). *An introduction to climate*. New York: McGraw-Hill.

Wetter, M. (2005). BuildOpt--a new building energy simulation program that is built on smooth models. *Building and Environment*, **40**(8), 1085-1092.

Wetter, M. and Polak, E. (2005). Building design optimization using a convergent pattern search algorithm with adaptive precision simulations. *Energy and Buildings*, **37**(6), 603-612.

Wetter, M. and Wright, J. (2003). Comparison of a generalized pattern search and a genetic algorithm optimization method. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 1401-1408.

Wetter, M. and Wright, J. (2004). A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment*, **39**(8), 989-999.

Wright, J. and Loosemore, H. (2001). The multi-criterion optimization of building thermal design and control. In: *BS2001: Seventh International IBPSA Conference*, Rio de Janeiro, Brazil, 873-880.

Zain-Ahmed, A., Sayigh, A.A.M., Surendran, P.N. and Othman, M.Y. (1998). The bioclimatic design approach to low-energy buildings in the Klang Valley, Malaysia. *Renewable Energy*, **15**(1-4), 437-440.

Chapter 7
Conclusions

This chapter considers the presented work and discusses the results in the context of the stated objectives in Chapter 1. Referring to the work done in this thesis the level of success achieved and the limitations overcome are discussed here. Finally, the role of interoperability in multi-domain simulation and optimisation within this thesis and in the wider context of sustainability is analysed to ascertain future avenues of research.

Findings

The work reported in this thesis started with the search for *the best possible way to integrate energy efficiency concepts during architectural design*. Four observations helped to formulate the framework: (a) data and definitions are ambiguous during early stages of design; (b) building simulation allows performance evaluation of buildings with regard to energy efficiency; (c) interoperable standards allow integration of current and future tools; and (d) optimisation techniques help to search the solution space effectively. Development of an information system was the natural choice to incorporate the observations.

As with any information systems development, the interoperability-based framework developed in this thesis connects multi-disciplinary domains through interconnected software components. Therefore, the discussion presented below contains overlapping related scenarios.

Integrating simulation in design

Standards based mapping allowed integration of simulation tools which would have been otherwise impossible to implement. It reinforces the incremental design development model of experience/heuristics based design theories. The number of design variables in simulation-based approaches makes it difficult to visualise the combined effect of interdependence. By implementing mathematical optimisation and concepts from decision sciences such as Pareto optimality, the framework enabled visualisation for informed decision making.

Modelling during early design stages

Information about design problems are ill-defined during early stages of building lifecycle (Hendricx 2000). Detailed-based building simulation, on the other hand, requires building parameters to be known which is not readily available at this stage. Papamichael *et al.* used default value selector in BDA (Papamichael *et al.* 1997) for the automatic assignment of default values of parameters required as input by the simulation tools. Similar methods have been used here and found to be effective in integrating simulation at early stages of design. The rules for the selection of default values follow building codes, standards and recommended practice, taken from a number of sources, as described in Chapter 5.

Formal optimisation as a design activity

The availability of performance evaluation at early stages of design through integration of simulation, described in sections 0 and 0 allowed formulation of architectural design as mathematical optimisation problems. The results and discussions in Chapter 6 showed the effectiveness of the framework in exploring the solution space. Solutions generated also conform to the conventional wisdom in environmental design of buildings. It has been observed that a number of solutions exist which are near to optimum allowing the designer the opportunity to explore the solution space, which is vital to reinforce the creative process in design. Selection of design solutions based on the visualisation of optimisation results ensures conscious and informed decision making for enhancing sustainability.

With regard to optimisation strategy, gradient based methods have been tested although other methods have been implemented in this thesis. Contrary to the suggestions by researchers on the suitability of nature based algorithms (Caldas and Soibelman 2003) in architectural design problems, gradient based methods have been observed as producing reliable results efficiently and rather inexpensively.

Information model for process specification

IFC or related information models do not offer a separate layer for incorporating certain tasks or processes as it would otherwise complicate the development of the specification. It may also leave such information models vulnerable to the changes in work practices. Although, process-driven information models have been suggested by some researchers (de Wilde *et al.* 2002), it is found in this thesis that process integration is best left to the application developers concerned. This is also emphasised by Bazjanac (Bazjanac 2004). To incorporate optimisation as part of the design process, a separate information model independent of IFCs has been developed; referred to as ardML (**ArD**OT **M**odelling **L**anguage). ardML is an XML schema based information model for specification of optimisation parameters which is not available in IFC. It contains application logic and allows messaging between vertically integrated applications; e.g. applications sharing a common definition of objects in the same discipline.

Interoperability-based software environment

Building on previous attempts at developing interoperability-based software environments such as COMBINE(Augenbroe 1994), SEMPER(Lam *et al.* 2004) and WISPER(Faraj *et al.* 2000), a process centric approach was taken in this thesis. This enables complex analysis activities, e.g. optimisation of environmental design of buildings to become part of the interoperable framework. The concept of optimisation implemented here is based on multiple-domain performance assessment of buildings. The automation of the design task required implementation of application-specific information model (ardML) independent of IFCs. The role of XML in this thesis was to enable intra-software communication and messaging whereas IFC enabled capture and storage of the data semantics for wider integration.

The framework supports the claims of IFC development efforts in achieving interoperability by using the building model created during concept development in the optimisation of environmental design of buildings.

The ArDOT calls to simulation engines were implemented using an *adhoc* approach involving customised mapping of IFC objects through ardML into EnergyPlus definitions. Developments in IFC-based simulation engines will eliminate the need for translation, enabling seamless integration. Calls to simulation engines will only require referencing the IFC object or p21 file in the near future.

Remarks

Some general remarks on interoperability standards and optimisation, stemmed out of this research are made here.

XML and EXPRESS

The hypothesis that XML and EXPRESS can co-exist in an information world is shown to be feasible. Part 28 (ISO 2003a), the XML binding specification of STEP and ifcXML version 1.0 (Nisbet and Liebich 2005b) has been published at a latter stage of research; were not investigated in this thesis. The use of ardML shows clearly defined boundaries for the two specifications where XML has the use for application specific messaging and EXPRESS is capable of representing rich semantics that is required for industry integration.

Process vs. product and single vs. multiple

Development of industry-wide information models need to be based on the principles of the lowest common denominator of information and semantics required by the wider community. Process driven approach (de Wilde *et al.* 2002) involving incorporation of evolving tasks will only complicate the development process. Moreover, the ways to accomplish tasks vary among professionals of same origin. Some levels of consensus on process-centric developments can be achieved within the same professional group but not across the wider industry.

The point on the feasibility of single data model vs. multiple data models (Amor and Faraj 2001) is appropriately made. It is widely understood that for industry-wide interoperability there is a need for a common data model such as IFC. There should also be some mechanisms for dynamic incorporation of project/task/domain specific data model. Property set definitions in the IFC can offer significant leverage in achieving wider integration and synergy.

Optimisation for informed decision making

Incorporating optimisation in the decision making process is another model for integration of simulation in design. The ability to visualise a large set of variables will invariably increase the richness of the decisions made.

Integrated software environments for teaching and learning

A final remark on the role of integrated software environment such as ArDOT in the teaching and learning of energy efficiency concepts and sustainability is appropriate. An example of a recent implementation can found in (Plume and Mitchell 2007).

Future work

Following the implementation and investigations carried out in this thesis, other avenues can be explored. Some of these are presented as areas of future work.

Performance-based design optimisation

This thesis involved the study of Annual Energy Consumption and Average Daylight Factor in guiding the optimisation process. The success of this combination of detailed-based simulation with a rule of thumb shows the effectiveness of the method in design exploration. However, further studies need to be conducted on the development of evaluation strategies.

Sensitivity of responses from simulation engines

Responses generated from EnergyPlus seemed to be of reasonable resolution for use in gradient based design when Purchased-air based system is used. Detailed HVAC systems may result in discontinuous solutions because: (a) they are based on iterative solvers and (b) discreet controls are employed in real systems. Detailed analysis of sensitivity of responses needs to be made, particularly when system sizing is used.

Efficiency of optimisation algorithms

Only Sequential Quadratic Programming method has been tested in this thesis. Other optimisation algorithms such as RSA (Response Surface Approximate), DOE (Design of Experiments) are known for their efficiency in engineering design optimisation. Investigations on the applicability of these algorithms in architectural design optimisation can be carried out. The comparison between different algorithms in solving particular architectural design problems can be attempted as well.

Horizontal integration of disciplines

Two different simulation engines from the domain of environmental design have been successfully integrated in this thesis. Development of application-specific information model; i.e. ardML was straightforward. Seamless integration of application from multiple horizontal domains needs to be investigated with respect to the suitability of a single information model.

References

Amor, R. and Faraj, I. (2001). Misconceptions about integrated project databases. *Electronic Journal of Information Technology in Construction*, **6**, 57-68.

Augenbroe, G. (1994). An overview of the COMBINE project. In: *The first European Conference on Product and Process Modelling in the Building Industry*, Dresden, Germany, 5-7 October 1994, 547-554.

Bazjanac, V. (2004). Building energy performance simulation as part of interoperable software environments. *Building and Environment*, **39**(8), 879-883.

Caldas, C.H. and Soibelman, L. (2003). Automating hierarchical document classification for construction management information systems. *Automation in Construction*, **12**(4), 395-406.

de Wilde, P., Augenbroe, G. and van der Voorden, M. (2002). Design analysis integration: supporting the selection of energy saving building components. *Building and Environment*, **37**(8-9), 807-816.

Faraj, I., Alshawi, M., Aouad, G., Child, T. and Underwood, J. (2000). An industry foundation classes Web-based collaborative construction computer environment: WISPER. *Automation in Construction*, **10**(1), 79-99.

Hendricx, A. (2000). *A core object model for architectural design*. PhD Thesis. Leuven: Katholieke Universiteit Leuven.

ISO (2003). *Industrial automation systems and integration - Product data representation and exchange - Part 28: Implementation methods: XML representations of EXPRESS schemas and data*. ISO 10303-28:2003. Geneva: ISO.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004). SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. *Automation in Construction*, **13**(5), 651-663.

Nisbet, N. and Liebich, T. (2005). *ifcXML Implementation Guide*. Washington: International Alliance for Interoperability.

Papamichael, K., LaPorta, J. and Chauvet, H. (1997). Building Design Advisor: automated integration of multiple simulation tools. *Automation in Construction*, **6**(4), 341-352.

Plume, J. and Mitchell, J. (2007). Collaborative design using a shared IFC building model--Learning from experience. *Automation in Construction*, **16**(1), 28-36.

Appendix A
IDF material and construction specifications

Material and construction specifications used in EnergyPlus simulations in this thesis are given here. Datasets are taken from EnergyPlus BLAST construction set and edited by the author.

IDF specifications for materials

```
MATERIAL:REGULAR,
  C5-100mm HW CONCRETE,  !- Material Name
  MediumRough,         !- Roughness
  0.100,               !- Thickness {m}
  1.720,               !- Conductivity {w/m-K}
  2242.580,            !- Density {kg/m3}
  830.0,               !- Specific Heat {J/kg-K}
  0.90,                !- Thermal Emittance
  0.650,               !- Solar Absorptance
  0.650;               !- Visible Absorptance
MATERIAL:AIR,
  E4-CEILING AIRSPACE,  !- Material Name
  0.170;               !- Resistance {m2-K/w}
MATERIAL:REGULAR,
  E5-ACOUSTIC TILE,     !- Material Name
  MediumSmooth,        !- Roughness
  0.0099999998,        !- Thickness {m}
  0.059999999,         !- Conductivity {w/m-K}
  480.550,             !- Density {kg/m3}
  830.0,               !- Specific Heat {J/kg-K}
  0.9,                 !- Thermal Emittance
  0.32,                !- Solar Absorptance
  0.32;                !- Visible Absorptance
MATERIAL:REGULAR,
  B10-50mm WOOD,   !- Material Name
  MediumSmooth,        !- Roughness
  0.050,               !- Thickness {m}
  0.120,               !- Conductivity {w/m-K}
  592.68,              !- Density {kg/m3}
  2510.0,              !- Specific Heat {J/kg-K}
  0.9,                 !- Thermal Emittance
  0.78,                !- Solar Absorptance
  0.78;                !- Visible Absorptance
MATERIAL:REGULAR,
```

```
  B11-75mm WOOD,    !- Material Name
  MediumSmooth,       !- Roughness
  0.075,              !- Thickness {m}
  0.12,               !- Conductivity {w/m-K}
  592.68,             !- Density {kg/m3}
  2510.0,             !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.78,               !- Solar Absorptance
  0.78;               !- Visible Absorptance
MATERIAL:REGULAR,
  C7-200mm LW CONCRETE BLOCK,  !- Material Name
  Rough,              !- Roughness
  0.200,              !- Thickness {m}
  0.57,               !- Conductivity {w/m-K}
  608.7,              !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.65,               !- Solar Absorptance
  0.65;               !- Visible Absorptance
MATERIAL:REGULAR,
  WOOD-3mm HARDWOOD,  !- Material Name
  MediumSmooth,       !- Roughness
  0.0031999999,       !- Thickness {m}
  0.15,               !- Conductivity {w/m-K}
  720.83,             !- Density {kg/m3}
  1250.,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.78,               !- Solar Absorptance
  0.78;               !- Visible Absorptance
MATERIAL:WINDOWGAS,
  WinAirB1-AIRSPACE RESISTANCE,  !- Material Name
  AIR,                !- Gas Type
  0.013000000;        !- Gap Width {m} = 1/2 inch
MATERIAL:REGULAR,
  E1-19mm PLASTER OR GYP BOARD,  !- Material Name
  Smooth,             !- Roughness
  0.01905,            !- Thickness {m}
  0.72,               !- Conductivity {w/m-K}
  1601.84,            !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.92,               !- Solar Absorptance
  0.92;               !- Visible Absorptance
MATERIAL:REGULAR,
  C2-100mm LW CONCRETE BLOCK,  !- Material Name
  MediumRough,        !- Roughness
  0.100,              !- Thickness {m}
  0.38,               !- Conductivity {w/m-K}
  608.7,              !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
```

```
  0.65,                 !- Solar Absorptance
  0.65;                 !- Visible Absorptance
MATERIAL:REGULAR,
  B7-25mm WOOD,     !- Material Name
  MediumSmooth,         !- Roughness
  0.025,                !- Thickness {m}
  0.12,                 !- Conductivity {w/m-K}
  592.68,               !- Density {kg/m3}
  2510.0,               !- Specific Heat {J/kg-K}
  0.9,                  !- Thermal Emittance
  0.78,                 !- Solar Absorptance
  0.78;                 !- Visible Absorptance
MATERIAL:AIR,
  B1-AIRSPACE RESISTANCE,  !- Material Name
  0.16;               !- Resistance {m2-K/w}
MATERIAL:REGULAR,
  E2-50mm SLAG OR STONE,  !- Material Name
  Rough,                !- Roughness
  0.0127,               !- Thickness {m}
  1.43,                 !- Conductivity {w/m-K}
  881.01,               !- Density {kg/m3}
  1670.0,               !- Specific Heat {J/kg-K}
  0.9,                  !- Thermal Emittance
  0.55,                 !- Solar Absorptance
  0.55;                 !- Visible Absorptance
MATERIAL:REGULAR,
  E3-9mm FELT AND MEMBRANE,  !- Material Name
  Rough,                !- Roughness
  0.0094999997,         !- Thickness {m}
  0.19,                 !- Conductivity {w/m-K}
  1121.29,              !- Density {kg/m3}
  1670.0,               !- Specific Heat {J/kg-K}
  0.9,                  !- Thermal Emittance
  0.75,                 !- Solar Absorptance
  0.75;                 !- Visible Absorptance
MATERIAL:REGULAR,
  B5-25mm DENSE INSULATION,  !- Material Name
  VeryRough,            !- Roughness
  0.02,                 !- Thickness {m}
  0.039999999,          !- Conductivity {w/m-K}
  91.3,                 !- Density {kg/m3}
  830.0,                !- Specific Heat {J/kg-K}
  0.9,                  !- Thermal Emittance
  0.5,                  !- Solar Absorptance
  0.5;                  !- Visible Absorptance
MATERIAL:REGULAR,
  B6-50mm DENSE INSULATION,  !- Material Name
  VeryRough,            !- Roughness
  0.050,                !- Thickness {m}
  0.039999999,          !- Conductivity {w/m-K}
  91.3,                 !- Density {kg/m3}
```

```
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.5,                !- Solar Absorptance
  0.5;                !- Visible Absorptance
MATERIAL:REGULAR,
  B9-100mm WOOD,     !- Material Name
  MediumSmooth,       !- Roughness
  0.100,              !- Thickness {m}
  0.12,               !- Conductivity {w/m-K}
  592.68,             !- Density {kg/m3}
  2510.0,             !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.78,               !- Solar Absorptance
  0.78;               !- Visible Absorptance
MATERIAL:REGULAR,
  C13-150mm HW CONCRETE,  !- Material Name
  MediumRough,        !- Roughness
  0.15,               !- Thickness {m}
  1.72,               !- Conductivity {w/m-K}
  2242.58,            !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.65,               !- Solar Absorptance
  0.65;               !- Visible Absorptance
MATERIAL:REGULAR,
  A2-100mm DENSE FACE BRICK,  !- Material Name
  Rough,              !- Roughness
  0.100,              !- Thickness {m}
  1.24,               !- Conductivity {w/m-K}
  2082.4,             !- Density {kg/m3}
  920.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.93,               !- Solar Absorptance
  0.93;               !- Visible Absorptance
MATERIAL:REGULAR,
  B3-50mm INSULATION,  !- Material Name
  VeryRough,          !- Roughness
  0.050,              !- Thickness {m}
  0.039999999,        !- Conductivity {w/m-K}
  32.03,              !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
  0.9,                !- Thermal Emittance
  0.5,                !- Solar Absorptance
  0.5;                !- Visible Absorptance
MATERIAL:REGULAR,
  C8-200mm HW CONCRETE BLOCK,  !- Material Name
  Rough,              !- Roughness
  0.200,              !- Thickness {m}
  1.03,               !- Conductivity {w/m-K}
  977.12,             !- Density {kg/m3}
  830.0,              !- Specific Heat {J/kg-K}
```

```
  0.9,                 !- Thermal Emittance
  0.65,                !- Solar Absorptance
  0.65;                !- Visible Absorptance
MATERIAL:WINDOWGLASS,
  GLASS-CLEAR-SHEET-3mm,  !- Material Name
  SpectralAverage,     !- Optical Data Type
  ,                    !- Name of Window Glass Spectral Data Set
  0.003,               !- Thickness {m}
  0.837,               !- Solar Transmittance at Normal Incidence
  0.075,               !- Solar Reflectance at Normal Incidence: Front
  0.075,               !- Solar Reflectance at Normal Incidence: Back
  0.898,               !- Visible Transmittance at Normal Incidence
  0.081,               !- Visible Reflectance at Normal Incidence: Front
  0.081,               !- Visible Reflectance at Normal Incidence: Back
  0.0,                 !- Ir Transmittance at Normal Incidence
  0.84,                !- Ir Emittance at Normal Incidence: Front
  0.84,                !- Ir Emittance at Normal Incidence: Back
  0.9;                 !- Conductivity {W/m-K}
```

IDF specifications for construction

```
CONSTRUCTION,
  CEILINGOffice,  !- Regular Office ceiling
  C5-100mm HW CONCRETE,
  E4-CEILING AIRSPACE,
  E5-ACOUSTIC TILE;
CONSTRUCTION,
  CEILINGHouse,  !- Regular House ceiling
  B10-50mm WOOD,
  E4-CEILING AIRSPACE,
  E5-ACOUSTIC TILE;
CONSTRUCTION,
  FLOORHouse,  !- House floor having contact with the ground
  B11-75mm WOOD;
CONSTRUCTION,
  FLOORHouseIntermediate, !- Intermediate Floor in a house
  E5-ACOUSTIC TILE,
  E4-CEILING AIRSPACE,
  B10-50mm WOOD;
CONSTRUCTION,
  FLOOROffice,  !- Office floor having contact with the ground
  C7-200mm LW CONCRETE BLOCK;
CONSTRUCTION,
  FLOOROfficeIntermediate,  !- Intermediate Office floor
  E5-ACOUSTIC TILE,
  E4-CEILING AIRSPACE,
  C5-100mm HW CONCRETE;
CONSTRUCTION,
  DOORInterior,  !- Hollow wood door
  WOOD-3mm HARDWOOD,
  B1-AIRSPACE RESISTANCE,
```

```
  WOOD-3mm HARDWOOD;
CONSTRUCTION,
  DOORExterior,  !- Solid wood door
  B10-50mm WOOD;
CONSTRUCTION,
  WALLInteriorStructural,  !- Interior Structural Wall
  C7-200mm LW CONCRETE BLOCK;
CONSTRUCTION,
  WALLInteriorPartitionConcBlock,  !- Lightweight Concrete interior wall
  E1-19mm PLASTER OR GYP BOARD,
  C2-100mm LW CONCRETE BLOCK,
  E1-19mm PLASTER OR GYP BOARD;
CONSTRUCTION,
  WALLInteriorPartitionWood,  !- Wood part. wall + Airspace Resistance
  B7-25mm WOOD,
  B1-AIRSPACE RESISTANCE,
  B7-25mm WOOD;
CONSTRUCTION,
  ROOFHouse,  !- Roof construction of house
  E2-50mm SLAG OR STONE,
  E3-9mm FELT AND MEMBRANE,
  B6-50mm DENSE INSULATION,
  B9-100mm WOOD,
  E4-CEILING AIRSPACE,
  E5-ACOUSTIC TILE;
CONSTRUCTION,
  ROOFOffice,  !- Material layer names follow:
  E2-50mm SLAG OR STONE,
  E3-9mm FELT AND MEMBRANE,
  B5-25mm DENSE INSULATION,
  C13-150mm HW CONCRETE,
  E4-CEILING AIRSPACE,
  E5-ACOUSTIC TILE;
CONSTRUCTION,
  WALLHouseExternal,  !- External wall of house with 50mm insulation
  A2-100mm DENSE FACE BRICK,
  B1-AIRSPACE RESISTANCE,
  B3-50mm INSULATION,
  C2-100mm LW CONCRETE BLOCK,
  E1-19mm PLASTER OR GYP BOARD;
CONSTRUCTION,
  WALLOfficeExternal,  !- External wall of office with 50mm insulation
  A2-100mm DENSE FACE BRICK,
  B1-AIRSPACE RESISTANCE,
  B3-50mm INSULATION,
  C8-200mm HW CONCRETE BLOCK,
  E1-19mm PLASTER OR GYP BOARD;
CONSTRUCTION,
  WINDOWDoubleGlazed,  !- Double glazed window with airspace resistance
  GLASS-CLEAR-SHEET-3mm,
  WinAirB1-AIRSPACE RESISTANCE,
```

```
GLASS-CLEAR-SHEET-3mm;
```

Appendix B
Operating schedules used in simulations

Two operating schedules are used in simulations in this thesis: regular office occupancy (09:00-17:00) and extended office occupancy (09:00-22:00). Schedules in EnergyPlus IDF formats are given in the following sections.

Regular schedule

```
ScheduleType,
  Any Number;          !- ScheduleType Name
ScheduleType,
  Fraction,            !- ScheduleType Name
  0.0 : 1.0,           !- range
  CONTINUOUS;          !- Numeric Type
ScheduleType,
  Temperature,         !- ScheduleType Name
  -60:200,             !- range
  CONTINUOUS;          !- Numeric Type
ScheduleType,
  Control Type,        !- ScheduleType Name
  0:4,                 !- range
  DISCRETE;            !- Numeric Type
DAYSCHEDULE,
  ActLevDay,           !- Name
  Any Number,          !- ScheduleType
  131.8,               !- Hour 1
  131.8,               !- Hour 2
  131.8,               !- Hour 3
  131.8,               !- Hour 4
  131.8,               !- Hour 5
  131.8,               !- Hour 6
  131.8,               !- Hour 7
  131.8,               !- Hour 8
  131.8,               !- Hour 9
  131.8,               !- Hour 10
  131.8,               !- Hour 11
  131.8,               !- Hour 12
  131.8,               !- Hour 13
  131.8,               !- Hour 14
  131.8,               !- Hour 15
  131.8,               !- Hour 16
  131.8,               !- Hour 17
```

```
  131.8,             !- Hour 18
  131.8,             !- Hour 19
  131.8,             !- Hour 20
  131.8,             !- Hour 21
  131.8,             !- Hour 22
  131.8,             !- Hour 23
  131.8;             !- Hour 24
WEEKSCHEDULE,
  ActLevWeek,        !- Name
  ActLevDay,         !- Sunday DAYSCHEDULE Name
  ActLevDay,         !- Monday DAYSCHEDULE Name
  ActLevDay,         !- Tuesday DAYSCHEDULE Name
  ActLevDay,         !- Wednesday DAYSCHEDULE Name
  ActLevDay,         !- Thursday DAYSCHEDULE Name
  ActLevDay,         !- Friday DAYSCHEDULE Name
  ActLevDay,         !- Saturday DAYSCHEDULE Name
  ActLevDay,         !- Holiday DAYSCHEDULE Name
  ActLevDay,         !- SummerDesignDay DAYSCHEDULE Name
  ActLevDay,         !- WinterDesignDay DAYSCHEDULE Name
  ActLevDay,         !- CustomDay1 DAYSCHEDULE Name
  ActLevDay;         !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Activity Sch,      !- Name
  Any Number,        !- ScheduleType
  ActLevWeek,        !- Name of WEEKSCHEDULE 1
  1,                 !- Start Month 1
  1,                 !- Start Day 1
  12,                !- End Month 1
  31;                !- End Day 1
DAYSCHEDULE,
  WorkEffDay,        !- Name
  Any Number,        !- ScheduleType
  0.0,               !- Hour 1
  0.0,               !- Hour 2
  0.0,               !- Hour 3
  0.0,               !- Hour 4
  0.0,               !- Hour 5
  0.0,               !- Hour 6
  0.0,               !- Hour 7
  0.0,               !- Hour 8
  0.0,               !- Hour 9
  0.0,               !- Hour 10
  0.0,               !- Hour 11
  0.0,               !- Hour 12
  0.0,               !- Hour 13
  0.0,               !- Hour 14
  0.0,               !- Hour 15
  0.0,               !- Hour 16
  0.0,               !- Hour 17
  0.0,               !- Hour 18
  0.0,               !- Hour 19
```

```
  0.0,                 !- Hour 20
  0.0,                 !- Hour 21
  0.0,                 !- Hour 22
  0.0,                 !- Hour 23
  0.0;                 !- Hour 24
WEEKSCHEDULE,
  WorkEffWeek,         !- Name
  WorkEffDay,          !- Sunday DAYSCHEDULE Name
  WorkEffDay,          !- Monday DAYSCHEDULE Name
  WorkEffDay,          !- Tuesday DAYSCHEDULE Name
  WorkEffDay,          !- Wednesday DAYSCHEDULE Name
  WorkEffDay,          !- Thursday DAYSCHEDULE Name
  WorkEffDay,          !- Friday DAYSCHEDULE Name
  WorkEffDay,          !- Saturday DAYSCHEDULE Name
  WorkEffDay,          !- Holiday DAYSCHEDULE Name
  WorkEffDay,          !- SummerDesignDay DAYSCHEDULE Name
  WorkEffDay,          !- WinterDesignDay DAYSCHEDULE Name
  WorkEffDay,          !- CustomDay1 DAYSCHEDULE Name
  WorkEffDay;          !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Work Eff Sch,        !- Name
  Any Number,          !- ScheduleType
  WorkEffWeek,         !- Name of WEEKSCHEDULE 1
  1,                   !- Start Month 1
  1,                   !- Start Day 1
  12,                  !- End Month 1
  31;                  !- End Day 1
DAYSCHEDULE,
  CloInsDay,           !- Name
  Any Number,          !- ScheduleType
  1.0,                 !- Hour 1
  1.0,                 !- Hour 2
  1.0,                 !- Hour 3
  1.0,                 !- Hour 4
  1.0,                 !- Hour 5
  1.0,                 !- Hour 6
  1.0,                 !- Hour 7
  1.0,                 !- Hour 8
  1.0,                 !- Hour 9
  1.0,                 !- Hour 10
  1.0,                 !- Hour 11
  1.0,                 !- Hour 12
  1.0,                 !- Hour 13
  1.0,                 !- Hour 14
  1.0,                 !- Hour 15
  1.0,                 !- Hour 16
  1.0,                 !- Hour 17
  1.0,                 !- Hour 18
  1.0,                 !- Hour 19
  1.0,                 !- Hour 20
  1.0,                 !- Hour 21
```

```
  1.0,                !- Hour 22
  1.0,                !- Hour 23
  1.0;                !- Hour 24
WEEKSCHEDULE,
  CloInsWeek,         !- Name
  CloInsDay,          !- Sunday DAYSCHEDULE Name
  CloInsDay,          !- Monday DAYSCHEDULE Name
  CloInsDay,          !- Tuesday DAYSCHEDULE Name
  CloInsDay,          !- Wednesday DAYSCHEDULE Name
  CloInsDay,          !- Thursday DAYSCHEDULE Name
  CloInsDay,          !- Friday DAYSCHEDULE Name
  CloInsDay,          !- Saturday DAYSCHEDULE Name
  CloInsDay,          !- Holiday DAYSCHEDULE Name
  CloInsDay,          !- SummerDesignDay DAYSCHEDULE Name
  CloInsDay,          !- WinterDesignDay DAYSCHEDULE Name
  CloInsDay,          !- CustomDay1 DAYSCHEDULE Name
  CloInsDay;          !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Clothing Sch,       !- Name
  Any Number,         !- ScheduleType
  CloInsWeek,         !- Name of WEEKSCHEDULE 1
  1,                  !- Start Month 1
  1,                  !- Start Day 1
  12,                 !- End Month 1
  31;                 !- End Day 1
DAYSCHEDULE,
  AirVelDay,          !- Name
  Any Number,         !- ScheduleType
  0.137,              !- Hour 1
  0.137,              !- Hour 2
  0.137,              !- Hour 3
  0.137,              !- Hour 4
  0.137,              !- Hour 5
  0.137,              !- Hour 6
  0.137,              !- Hour 7
  0.137,              !- Hour 8
  0.137,              !- Hour 9
  0.137,              !- Hour 10
  0.137,              !- Hour 11
  0.137,              !- Hour 12
  0.137,              !- Hour 13
  0.137,              !- Hour 14
  0.137,              !- Hour 15
  0.137,              !- Hour 16
  0.137,              !- Hour 17
  0.137,              !- Hour 18
  0.137,              !- Hour 19
  0.137,              !- Hour 20
  0.137,              !- Hour 21
  0.137,              !- Hour 22
  0.137,              !- Hour 23
```

```
  0.137;           !- Hour 24
WEEKSCHEDULE,
  AirVelWeek,      !- Name
  AirVelDay,       !- Sunday DAYSCHEDULE Name
  AirVelDay,       !- Monday DAYSCHEDULE Name
  AirVelDay,       !- Tuesday DAYSCHEDULE Name
  AirVelDay,       !- Wednesday DAYSCHEDULE Name
  AirVelDay,       !- Thursday DAYSCHEDULE Name
  AirVelDay,       !- Friday DAYSCHEDULE Name
  AirVelDay,       !- Saturday DAYSCHEDULE Name
  AirVelDay,       !- Holiday DAYSCHEDULE Name
  AirVelDay,       !- SummerDesignDay DAYSCHEDULE Name
  AirVelDay,       !- WinterDesignDay DAYSCHEDULE Name
  AirVelDay,       !- CustomDay1 DAYSCHEDULE Name
  AirVelDay;       !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Air Velo Sch,    !- Name
  Any Number,      !- ScheduleType
  AirVelWeek,      !- Name of WEEKSCHEDULE 1
  1,               !- Start Month 1
  1,               !- Start Day 1
  12,              !- End Month 1
  31;              !- End Day 1
DAYSCHEDULE,
  BLDG Day   1,    !- Name
  Any Number,      !- ScheduleType
  0.00,            !- Hour 1
  0.00,            !- Hour 2
  0.00,            !- Hour 3
  0.00,            !- Hour 4
  0.00,            !- Hour 5
  0.00,            !- Hour 6
  0.10,            !- Hour 7
  0.50,            !- Hour 8
  1.00,            !- Hour 9
  1.00,            !- Hour 10
  1.00,            !- Hour 11
  1.00,            !- Hour 12
  0.50,            !- Hour 13
  1.00,            !- Hour 14
  1.00,            !- Hour 15
  1.00,            !- Hour 16
  0.50,            !- Hour 17
  0.10,            !- Hour 18
  0.00,            !- Hour 19
  0.00,            !- Hour 20
  0.00,            !- Hour 21
  0.00,            !- Hour 22
  0.00,            !- Hour 23
  0.00;            !- Hour 24
DAYSCHEDULE,
```

```
  BLDG Day   2,      !- Name
  Any Number,        !- ScheduleType
  0.00,              !- Hour 1
  0.00,              !- Hour 2
  0.00,              !- Hour 3
  0.00,              !- Hour 4
  0.00,              !- Hour 5
  0.00,              !- Hour 6
  0.00,              !- Hour 7
  0.00,              !- Hour 8
  0.00,              !- Hour 9
  0.00,              !- Hour 10
  0.00,              !- Hour 11
  0.00,              !- Hour 12
  0.00,              !- Hour 13
  0.00,              !- Hour 14
  0.00,              !- Hour 15
  0.00,              !- Hour 16
  0.00,              !- Hour 17
  0.00,              !- Hour 18
  0.00,              !- Hour 19
  0.00,              !- Hour 20
  0.00,              !- Hour 21
  0.00,              !- Hour 22
  0.00,              !- Hour 23
  0.00;              !- Hour 24
DAYSCHEDULE,
  BLDG Day   5,      !- Name
  Any Number,        !- ScheduleType
  0.00,              !- Hour 1
  0.00,              !- Hour 2
  0.00,              !- Hour 3
  0.00,              !- Hour 4
  0.00,              !- Hour 5
  0.00,              !- Hour 6
  0.00,              !- Hour 7
  0.00,              !- Hour 8
  1.00,              !- Hour 9
  1.00,              !- Hour 10
  1.00,              !- Hour 11
  1.00,              !- Hour 12
  1.00,              !- Hour 13
  1.00,              !- Hour 14
  1.00,              !- Hour 15
  1.00,              !- Hour 16
  1.00,              !- Hour 17
  1.00,              !- Hour 18
  0.00,              !- Hour 19
  0.00,              !- Hour 20
  0.00,              !- Hour 21
  0.00,              !- Hour 22
```

```
  0.00,              !- Hour 23
  0.00;              !- Hour 24
DAYSCHEDULE,
  BLDG Day   6,      !- Name
  Any Number,        !- ScheduleType
  0.05,              !- Hour 1
  0.05,              !- Hour 2
  0.05,              !- Hour 3
  0.05,              !- Hour 4
  0.05,              !- Hour 5
  0.05,              !- Hour 6
  0.20,              !- Hour 7
  1.00,              !- Hour 8
  1.00,              !- Hour 9
  1.00,              !- Hour 10
  1.00,              !- Hour 11
  1.00,              !- Hour 12
  1.00,              !- Hour 13
  1.00,              !- Hour 14
  1.00,              !- Hour 15
  1.00,              !- Hour 16
  1.00,              !- Hour 17
  0.50,              !- Hour 18
  0.05,              !- Hour 19
  0.05,              !- Hour 20
  0.05,              !- Hour 21
  0.05,              !- Hour 22
  0.05,              !- Hour 23
  0.05;              !- Hour 24
DAYSCHEDULE,
  BLDG Day   7,      !- Name
  Any Number,        !- ScheduleType
  0.05,              !- Hour 1
  0.05,              !- Hour 2
  0.05,              !- Hour 3
  0.05,              !- Hour 4
  0.05,              !- Hour 5
  0.05,              !- Hour 6
  0.05,              !- Hour 7
  0.05,              !- Hour 8
  0.05,              !- Hour 9
  0.05,              !- Hour 10
  0.05,              !- Hour 11
  0.05,              !- Hour 12
  0.05,              !- Hour 13
  0.05,              !- Hour 14
  0.05,              !- Hour 15
  0.05,              !- Hour 16
  0.05,              !- Hour 17
  0.05,              !- Hour 18
  0.05,              !- Hour 19
```

```
  0.05,             !- Hour 20
  0.05,             !- Hour 21
  0.05,             !- Hour 22
  0.05,             !- Hour 23
  0.05;             !- Hour 24
WEEKSCHEDULE,
  OFFICE OCCUPANCY, !- Name
  BLDG Day   2,     !- Sunday DAYSCHEDULE Name
  BLDG Day   1,     !- Monday DAYSCHEDULE Name
  BLDG Day   1,     !- Tuesday DAYSCHEDULE Name
  BLDG Day   1,     !- Wednesday DAYSCHEDULE Name
  BLDG Day   1,     !- Thursday DAYSCHEDULE Name
  BLDG Day   1,     !- Friday DAYSCHEDULE Name
  BLDG Day   2,     !- Saturday DAYSCHEDULE Name
  BLDG Day   2,     !- Holiday DAYSCHEDULE Name
  BLDG Day   2,     !- SummerDesignDay DAYSCHEDULE Name
  BLDG Day   2,     !- WinterDesignDay DAYSCHEDULE Name
  BLDG Day   2,     !- CustomDay1 DAYSCHEDULE Name
  BLDG Day   2;     !- CustomDay2 DAYSCHEDULE Name
WEEKSCHEDULE,
  INTERMITTENT,     !- Name
  BLDG Day   2,     !- Sunday DAYSCHEDULE Name
  BLDG Day   5,     !- Monday DAYSCHEDULE Name
  BLDG Day   5,     !- Tuesday DAYSCHEDULE Name
  BLDG Day   5,     !- Wednesday DAYSCHEDULE Name
  BLDG Day   5,     !- Thursday DAYSCHEDULE Name
  BLDG Day   5,     !- Friday DAYSCHEDULE Name
  BLDG Day   2,     !- Saturday DAYSCHEDULE Name
  BLDG Day   2,     !- Holiday DAYSCHEDULE Name
  BLDG Day   2,     !- SummerDesignDay DAYSCHEDULE Name
  BLDG Day   2,     !- WinterDesignDay DAYSCHEDULE Name
  BLDG Day   2,     !- CustomDay1 DAYSCHEDULE Name
  BLDG Day   2;     !- CustomDay2 DAYSCHEDULE Name
WEEKSCHEDULE,
  OFFICE LIGHTING,  !- Name
  BLDG Day   7,     !- Sunday DAYSCHEDULE Name
  BLDG Day   6,     !- Monday DAYSCHEDULE Name
  BLDG Day   6,     !- Tuesday DAYSCHEDULE Name
  BLDG Day   6,     !- Wednesday DAYSCHEDULE Name
  BLDG Day   6,     !- Thursday DAYSCHEDULE Name
  BLDG Day   6,     !- Friday DAYSCHEDULE Name
  BLDG Day   7,     !- Saturday DAYSCHEDULE Name
  BLDG Day   7,     !- Holiday DAYSCHEDULE Name
  BLDG Day   7,     !- SummerDesignDay DAYSCHEDULE Name
  BLDG Day   7,     !- WinterDesignDay DAYSCHEDULE Name
  BLDG Day   7,     !- CustomDay1 DAYSCHEDULE Name
  BLDG Day   7;     !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  BLDG Sch   1,     !- Name
  Any Number,       !- ScheduleType
  OFFICE OCCUPANCY, !- Name of WEEKSCHEDULE 1
```

```
  1,                   !- Start Month 1
  1,                   !- Start Day 1
  12,                  !- End Month 1
  31;                  !- End Day 1
SCHEDULE,
  BLDG Sch   2,        !- Name
  Any Number,          !- ScheduleType
  INTERMITTENT,        !- Name of WEEKSCHEDULE 1
  1,                   !- Start Month 1
  1,                   !- Start Day 1
  12,                  !- End Month 1
  31;                  !- End Day 1
SCHEDULE,
  BLDG Sch   3,        !- Name
  Any Number,          !- ScheduleType
  OFFICE LIGHTING,     !- Name of WEEKSCHEDULE 1
  1,                   !- Start Month 1
  1,                   !- Start Day 1
  12,                  !- End Month 1
  31;                  !- End Day 1
DAYSCHEDULE,
  Day On Peak,         !- Name
  Fraction,            !- ScheduleType
  0.0,                 !- Hour 1
  0.0,                 !- Hour 2
  0.0,                 !- Hour 3
  0.0,                 !- Hour 4
  0.0,                 !- Hour 5
  0.0,                 !- Hour 6
  0.0,                 !- Hour 7
  0.0,                 !- Hour 8
  0.0,                 !- Hour 9
  1.0,                 !- Hour 10
  1.0,                 !- Hour 11
  1.0,                 !- Hour 12
  1.0,                 !- Hour 13
  1.0,                 !- Hour 14
  1.0,                 !- Hour 15
  1.0,                 !- Hour 16
  1.0,                 !- Hour 17
  1.0,                 !- Hour 18
  0.0,                 !- Hour 19
  0.0,                 !- Hour 20
  0.0,                 !- Hour 21
  0.0,                 !- Hour 22
  0.0,                 !- Hour 23
  0.0;                 !- Hour 24
WEEKSCHEDULE,
  Week on Peak,        !- Name
  Day On Peak,         !- Sunday DAYSCHEDULE Name
  Day On Peak,         !- Monday DAYSCHEDULE Name
```

```
  Day On Peak,        !- Tuesday DAYSCHEDULE Name
  Day On Peak,        !- Wednesday DAYSCHEDULE Name
  Day On Peak,        !- Thursday DAYSCHEDULE Name
  Day On Peak,        !- Friday DAYSCHEDULE Name
  Day On Peak,        !- Saturday DAYSCHEDULE Name
  Day On Peak,        !- Holiday DAYSCHEDULE Name
  Day On Peak,        !- SummerDesignDay DAYSCHEDULE Name
  Day On Peak,        !- WinterDesignDay DAYSCHEDULE Name
  Day On Peak,        !- CustomDay1 DAYSCHEDULE Name
  Day On Peak;        !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  On Peak,            !- Name
  Fraction,           !- ScheduleType
  Week On Peak,       !- Name of WEEKSCHEDULE 1
  1,                  !- Start Month 1
  1,                  !- Start Day 1
  12,                 !- End Month 1
  31;                 !- End Day 1
DAYSCHEDULE,
  Day Off Peak,       !- Name
  Fraction,           !- ScheduleType
  1.0,                !- Hour 1
  1.0,                !- Hour 2
  1.0,                !- Hour 3
  1.0,                !- Hour 4
  1.0,                !- Hour 5
  1.0,                !- Hour 6
  1.0,                !- Hour 7
  1.0,                !- Hour 8
  1.0,                !- Hour 9
  0.0,                !- Hour 10
  0.0,                !- Hour 11
  0.0,                !- Hour 12
  0.0,                !- Hour 13
  0.0,                !- Hour 14
  0.0,                !- Hour 15
  0.0,                !- Hour 16
  0.0,                !- Hour 17
  0.0,                !- Hour 18
  1.0,                !- Hour 19
  1.0,                !- Hour 20
  1.0,                !- Hour 21
  1.0,                !- Hour 22
  1.0,                !- Hour 23
  1.0;                !- Hour 24
WEEKSCHEDULE,
  Week Off Peak,      !- Name
  Day Off Peak,       !- Sunday DAYSCHEDULE Name
  Day Off Peak,       !- Monday DAYSCHEDULE Name
  Day Off Peak,       !- Tuesday DAYSCHEDULE Name
  Day Off Peak,       !- Wednesday DAYSCHEDULE Name
```

```
  Day Off Peak,      !- Thursday DAYSCHEDULE Name
  Day Off Peak,      !- Friday DAYSCHEDULE Name
  Day Off Peak,      !- Saturday DAYSCHEDULE Name
  Day Off Peak,      !- Holiday DAYSCHEDULE Name
  Day Off Peak,      !- SummerDesignDay DAYSCHEDULE Name
  Day Off Peak,      !- WinterDesignDay DAYSCHEDULE Name
  Day Off Peak,      !- CustomDay1 DAYSCHEDULE Name
  Day Off Peak;      !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Off Peak,          !- Name
  Fraction,          !- ScheduleType
  Week Off Peak,     !- Name of WEEKSCHEDULE 1
  1,                 !- Start Month 1
  1,                 !- Start Day 1
  12,                !- End Month 1
  31;                !- End Day 1
DAYSCHEDULE,
  DayON,             !- Name
  Fraction,          !- ScheduleType
  1.0,               !- Hour 1
  1.0,               !- Hour 2
  1.0,               !- Hour 3
  1.0,               !- Hour 4
  1.0,               !- Hour 5
  1.0,               !- Hour 6
  1.0,               !- Hour 7
  1.0,               !- Hour 8
  1.0,               !- Hour 9
  1.0,               !- Hour 10
  1.0,               !- Hour 11
  1.0,               !- Hour 12
  1.0,               !- Hour 13
  1.0,               !- Hour 14
  1.0,               !- Hour 15
  1.0,               !- Hour 16
  1.0,               !- Hour 17
  1.0,               !- Hour 18
  1.0,               !- Hour 19
  1.0,               !- Hour 20
  1.0,               !- Hour 21
  1.0,               !- Hour 22
  1.0,               !- Hour 23
  1.0;               !- Hour 24
WEEKSCHEDULE,
  WeekON,            !- Name
  DayON,             !- Sunday DAYSCHEDULE Name
  DayON,             !- Monday DAYSCHEDULE Name
  DayON,             !- Tuesday DAYSCHEDULE Name
  DayON,             !- Wednesday DAYSCHEDULE Name
  DayON,             !- Thursday DAYSCHEDULE Name
  DayON,             !- Friday DAYSCHEDULE Name
```

```
  DayON,               !- Saturday DAYSCHEDULE Name
  DayON,               !- Holiday DAYSCHEDULE Name
  DayON,               !- SummerDesignDay DAYSCHEDULE Name
  DayON,               !- WinterDesignDay DAYSCHEDULE Name
  DayON,               !- CustomDay1 DAYSCHEDULE Name
  DayON;               !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  ON,                  !- Name
  Fraction,            !- ScheduleType
  WeekON,              !- Name of WEEKSCHEDULE 1
  1,                   !- Start Month 1
  1,                   !- Start Day 1
  12,                  !- End Month 1
  31;                  !- End Day 1
DAYSCHEDULE,
  Heating Setpoint Day Sch,  !- Name
  Temperature,         !- ScheduleType
  15.0,                !- Hour 1
  15.0,                !- Hour 2
  15.0,                !- Hour 3
  15.0,                !- Hour 4
  15.0,                !- Hour 5
  15.0,                !- Hour 6
  15.0,                !- Hour 7
  20.0,                !- Hour 8
  20.0,                !- Hour 9
  20.0,                !- Hour 10
  20.0,                !- Hour 11
  20.0,                !- Hour 12
  20.0,                !- Hour 13
  20.0,                !- Hour 14
  20.0,                !- Hour 15
  20.0,                !- Hour 16
  20.0,                !- Hour 17
  15.0,                !- Hour 18
  15.0,                !- Hour 19
  15.0,                !- Hour 20
  15.0,                !- Hour 21
  15.0,                !- Hour 22
  15.0,                !- Hour 23
  15.0;                !- Hour 24
WEEKSCHEDULE,
  Heating Setpoint Week Sch, !- Name
  Heating Setpoint Day Sch,  !- Sunday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Monday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Tuesday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Wednesday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Thursday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Friday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Saturday DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- Holiday DAYSCHEDULE Name
```

```
  Heating Setpoint Day Sch,  !- SummerDesignDay DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- WinterDesignDay DAYSCHEDULE Name
  Heating Setpoint Day Sch,  !- CustomDay1 DAYSCHEDULE Name
  Heating Setpoint Day Sch;  !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Heating Setpoints,         !- Name
  Temperature,               !- ScheduleType
  Heating Setpoint Week Sch, !- Name of WEEKSCHEDULE 1
  1,                         !- Start Month 1
  1,                         !- Start Day 1
  12,                        !- End Month 1
  31;                        !- End Day 1
DAYSCHEDULE,
  Cooling Setpoint Day Sch,  !- Name
  Temperature,      !- ScheduleType
  30.0,             !- Hour 1
  30.0,             !- Hour 2
  30.0,             !- Hour 3
  30.0,             !- Hour 4
  30.0,             !- Hour 5
  30.0,             !- Hour 6
  30.0,             !- Hour 7
  23.0,             !- Hour 8
  23.0,             !- Hour 9
  23.0,             !- Hour 10
  23.0,             !- Hour 11
  23.0,             !- Hour 12
  23.0,             !- Hour 13
  23.0,             !- Hour 14
  23.0,             !- Hour 15
  23.0,             !- Hour 16
  23.0,             !- Hour 17
  23.0,             !- Hour 18
  23.0,             !- Hour 19
  23.0,             !- Hour 20
  30.0,             !- Hour 21
  30.0,             !- Hour 22
  30.0,             !- Hour 23
  30.0;             !- Hour 24
WEEKSCHEDULE,
  Cooling Setpoint Week Sch, !- Name
  Cooling Setpoint Day Sch,  !- Sunday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Monday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Tuesday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Wednesday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Thursday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Friday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Saturday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- Holiday DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- SummerDesignDay DAYSCHEDULE Name
  Cooling Setpoint Day Sch,  !- WinterDesignDay DAYSCHEDULE Name
```

```
  Cooling Setpoint Day Sch,  !- CustomDay1 DAYSCHEDULE Name
  Cooling Setpoint Day Sch;  !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Cooling Setpoints,!- Name
  Temperature,      !- ScheduleType
  Cooling Setpoint Week Sch,  !- Name of WEEKSCHEDULE 1
  1,                !- Start Month 1
  1,                !- Start Day 1
  12,               !- End Month 1
  31;               !- End Day 1
DAYSCHEDULE,
  Summer Control Type Day Sch,  !- Name
  Control Type,     !- ScheduleType
  2,                !- Hour 1
  2,                !- Hour 2
  2,                !- Hour 3
  2,                !- Hour 4
  2,                !- Hour 5
  2,                !- Hour 6
  2,                !- Hour 7
  2,                !- Hour 8
  2,                !- Hour 9
  2,                !- Hour 10
  2,                !- Hour 11
  2,                !- Hour 12
  2,                !- Hour 13
  2,                !- Hour 14
  2,                !- Hour 15
  2,                !- Hour 16
  2,                !- Hour 17
  2,                !- Hour 18
  2,                !- Hour 19
  2,                !- Hour 20
  2,                !- Hour 21
  2,                !- Hour 22
  2,                !- Hour 23
  2;                !- Hour 24
DAYSCHEDULE,
  Winter Control Type Day Sch,  !- Name
  Control Type,     !- ScheduleType
  1,                !- Hour 1
  1,                !- Hour 2
  1,                !- Hour 3
  1,                !- Hour 4
  1,                !- Hour 5
  1,                !- Hour 6
  1,                !- Hour 7
  1,                !- Hour 8
  1,                !- Hour 9
  1,                !- Hour 10
  1,                !- Hour 11
```

```
  1,                !- Hour 12
  1,                !- Hour 13
  1,                !- Hour 14
  1,                !- Hour 15
  1,                !- Hour 16
  1,                !- Hour 17
  1,                !- Hour 18
  1,                !- Hour 19
  1,                !- Hour 20
  1,                !- Hour 21
  1,                !- Hour 22
  1,                !- Hour 23
  1;                !- Hour 24
WEEKSCHEDULE,
  Summer Control Type Week Sch, !- Name
  Summer Control Type Day Sch,  !- Sunday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Monday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Tuesday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Wednesday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Thursday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Friday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Saturday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- Holiday DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- SummerDesignDay DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- WinterDesignDay DAYSCHEDULE Name
  Summer Control Type Day Sch,  !- CustomDay1 DAYSCHEDULE Name
  Summer Control Type Day Sch;  !- CustomDay2 DAYSCHEDULE Name
WEEKSCHEDULE,
  Winter Control Type Week Sch, !- Name
  Winter Control Type Day Sch,  !- Sunday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Monday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Tuesday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Wednesday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Thursday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Friday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Saturday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- Holiday DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- SummerDesignDay DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- WinterDesignDay DAYSCHEDULE Name
  Winter Control Type Day Sch,  !- CustomDay1 DAYSCHEDULE Name
  Winter Control Type Day Sch;  !- CustomDay2 DAYSCHEDULE Name
SCHEDULE,
  Zone Control Type Sched,  !- Name
  Control Type,     !- ScheduleType
  Winter Control Type Week Sch,  !- Name of WEEKSCHEDULE 1
  1,                !- Start Month 1
  1,                !- Start Day 1
  3,                !- End Month 1
  31,               !- End Day 1
  Summer Control Type Week Sch,  !- Name of WEEKSCHEDULE 2
  4,                !- Start Month 2
```

```
  1,                !- Start Day 2
  9,                !- End Month 2
  30,               !- End Day 2
  Winter Control Type Week Sch,  !- Name of WEEKSCHEDULE 3
  10,               !- Start Month 3
  1,                !- Start Day 3
  12,               !- End Month 3
  31;               !- End Day 3
```

Extended schedule

Only dayschedules are provided here for extended occupancy as the other parts are same as regular occupancy schedule.

```
DAYSCHEDULE,
  BLDG Day   1,     !- Name
  Any Number,       !- ScheduleType
  0.00,             !- Hour 1
  0.00,             !- Hour 2
  0.00,             !- Hour 3
  0.00,             !- Hour 4
  0.00,             !- Hour 5
  0.00,             !- Hour 6
  0.10,             !- Hour 7
  0.50,             !- Hour 8
  1.00,             !- Hour 9
  1.00,             !- Hour 10
  1.00,             !- Hour 11
  1.00,             !- Hour 12
  0.50,             !- Hour 13
  1.00,             !- Hour 14
  1.00,             !- Hour 15
  1.00,             !- Hour 16
  1.00,             !- Hour 17
  1.00,             !- Hour 18
  1.00,             !- Hour 19
  1.00,             !- Hour 20
  0.50,             !- Hour 21
  0.50,             !- Hour 22
  0.00,             !- Hour 23
  0.00;             !- Hour 24
DAYSCHEDULE,
  BLDG Day   2,     !- Name
  Any Number,       !- ScheduleType
  0.00,             !- Hour 1
  0.00,             !- Hour 2
  0.00,             !- Hour 3
  0.00,             !- Hour 4
  0.00,             !- Hour 5
  0.00,             !- Hour 6
```

```
  0.00,              !- Hour 7
  0.00,              !- Hour 8
  0.00,              !- Hour 9
  0.00,              !- Hour 10
  0.00,              !- Hour 11
  0.00,              !- Hour 12
  0.00,              !- Hour 13
  0.00,              !- Hour 14
  0.00,              !- Hour 15
  0.00,              !- Hour 16
  0.00,              !- Hour 17
  0.00,              !- Hour 18
  0.00,              !- Hour 19
  0.00,              !- Hour 20
  0.00,              !- Hour 21
  0.00,              !- Hour 22
  0.00,              !- Hour 23
  0.00;              !- Hour 24
DAYSCHEDULE,
  BLDG Day   5,      !- Name
  Any Number,        !- ScheduleType
  0.00,              !- Hour 1
  0.00,              !- Hour 2
  0.00,              !- Hour 3
  0.00,              !- Hour 4
  0.00,              !- Hour 5
  0.00,              !- Hour 6
  0.00,              !- Hour 7
  0.00,              !- Hour 8
  1.00,              !- Hour 9
  1.00,              !- Hour 10
  1.00,              !- Hour 11
  1.00,              !- Hour 12
  1.00,              !- Hour 13
  1.00,              !- Hour 14
  1.00,              !- Hour 15
  1.00,              !- Hour 16
  1.00,              !- Hour 17
  1.00,              !- Hour 18
  1.00,              !- Hour 19
  1.00,              !- Hour 20
  1.00,              !- Hour 21
  1.00,              !- Hour 22
  0.00,              !- Hour 23
  0.00;              !- Hour 24
DAYSCHEDULE,
  BLDG Day   6,      !- Name
  Any Number,        !- ScheduleType
  0.05,              !- Hour 1
  0.05,              !- Hour 2
  0.05,              !- Hour 3
```

```
  0.05,             !- Hour 4
  0.05,             !- Hour 5
  0.05,             !- Hour 6
  0.20,             !- Hour 7
  1.00,             !- Hour 8
  1.00,             !- Hour 9
  1.00,             !- Hour 10
  1.00,             !- Hour 11
  1.00,             !- Hour 12
  1.00,             !- Hour 13
  1.00,             !- Hour 14
  1.00,             !- Hour 15
  1.00,             !- Hour 16
  1.00,             !- Hour 17
  0.50,             !- Hour 18
  0.05,             !- Hour 19
  0.05,             !- Hour 20
  0.05,             !- Hour 21
  0.05,             !- Hour 22
  0.05,             !- Hour 23
  0.05;             !- Hour 24
DAYSCHEDULE,
  BLDG Day   7,     !- Name
  Any Number,       !- ScheduleType
  0.05,             !- Hour 1
  0.05,             !- Hour 2
  0.05,             !- Hour 3
  0.05,             !- Hour 4
  0.05,             !- Hour 5
  0.05,             !- Hour 6
  0.05,             !- Hour 7
  0.05,             !- Hour 8
  0.05,             !- Hour 9
  0.05,             !- Hour 10
  0.05,             !- Hour 11
  0.05,             !- Hour 12
  0.05,             !- Hour 13
  0.05,             !- Hour 14
  0.05,             !- Hour 15
  0.05,             !- Hour 16
  0.05,             !- Hour 17
  0.05,             !- Hour 18
  0.05,             !- Hour 19
  0.05,             !- Hour 20
  0.05,             !- Hour 21
  0.05,             !- Hour 22
  0.05,             !- Hour 23
  0.05;             !- Hour 24
DAYSCHEDULE,
  Day On Peak,      !- Name
  Fraction,         !- ScheduleType
```

```
  0.0,                !- Hour 1
  0.0,                !- Hour 2
  0.0,                !- Hour 3
  0.0,                !- Hour 4
  0.0,                !- Hour 5
  0.0,                !- Hour 6
  0.0,                !- Hour 7
  0.0,                !- Hour 8
  1.0,                !- Hour 9
  1.0,                !- Hour 10
  1.0,                !- Hour 11
  1.0,                !- Hour 12
  1.0,                !- Hour 13
  1.0,                !- Hour 14
  1.0,                !- Hour 15
  1.0,                !- Hour 16
  1.0,                !- Hour 17
  1.0,                !- Hour 18
  1.0,                !- Hour 19
  1.0,                !- Hour 20
  1.0,                !- Hour 21
  1.0,                !- Hour 22
  0.0,                !- Hour 23
  0.0;                !- Hour 24
DAYSCHEDULE,
  Day Off Peak,       !- Name
  Fraction,           !- ScheduleType
  1.0,                !- Hour 1
  1.0,                !- Hour 2
  1.0,                !- Hour 3
  1.0,                !- Hour 4
  1.0,                !- Hour 5
  1.0,                !- Hour 6
  1.0,                !- Hour 7
  1.0,                !- Hour 8
  1.0,                !- Hour 9
  0.0,                !- Hour 10
  0.0,                !- Hour 11
  0.0,                !- Hour 12
  0.0,                !- Hour 13
  0.0,                !- Hour 14
  0.0,                !- Hour 15
  0.0,                !- Hour 16
  0.0,                !- Hour 17
  0.0,                !- Hour 18
  1.0,                !- Hour 19
  1.0,                !- Hour 20
  1.0,                !- Hour 21
  1.0,                !- Hour 22
  1.0,                !- Hour 23
  1.0;                !- Hour 24
```

```
DAYSCHEDULE,
  DayON,                !- Name
  Fraction,             !- ScheduleType
  1.0,                  !- Hour 1
  1.0,                  !- Hour 2
  1.0,                  !- Hour 3
  1.0,                  !- Hour 4
  1.0,                  !- Hour 5
  1.0,                  !- Hour 6
  1.0,                  !- Hour 7
  1.0,                  !- Hour 8
  1.0,                  !- Hour 9
  1.0,                  !- Hour 10
  1.0,                  !- Hour 11
  1.0,                  !- Hour 12
  1.0,                  !- Hour 13
  1.0,                  !- Hour 14
  1.0,                  !- Hour 15
  1.0,                  !- Hour 16
  1.0,                  !- Hour 17
  1.0,                  !- Hour 18
  1.0,                  !- Hour 19
  1.0,                  !- Hour 20
  1.0,                  !- Hour 21
  1.0,                  !- Hour 22
  1.0,                  !- Hour 23
  1.0;                  !- Hour 24
DAYSCHEDULE,
  Heating Setpoint Day Sch,  !- Name
  Temperature,          !- ScheduleType
  15.0,                 !- Hour 1
  15.0,                 !- Hour 2
  15.0,                 !- Hour 3
  15.0,                 !- Hour 4
  15.0,                 !- Hour 5
  15.0,                 !- Hour 6
  15.0,                 !- Hour 7
  20.0,                 !- Hour 8
  20.0,                 !- Hour 9
  20.0,                 !- Hour 10
  20.0,                 !- Hour 11
  20.0,                 !- Hour 12
  20.0,                 !- Hour 13
  20.0,                 !- Hour 14
  20.0,                 !- Hour 15
  20.0,                 !- Hour 16
  20.0,                 !- Hour 17
  20.0,                 !- Hour 18
  20.0,                 !- Hour 19
  20.0,                 !- Hour 20
  20.0,                 !- Hour 21
```

```
  20.0,              !- Hour 22
  15.0,              !- Hour 23
  15.0;              !- Hour 24
DAYSCHEDULE,
  Cooling Setpoint Day Sch,  !- Name
  Temperature,       !- ScheduleType
  30.0,              !- Hour 1
  30.0,              !- Hour 2
  30.0,              !- Hour 3
  30.0,              !- Hour 4
  30.0,              !- Hour 5
  30.0,              !- Hour 6
  30.0,              !- Hour 7
  23.0,              !- Hour 8
  23.0,              !- Hour 9
  23.0,              !- Hour 10
  23.0,              !- Hour 11
  23.0,              !- Hour 12
  23.0,              !- Hour 13
  23.0,              !- Hour 14
  23.0,              !- Hour 15
  23.0,              !- Hour 16
  23.0,              !- Hour 17
  23.0,              !- Hour 18
  23.0,              !- Hour 19
  23.0,              !- Hour 20
  23.0,              !- Hour 21
  23.0,              !- Hour 22
  30.0,              !- Hour 23
  30.0;              !- Hour 24
```

Appendix C
ardML schema

*The following is the ardML (**ard**ot X**ML** Schema) schema documentation developed as part of this thesis for ArDOT application logic and messaging.*

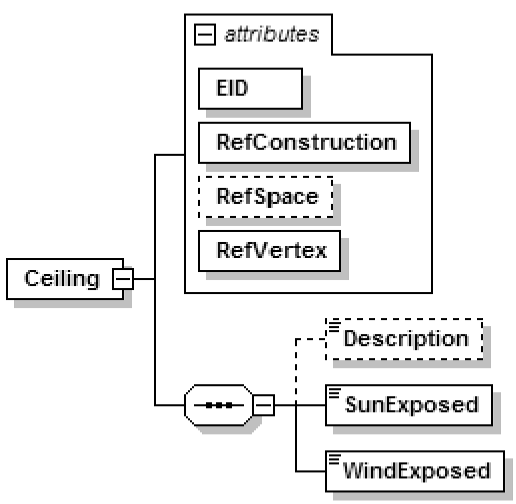
Schema: ardML-0-5.xsd

schema location:	**ardML-0-5.xsd**
attribute form default:	**unqualified**
element form default:	**unqualified**
targetNamespace:	**http://www.mourshed.org/schema/0-05**

Elements
Ceiling
Construction
Door
Floor
Location
Material
MaterialWndGas
MaterialWndGlass
Project
Roof
ShadingAttached
ShadingDetachedFixed
Space
Vertex
Wall
Window

element **Ceiling**

diagram



properties	content	complex

children **Description SunExposed WindExposed**

used by element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefSpace	**xs:IDREF**	optional			
RefVertex	**xs:IDREFS**	required			

source
```
<xs:element name="Ceiling">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="SunExposed" type="xs:boolean" default="1"/>
      <xs:element name="WindExposed" type="xs:boolean" default="1"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefSpace" type="xs:IDREF" use="optional"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Ceiling/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source
```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Ceiling/SunExposed**

diagram 

type **xs:boolean**

properties

isRef	0
content	simple
default	1

source
```
<xs:element name="SunExposed" type="xs:boolean" default="1"/>
```

element **Ceiling/WindExposed**

diagram 

type **xs:boolean**

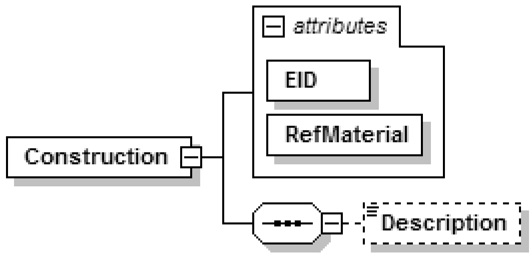
properties

isRef	0
content	simple
default	1

source
```
<xs:element name="WindExposed" type="xs:boolean" default="1"/>
```

element **Construction**

diagram



properties content complex

children **Description**

used by element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefMaterial	**xs:IDREFS**	required			

source

```
<xs:element name="Construction">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefMaterial" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Construction/Description**

diagram

type **xs:string**

properties

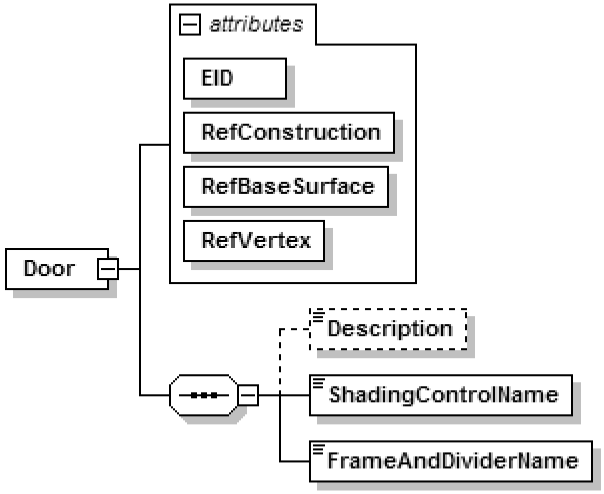
isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Door**

diagram



properties content complex

children **Description ShadingControlName FrameAndDividerName**

used by element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefBaseSurface	**xs:IDREF**	required			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="Door">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="ShadingControlName" type="xs:string" nillable="true"/>
      <xs:element name="FrameAndDividerName" type="xs:string" nillable="true"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefBaseSurface" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Door/Description**

diagram

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Door/ShadingControlName**

diagram

type **xs:string**

properties

isRef	0
content	simple
nillable	true

source

```
<xs:element name="ShadingControlName" type="xs:string" nillable="true"/>
```

element **Door/FrameAndDividerName**

diagram

type **xs:string**

properties

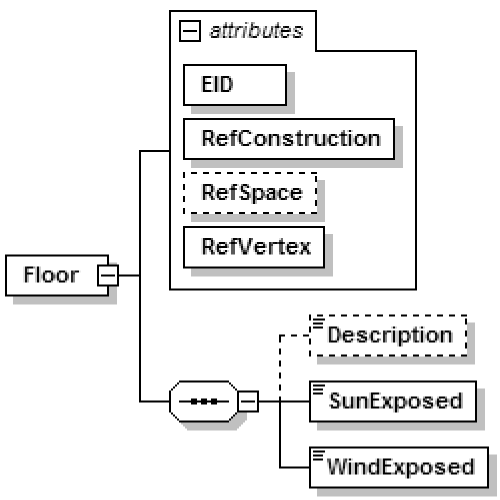
isRef	0
content	simple
nillable	true

source

```
<xs:element name="FrameAndDividerName" type="xs:string" nillable="true"/>
```

element **Floor**

diagram



properties

content	complex

children **Description SunExposed WindExposed**

used by

element	**Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefSpace	**xs:IDREF**	optional			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="Floor">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Description" type="xs:string" minOccurs="0"/>
   <xs:element name="SunExposed" type="xs:boolean" default="1"/>
```

```
      <xs:element name="WindExposed" type="xs:boolean" default="1"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefSpace" type="xs:IDREF" use="optional"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Floor/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source <xs:element name="Description" type="xs:string" minOccurs="0"/>

element **Floor/SunExposed**

diagram 

type **xs:boolean**

properties

isRef	0
content	simple
default	1

source <xs:element name="SunExposed" type="xs:boolean" default="1"/>

element **Floor/WindExposed**

diagram 

type **xs:boolean**

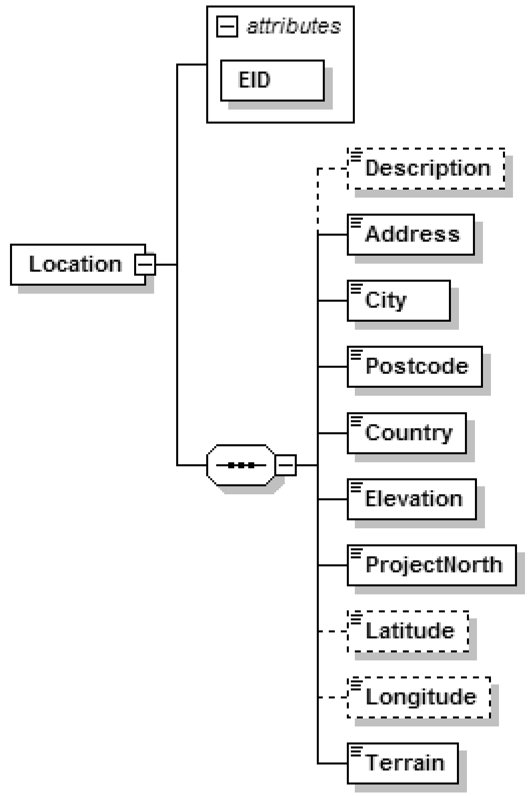
properties

isRef	0
content	simple
default	1

source <xs:element name="WindExposed" type="xs:boolean" default="1"/>

element **Location**

diagram



properties	content	complex
children	**Description Address City Postcode Country Elevation ProjectNorth Latitude Longitude Terrain**	
used by	element	**Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			

source

```
<xs:element name="Location">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Description" type="xs:string" minOccurs="0"/>
   <xs:element name="Address" type="xs:string"/>
   <xs:element name="City" type="xs:string"/>
   <xs:element name="Postcode" type="xs:string"/>
   <xs:element name="Country" type="xs:string"/>
   <xs:element name="Elevation" type="xs:double"/>
   <xs:element name="ProjectNorth" default="0.0">
    <xs:simpleType>
     <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="360.0"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
   <xs:element name="Latitude" minOccurs="0">
    <xs:simpleType>
     <xs:restriction base="xs:double">
```

```
        <xs:minInclusive value="-180.0"/>
        <xs:maxInclusive value="180.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Longitude" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minInclusive value="-180.0"/>
        <xs:maxInclusive value="180.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Terrain" default="City">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="City"/>
        <xs:enumeration value="Suburbs"/>
        <xs:enumeration value="Country"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
 </xs:sequence>
 <xs:attribute name="EID" type="xs:ID" use="required"/>
</xs:complexType>
</xs:element>
```

element **Location/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source <xs:element name="Description" type="xs:string" minOccurs="0"/>

element **Location/Address**

diagram 

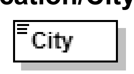
type **xs:string**

properties

isRef	0
content	simple

source <xs:element name="Address" type="xs:string"/>

element **Location/City**

diagram 

type **xs:string**

properties

isRef	0
content	simple

source <xs:element name="City" type="xs:string"/>

element **Location/Postcode**

diagram 

type **xs:string**

properties
isRef 0
content simple

source
```
<xs:element name="Postcode" type="xs:string"/>
```

element **Location/Country**

diagram 

type **xs:string**

properties
isRef 0
content simple

source
```
<xs:element name="Country" type="xs:string"/>
```

element **Location/Elevation**

diagram 

type **xs:double**

properties
isRef 0
content simple

source
```
<xs:element name="Elevation" type="xs:double"/>
```

element **Location/ProjectNorth**

diagram 

type restriction of **xs:double**

properties
isRef 0
content simple
default 0.0

facets
minInclusive 0.0
maxExclusive 360.0

source
```
<xs:element name="ProjectNorth" default="0.0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="360.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Location/Latitude**

diagram 

type restriction of **xs:double**

properties
isRef 0
minOcc 0

	maxOcc	1
	content	simple
facets	minInclusive	-180.0
	maxInclusive	180.0

source

```
<xs:element name="Latitude" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="-180.0"/>
      <xs:maxInclusive value="180.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Location/Longitude**

diagram

type restriction of **xs:double**

properties	isRef	0
	minOcc	0
	maxOcc	1
	content	simple
facets	minInclusive	-180.0
	maxInclusive	180.0

source

```
<xs:element name="Longitude" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="-180.0"/>
      <xs:maxInclusive value="180.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Location/Terrain**

diagram

type restriction of **xs:string**

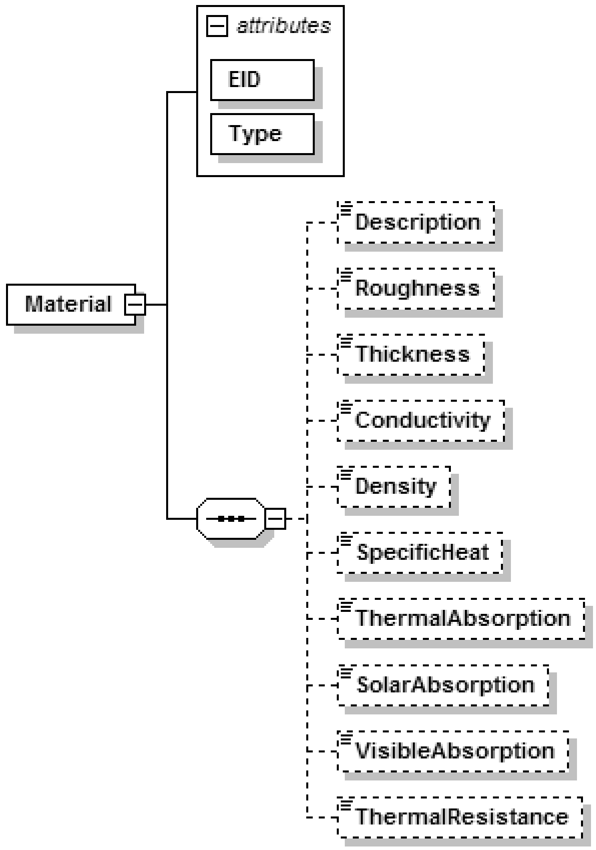
properties	isRef	0
	content	simple
	default	City
facets	enumeration	City
	enumeration	Suburbs
	enumeration	Country

source

```
<xs:element name="Terrain" default="City">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="City"/>
      <xs:enumeration value="Suburbs"/>
      <xs:enumeration value="Country"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material**

diagram



properties content complex

children **Description Roughness Thickness Conductivity Density SpecificHeat ThermalAbsorption SolarAbsorption VisibleAbsorption ThermalResistance**

used by element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
Type	**derived by: xs:string**	required			

source

```
<xs:element name="Material">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Description" type="xs:string" minOccurs="0"/>
   <xs:element name="Roughness" minOccurs="0">
    <xs:simpleType>
     <xs:restriction base="xs:string">
      <xs:enumeration value="VeryRough"/>
      <xs:enumeration value="MediumRough"/>
      <xs:enumeration value="Rough"/>
      <xs:enumeration value="Smooth"/>
      <xs:enumeration value="MediumSmooth"/>
      <xs:enumeration value="VerySmooth"/>
     </xs:restriction>
    </xs:simpleType>
```

```
  </xs:element>
  <xs:element name="Thickness" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minExclusive value="0.0"/>
        <xs:maxInclusive value="3.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Conductivity" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minExclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Density" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minExclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="SpecificHeat" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minExclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="ThermalAbsorption" default="0.9" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:maxInclusive value="0.99999"/>
        <xs:minInclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="SolarAbsorption" default="0.7" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:maxInclusive value="1.0"/>
        <xs:minInclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="VisibleAbsorption" default="0.7" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minInclusive value="0.0"/>
        <xs:maxInclusive value="1.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="ThermalResistance" minOccurs="0">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:minExclusive value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
</xs:sequence>
<xs:attribute name="EID" type="xs:ID" use="required"/>
```

```
    <xs:attribute name="Type" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Regular"/>
          <xs:enumeration value="Regular-R"/>
          <xs:enumeration value="Air"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
```

element **Material/Description**

diagram

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Material/Roughness**

diagram

type restriction of **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

enumeration	VeryRough
enumeration	MediumRough
enumeration	Rough
enumeration	Smooth
enumeration	MediumSmooth
enumeration	VerySmooth

source

```
<xs:element name="Roughness" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="VeryRough"/>
      <xs:enumeration value="MediumRough"/>
      <xs:enumeration value="Rough"/>
      <xs:enumeration value="Smooth"/>
      <xs:enumeration value="MediumSmooth"/>
      <xs:enumeration value="VerySmooth"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/Thickness**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0

maxOcc	1
content	simple

facets

maxInclusive	3.0
minExclusive	0.0

source

```
<xs:element name="Thickness" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
      <xs:maxInclusive value="3.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/Conductivity**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minExclusive	0.0

source

```
<xs:element name="Conductivity" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/Density**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minExclusive	0.0

source

```
<xs:element name="Density" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/SpecificHeat**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0

maxOcc 1
content simple

facets minExclusive 0.0

source
```
<xs:element name="SpecificHeat" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/ThermalAbsorption**

diagram

type restriction of **xs:double**

properties
isRef 0
minOcc 0
maxOcc 1
content simple
default 0.9

facets
minInclusive 0.0
maxInclusive 0.99999

source
```
<xs:element name="ThermalAbsorption" default="0.9" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:maxInclusive value="0.99999"/>
      <xs:minInclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/SolarAbsorption**

diagram

type restriction of **xs:double**

properties
isRef 0
minOcc 0
maxOcc 1
content simple
default 0.7

facets
minInclusive 0.0
maxInclusive 1.0

source
```
<xs:element name="SolarAbsorption" default="0.7" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:maxInclusive value="1.0"/>
      <xs:minInclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/VisibleAbsorption**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple
default	0.7

facets

minInclusive	0.0
maxInclusive	1.0

source

```
<xs:element name="VisibleAbsorption" default="0.7" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxInclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Material/ThermalResistance**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

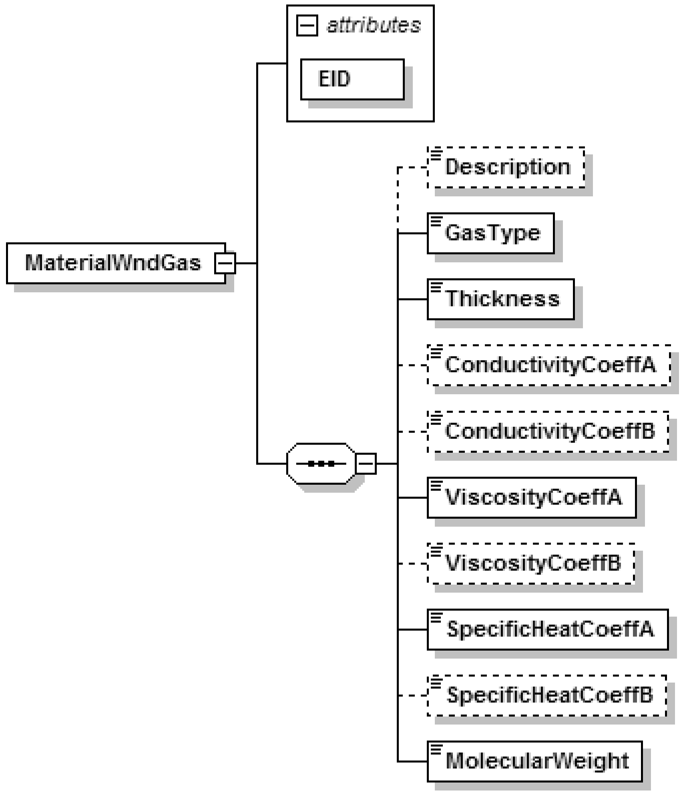
minExclusive	0.0

source

```
<xs:element name="ThermalResistance" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas**

diagram



properties content complex

children **Description GasType Thickness ConductivityCoeffA ConductivityCoeffB ViscosityCoeffA ViscosityCoeffB SpecificHeatCoeffA SpecificHeatCoeffB MolecularWeight**

used by element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			

source

```
<xs:element name="MaterialWndGas">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="GasType">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Air"/>
            <xs:enumeration value="Argon"/>
            <xs:enumeration value="Krypton"/>
            <xs:enumeration value="Xenon"/>
            <xs:enumeration value="Custom"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="Thickness">
        <xs:simpleType>
          <xs:restriction base="xs:double">
            <xs:minExclusive value="0.0"/>
          </xs:restriction>
```

```
      </xs:simpleType>
    </xs:element>
    <xs:element name="ConductivityCoeffA" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:double">
          <xs:minExclusive value="0.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="ConductivityCoeffB" type="xs:double" minOccurs="0"/>
    <xs:element name="ViscosityCoeffA" nillable="true">
      <xs:simpleType>
        <xs:restriction base="xs:double">
          <xs:minExclusive value="0.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="ViscosityCoeffB" type="xs:double" minOccurs="0"/>
    <xs:element name="SpecificHeatCoeffA" nillable="true">
      <xs:simpleType>
        <xs:restriction base="xs:double">
          <xs:minExclusive value="0.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="SpecificHeatCoeffB" type="xs:double" minOccurs="0"/>
    <xs:element name="MolecularWeight" nillable="true">
      <xs:simpleType>
        <xs:restriction base="xs:double">
          <xs:minInclusive value="20.0"/>
          <xs:maxInclusive value="200.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="EID" type="xs:ID" use="required"/>
 </xs:complexType>
</xs:element>
```

element **MaterialWndGas/Description**

diagram

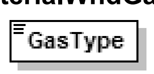
type **xs:string**

properties	isRef	0
	minOcc	0
	maxOcc	1
	content	simple

source <xs:element name="Description" type="xs:string" minOccurs="0"/>

element **MaterialWndGas/GasType**

diagram



type restriction of **xs:string**

properties	isRef	0
	content	simple
facets	enumeration	Air
	enumeration	Argon
	enumeration	Krypton

enumeration	Xenon
enumeration	Custom

source
```
<xs:element name="GasType">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Air"/>
      <xs:enumeration value="Argon"/>
      <xs:enumeration value="Krypton"/>
      <xs:enumeration value="Xenon"/>
      <xs:enumeration value="Custom"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas/Thickness**

diagram

type restriction of **xs:double**

properties

isRef	0
content	simple

facets

minExclusive	0.0

source
```
<xs:element name="Thickness">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas/ConductivityCoeffA**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

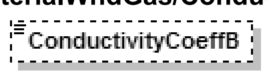
facets

minExclusive	0.0

source
```
<xs:element name="ConductivityCoeffA" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas/ConductivityCoeffB**

diagram



type **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1

content	simple

source
```
<xs:element name="ConductivityCoeffB" type="xs:double" minOccurs="0"/>
```

element **MaterialWndGas/ViscosityCoeffA**

diagram

type restriction of **xs:double**

properties

isRef	0
content	simple
nillable	true

facets

minExclusive	0.0

source
```
<xs:element name="ViscosityCoeffA" nillable="true">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas/ViscosityCoeffB**

diagram

type **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source
```
<xs:element name="ViscosityCoeffB" type="xs:double" minOccurs="0"/>
```

element **MaterialWndGas/SpecificHeatCoeffA**

diagram

type restriction of **xs:double**

properties

isRef	0
content	simple
nillable	true

facets

minExclusive	0.0

source
```
<xs:element name="SpecificHeatCoeffA" nillable="true">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGas/SpecificHeatCoeffB**

diagram

type **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="SpecificHeatCoeffB" type="xs:double" minOccurs="0"/>
```

element **MaterialWndGas/MolecularWeight**

diagram

type restriction of **xs:double**

properties

isRef	0
content	simple
nillable	true

facets

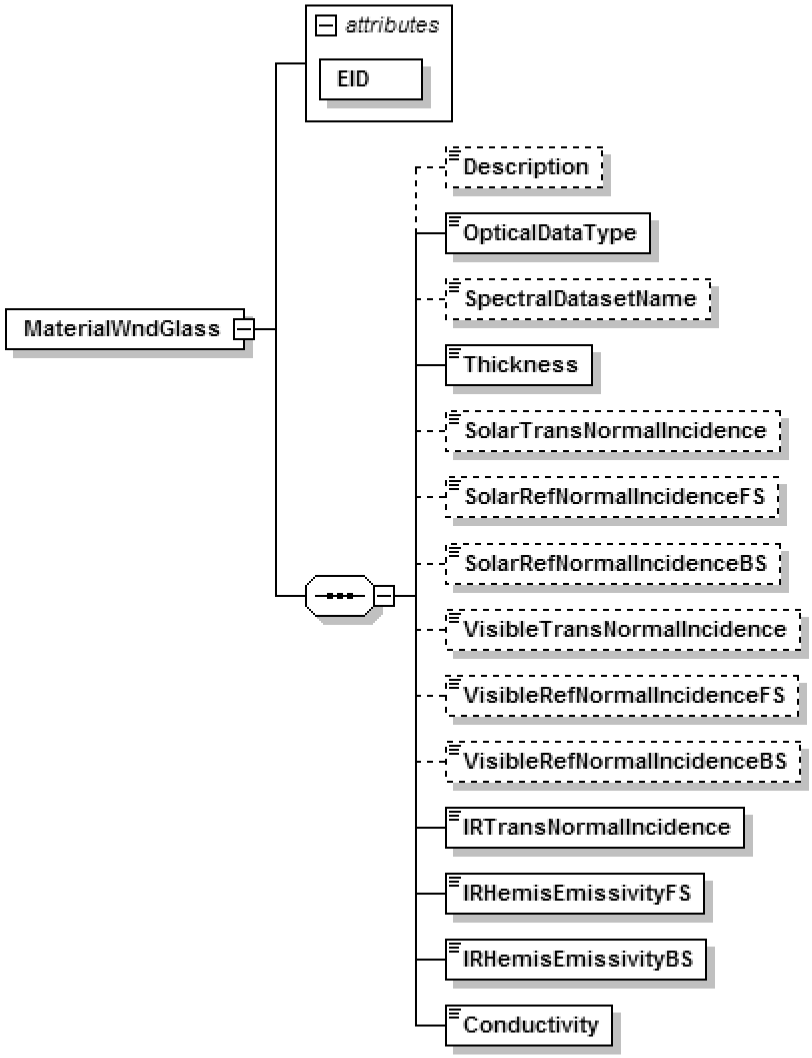
minInclusive	20.0
maxInclusive	200.0

source

```
<xs:element name="MolecularWeight" nillable="true">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="20.0"/>
      <xs:maxInclusive value="200.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass**

diagram



properties — content: complex

children: **Description OpticalDataType SpectralDatasetName Thickness SolarTransNormalIncidence SolarRefNormalIncidenceFS SolarRefNormalIncidenceBS VisibleTransNormalIncidence VisibleRefNormalIncidenceFS VisibleRefNormalIncidenceBS IRTransNormalIncidence IRHemisEmissivityFS IRHemisEmissivityBS Conductivity**

used by — element: **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			

source

```
<xs:element name="MaterialWndGlass">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="OpticalDataType">
```

```
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="SpectralAverage"/>
      <xs:enumeration value="Spectral"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SpectralDatasetName" type="xs:string" minOccurs="0"/>
<xs:element name="Thickness">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SolarTransNormalIncidence" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SolarRefNormalIncidenceFS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SolarRefNormalIncidenceBS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="VisibleTransNormalIncidence" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="VisibleRefNormalIncidenceFS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="VisibleRefNormalIncidenceBS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
```

```
        </xs:element>
        <xs:element name="IRTransNormalIncidence" default="0.0">
          <xs:simpleType>
            <xs:restriction base="xs:double">
              <xs:minInclusive value="0.0"/>
              <xs:maxExclusive value="1.0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="IRHemisEmissivityFS" default="0.84">
          <xs:simpleType>
            <xs:restriction base="xs:double">
              <xs:minExclusive value="0.0"/>
              <xs:maxExclusive value="1.0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="IRHemisEmissivityBS" default="0.84">
          <xs:simpleType>
            <xs:restriction base="xs:double">
              <xs:minExclusive value="0.0"/>
              <xs:maxExclusive value="1.0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="Conductivity" default="0.9">
          <xs:simpleType>
            <xs:restriction base="xs:double">
              <xs:minExclusive value="0.0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="EID" type="xs:ID" use="required"/>
    </xs:complexType>
  </xs:element>
```

element **MaterialWndGlass/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **MaterialWndGlass/OpticalDataType**

diagram 

type restriction of **xs:string**

properties

isRef	0
content	simple

facets

enumeration	SpectralAverage
enumeration	Spectral

source

```
<xs:element name="OpticalDataType">
  <xs:simpleType>
```

```
      <xs:restriction base="xs:string">
        <xs:enumeration value="SpectralAverage"/>
        <xs:enumeration value="Spectral"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
```

element **MaterialWndGlass/SpectralDatasetName**

diagram

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="SpectralDatasetName" type="xs:string" minOccurs="0"/>
```

element **MaterialWndGlass/Thickness**

diagram

type restriction of **xs:double**

properties

isRef	0
content	simple

facets

minExclusive	0.0

source

```
<xs:element name="Thickness">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/SolarTransNormalIncidence**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minInclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="SolarTransNormalIncidence" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/SolarRefNormalIncidenceFS**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minInclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="SolarRefNormalIncidenceFS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/SolarRefNormalIncidenceBS**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minInclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="SolarRefNormalIncidenceBS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/VisibleTransNormalIncidence**

diagram

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

facets

minInclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="VisibleTransNormalIncidence" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
```

```
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/VisibleRefNormalIncidenceFS**

diagram 

type restriction of **xs:double**

properties
isRef 0
minOcc 0
maxOcc 1
content simple

facets
minInclusive 0.0
maxExclusive 1.0

source
```
<xs:element name="VisibleRefNormalIncidenceFS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/VisibleRefNormalIncidenceBS**

diagram 

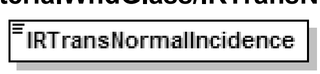
type restriction of **xs:double**

properties
isRef 0
minOcc 0
maxOcc 1
content simple

facets
minInclusive 0.0
maxExclusive 1.0

source
```
<xs:element name="VisibleRefNormalIncidenceBS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/IRTransNormalIncidence**

diagram 

type restriction of **xs:double**

properties
isRef 0
content simple
default 0.0

facets
minInclusive 0.0
maxExclusive 1.0

source
```
<xs:element name="IRTransNormalIncidence" default="0.0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minInclusive value="0.0"/>
```

```
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/IRHemisEmissivityFS**

diagram 

type restriction of **xs:double**

properties

isRef	0
content	simple
default	0.84

facets

minExclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="IRHemisEmissivityFS" default="0.84">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/IRHemisEmissivityBS**

diagram 

type restriction of **xs:double**

properties

isRef	0
content	simple
default	0.84

facets

minExclusive	0.0
maxExclusive	1.0

source

```
<xs:element name="IRHemisEmissivityBS" default="0.84">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
      <xs:maxExclusive value="1.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **MaterialWndGlass/Conductivity**

diagram 

type restriction of **xs:double**

properties

isRef	0
content	simple
default	0.9

facets

minExclusive	0.0

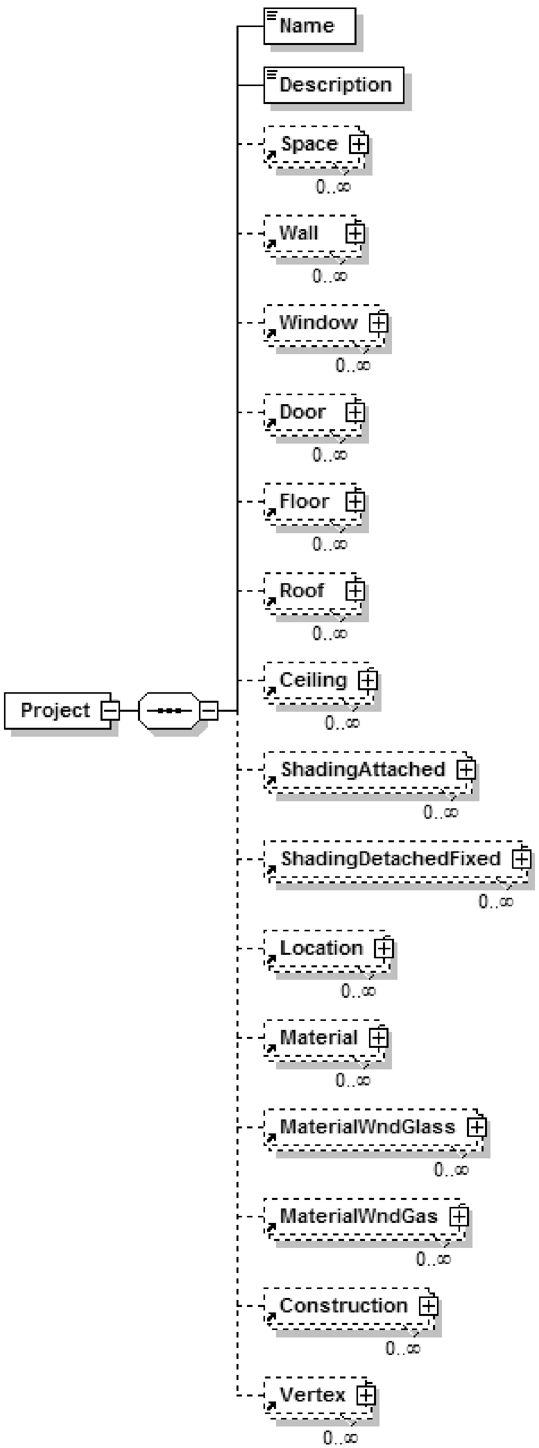
source

```
<xs:element name="Conductivity" default="0.9">
  <xs:simpleType>
```

```
        <xs:restriction base="xs:double">
          <xs:minExclusive value="0.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
```

element **Project**

diagram



properties content complex

children **Name Description Space Wall Window Door Floor Roof Ceiling ShadingAttached ShadingDetachedFixed Location Material MaterialWndGlass MaterialWndGas Construction Vertex**

source

```
<xs:element name="Project">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Name" type="xs:string"/>
      <xs:element name="Description" type="xs:string"/>
      <xs:element ref="Space" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Wall" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Window" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Door" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Floor" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Roof" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Ceiling" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="ShadingAttached" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="ShadingDetachedFixed" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Location" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Material" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="MaterialWndGlass" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="MaterialWndGas" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Construction" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Vertex" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

element **Project/Name**

diagram

type **xs:string**

properties

isRef	0
content	simple

source

```
<xs:element name="Name" type="xs:string"/>
```

element **Project/Description**

diagram

type **xs:string**

properties

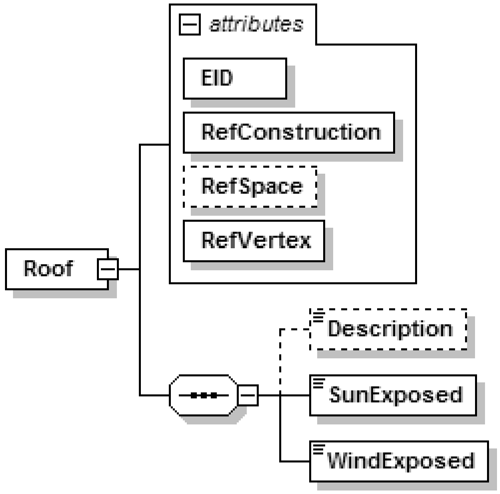
isRef	0
content	simple

source

```
<xs:element name="Description" type="xs:string"/>
```

element **Roof**

diagram



properties — content: complex

children: **Description SunExposed WindExposed**

used by — element: **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefSpace	**xs:IDREF**	optional			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="Roof">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="SunExposed" type="xs:boolean" default="1"/>
      <xs:element name="WindExposed" type="xs:boolean" default="1"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefSpace" type="xs:IDREF" use="optional"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Roof/Description**

diagram

type: **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Roof/SunExposed**

diagram

type **xs:boolean**

properties
isRef 0
content simple
default 1

source

```
<xs:element name="SunExposed" type="xs:boolean" default="1"/>
```

element **Roof/WindExposed**

diagram

type **xs:boolean**

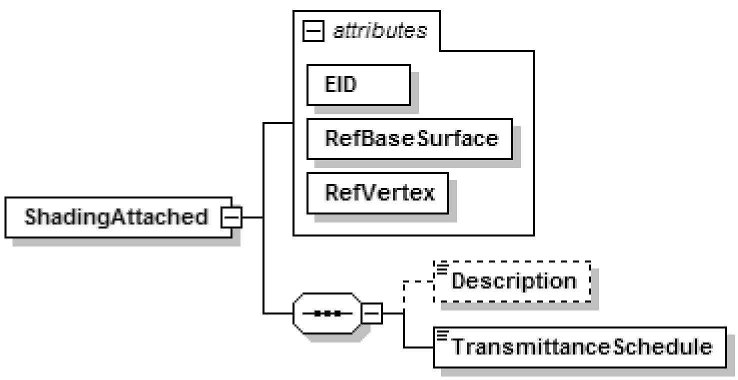
properties
isRef 0
content simple
default 1

source

```
<xs:element name="WindExposed" type="xs:boolean" default="1"/>
```

element **ShadingAttached**

diagram



properties
content complex

children **Description TransmittanceSchedule**

used by
element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefBaseSurface	**xs:IDREF**	required			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="ShadingAttached">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Description" type="xs:string" minOccurs="0"/>
   <xs:element name="TransmittanceSchedule" type="xs:string" nillable="true"/>
  </xs:sequence>
  <xs:attribute name="EID" type="xs:ID" use="required"/>
  <xs:attribute name="RefBaseSurface" type="xs:IDREF" use="required"/>
  <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
```

```
  </xs:complexType>
</xs:element>
```

element **ShadingAttached/Description**

diagram

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **ShadingAttached/TransmittanceSchedule**

diagram

type **xs:string**

properties

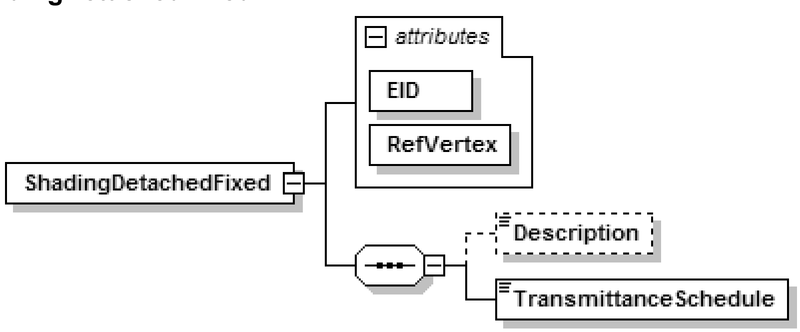
isRef	0
content	simple
nillable	true

source

```
<xs:element name="TransmittanceSchedule" type="xs:string" nillable="true"/>
```

element **ShadingDetachedFixed**

diagram



properties

content	complex

children **Description TransmittanceSchedule**

used by

element	**Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="ShadingDetachedFixed">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="TransmittanceSchedule" type="xs:string" nillable="true"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **ShadingDetachedFixed/Description**

diagram

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **ShadingDetachedFixed/TransmittanceSchedule**

diagram

type **xs:string**

properties

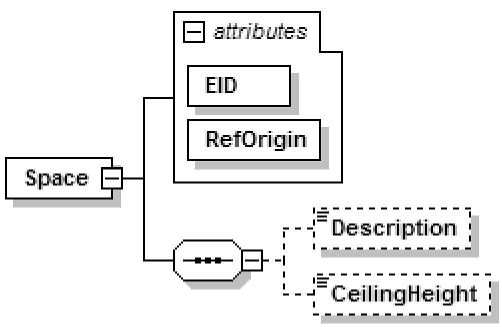
isRef	0
content	simple
nillable	true

source

```
<xs:element name="TransmittanceSchedule" type="xs:string" nillable="true"/>
```

element **Space**

diagram



properties

content	complex

children **Description CeilingHeight**

used by

element	**Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefOrigin	**xs:IDREF**	required			

source

```
<xs:element name="Space">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Description" type="xs:string" minOccurs="0"/>
   <xs:element name="CeilingHeight" minOccurs="0">
    <xs:simpleType>
     <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="EID" type="xs:ID" use="required"/>
  <xs:attribute name="RefOrigin" type="xs:IDREF" use="required"/>
```

```
  </xs:complexType>
</xs:element>
```

element **Space/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source

```
<xs:element name="Description" type="xs:string" minOccurs="0"/>
```

element **Space/CeilingHeight**

diagram 

type restriction of **xs:double**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

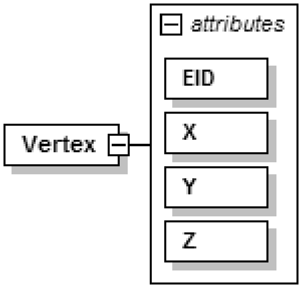
facets

minExclusive	0.0

source

```
<xs:element name="CeilingHeight" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:double">
      <xs:minExclusive value="0.0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

element **Vertex**

diagram 

properties

content	complex

used by

element	**Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
X	**xs:double**	required			
Y	**xs:double**	required			
Z	**xs:double**	required			

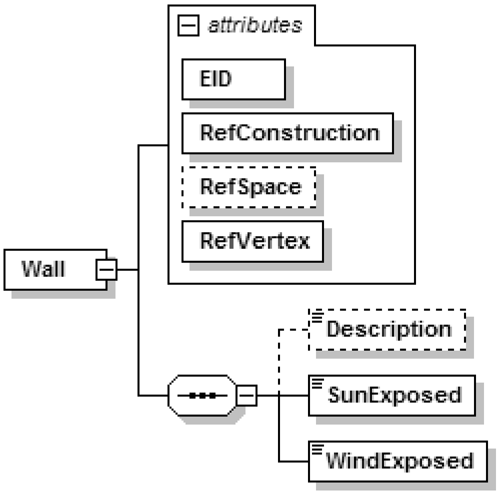
source

```
<xs:element name="Vertex">
  <xs:complexType>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
```

```
    <xs:attribute name="X" type="xs:double" use="required"/>
    <xs:attribute name="Y" type="xs:double" use="required"/>
    <xs:attribute name="Z" type="xs:double" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Wall**

diagram



properties — content: complex

children — **Description SunExposed WindExposed**

used by — element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefSpace	**xs:IDREF**	optional			
RefVertex	**xs:IDREFS**	required			

source

```
<xs:element name="Wall">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Description" type="xs:string" minOccurs="0"/>
      <xs:element name="SunExposed" type="xs:boolean" default="1"/>
      <xs:element name="WindExposed" type="xs:boolean" default="1"/>
    </xs:sequence>
    <xs:attribute name="EID" type="xs:ID" use="required"/>
    <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
    <xs:attribute name="RefSpace" type="xs:IDREF" use="optional"/>
    <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
  </xs:complexType>
</xs:element>
```

element **Wall/Description**

diagram

type — **xs:string**

properties — isRef: 0; minOcc: 0

maxOcc 1
content simple

source <xs:element name="Description" type="xs:string" minOccurs="0"/>

element **Wall/SunExposed**

diagram

type **xs:boolean**

properties
isRef 0
content simple
default 1

source <xs:element name="SunExposed" type="xs:boolean" default="1"/>

element **Wall/WindExposed**

diagram

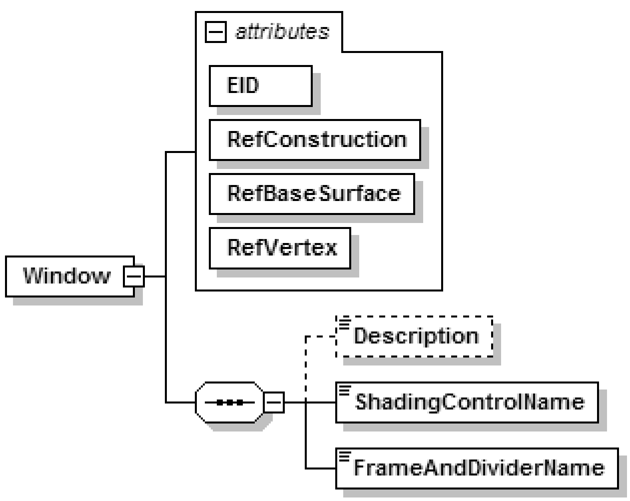
type **xs:boolean**

properties
isRef 0
content simple
default 1

source <xs:element name="WindExposed" type="xs:boolean" default="1"/>

element **Window**

diagram



properties
content complex

children **Description ShadingControlName FrameAndDividerName**

used by
element **Project**

attributes

Name	Type	Use	Default	Fixed	Annotation
EID	**xs:ID**	required			
RefConstruction	**xs:IDREF**	required			
RefBaseSurface	**xs:IDREF**	required			
RefVertex	**xs:IDREFS**	required			

source <xs:element name="Window">

```
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Description" type="xs:string" minOccurs="0"/>
        <xs:element name="ShadingControlName" type="xs:string" nillable="true"/>
        <xs:element name="FrameAndDividerName" type="xs:string" nillable="true"/>
      </xs:sequence>
      <xs:attribute name="EID" type="xs:ID" use="required"/>
      <xs:attribute name="RefConstruction" type="xs:IDREF" use="required"/>
      <xs:attribute name="RefBaseSurface" type="xs:IDREF" use="required"/>
      <xs:attribute name="RefVertex" type="xs:IDREFS" use="required"/>
    </xs:complexType>
  </xs:element>
```

element **Window/Description**

diagram 

type **xs:string**

properties

isRef	0
minOcc	0
maxOcc	1
content	simple

source <xs:element name="Description" type="xs:string" minOccurs="0"/>

element **Window/ShadingControlName**

diagram 

type **xs:string**

properties

isRef	0
content	simple
nillable	true

source <xs:element name="ShadingControlName" type="xs:string" nillable="true"/>

element **Window/FrameAndDividerName**

diagram 

type **xs:string**

properties

isRef	0
content	simple
nillable	true

source <xs:element name="FrameAndDividerName" type="xs:string" nillable="true"/>

Appendix D
EXPRESS: information modelling language

The following sections contain details on EXPRESS basic types, constructors, entities, attributes, rules and programming constructs.

Basic types

A set of pre predefined basic types are provided with EXPRESS and available for use in the definition of higher level types. The basic types are `NUMBER`, `REAL`, `INTEGER`, `STRING`, `LOGICAL`, `BOOLEAN` and `BINARY`. `INTEGER` is an unconstrained whole number and `REAL` is an unconstrained rational, irrational or scientific number. `NUMBER` is a supertype of `INTEGER` and `REAL`. `STRING` is a quoted list of characters, bounded by single quotes. `LOGICAL` types have the values (`TRUE, FALSE, UNKNOWN`) while `BOOLEAN` only has (`TRUE, FALSE`). `BINARY` is a vector of binary values, with user-defined encoding. EXPRESS also has an enumeration type, where the possible values are defined explicitly; e.g.

```
TYPE consultants = ENUMERATION OF (architect, engineer, planner,
urban designer);
END_TYPE;
```

Constructors

Constructors are used to aggregate variables and other structures of the same type into larger groupings. EXPRESS has four different types of constructors namely `ARRAY`, `BAG`, `LIST` and `SET`. `ARRAY` is used to define an ordered list of elements of fixed size which can be concatenated, e.g.

```
point : ARRAY[1:3] OF REAL;
```

`BAG` is an unordered collection of like elements. Duplicates are allowed. Its lower bound and upper bound may or may not be specified; e.g.

```
bag_of_points : BAG[3:?] OF point;
```

`LIST` is an ordered collection of like elements, similar to the `ARRAY`, but `LIST` may have a variable length; i.e., LIST may grow or shrink but remains ordered with the subscripts as assigned. A wall can be defined as a `LIST` of points with a minimum of 3 points, e.g.

```
wall : LIST[3:?] OF point;
```

SET is an unordered collection of like elements where duplicates are not allowed. SETs may be of fixed or variable size, e.g.

```
set_of_buildings : SET OF [1:100] OF building;
set_of_buildings : SET OF [1:?] OF building;
```

A generalization of all the constructor types is the AGGREGATE. AGGREGATE may be used in any declaration where any of the constructors may be utilized. In general, access to items in any aggregate (ARRAY, BAG, LIST, SET) is by using subscripts ranging from the lower to the upper bounds. The SELECT type definition (in other languages, it is known as union type) allows specification of a type that is a selection from among a set of types, e.g.

```
TYPE NUMBER = SELECT(REAL,INTEGER);
END_TYPE;
```

Entities

An entity is a general object type supporting definition of a wide range of complex elements, from which instances of objects are made; e.g.

```
ENTITY point;
x,y,z : REAL;
END_ENTITY;
```

This point may have instances that may be independent of any objects using the point object. Instances of point may be defined, while a type may only be used to define an Entity or another type. Entities can be inherited or subtyped into other Entities, e.g.

```
ENTITY homogeneous_point SUBTYPE OF (point);
w : REAL;
END_ENTITY;
```

An Entity may define both SUBTYPE and SUPERTYPE relations with other types. It may be a SUBTYPE to zero, one, or more than one other type. Similarly, an Entity may be a SUPERTYPE to any number of subtypes. Either the SUPERTYPE or the SUBTYPE may define a relation between the two Entity types. One or both Entity types may carry the relation declaration. However, all SUPERTYPE and SUBTYPE declarations, taken together, must be consistent with a directed acyclic graph. That is, an Entity cannot be a SUBTYPE and a SUPERTYPE of the same Entity, at any level of the relationship. There can be no cycles in the Entity subtype graph. A SUBTYPE inherits all attributes of the SUPERTYPE, including DERIVED and INVERSE attributes. Overwriting of rules is not allowed in EXPRESS which addresses the consistency of polymorphic types.

Attributes

There are three general kinds of attributes in EXPRESS: *explic*it, *derived* and *inverse*. In *explicit*, the values are provided directly where in the *derived*, the values can be calculated from other attributes. *Inverse* captures the relationship between the entity being declared and the named attribute. If an empty value is allowed, an attribute is declared as `OPTIONAL`. Like object-oriented languages, EXPRESS does not have a structure for defining relations and they must be defined using attributes.

Rules

EXPRESS supports the definition of a variety of rules that can implement semantic conditions of importance to product modelling. It provides a variety of structures for embedding these rules and allows definition of restrictions on allowed values or combination of values in the attributes of an Entity. Domain rules are specified using a `WHERE` clause within an Entity specification, e.g.

```
ENTITY vector;
a, b, c :
REAL;
WHERE
length1 : a**2 + b**2 + c**2 = 1.0;
END_ENTITY;
```

The domain rule `length1` is an integrity constraint of type `LOGICAL` (all `WHERE` clauses are of type `LOGI`CAL). When accessed, it evaluates the expression and returns one of the values: `TRUE`, `FALSE` or `UNKNOWN`. EXPRESS incorporates a fairly complete set of system functions (`ABS`, `ACOS`, `ASIN`, `EXISTS`, `EXP`, `FORMAT`, etc.) in order to define complex rules.

Programming constructs

A large set of language constructs have been incorporated in EXPRESS to define the expressions for `FUNCTION`s, `WHERE` clauses and `DERIVE` clauses. These are similar to those provided in standard programming languages, such as C. `FUNCTION`s are used as repeatedly called routines in the definition of complex derivations or rules. Most commonly used construct is `Typeof`, used extensively in IFCs - used as a function that returns a set of strings of all the types of which the parameter element instance is a member. EXPRESS also incorporates a query language, allowing data carried within an EXPRESS schema to be interrogated, compared and extracted.

Appendix E
Structure of SPF

The following section contains a brief introduction to the structure of SPF (STEP Physical File).

Structure

SPF is structured into two sections: HEADER and DATA following the initial keyword ISO-10303-21. The HEADER section has a fixed structure consisting of 3 to 6 groups in the following order:

- FILE_DESCRIPTION: contains *description* and *implementation_level*, the version and conformance option of this file.
- FILE_NAME: contains *name* of the exchange structure; *time_stamp* indicating the time when this file was created; *author,* the name and mailing address of the person created this exchange structure; *organization,* the organization to whom the person belongs to; *preprocessor_version,* the name and version of the system which produced this STEP-file; *originating_system,* the name and version of the software system which originally created the information contained; and *authorization,* the name and mailing address of the person who authorized this file.
- FILE_SCHEMA: EXPRESS schema together with the object identifier.

There are 3 more header groups valid from version 3.0 or later. They are: FILE_POPULATION (*governing_schema, determination_method, governed_ sections*), SECTION_LANGUAGE and SECTION CONTEXT. The DATA section contains application data conforming to the schema stated in the HEADER section. Each instance has a unique integer identifier in #nnn format. Identifiers are used to refer to objects within the file through attribute values or aggregate members. Entity instances are written using an internal mapping where the name of the entity type is followed by a list of attributes in superclass-to-subclass order. Only explicit attributes are mapped. Inverse, derived and re-declared attributes are not listed since their values can be determined from other entities. Unset attributes values are given as '$' where as re-declared explicit attributes as derived in a subtype are encoded as '*' in the position of the supertype attribute. Instances of complex entities are represented by using either the internal mapping or the external mapping. Instances of single entity data types are represented by writing the name of the entity in capital letters and then followed by the attribute values in the defined order within parenthesis, e.g. '#234=IFCPOINT(…)'. The following is an example of a SPF:

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (
/* description */ (' An example of p21 file: SPF ',
/* implementation_level */ '2;1');
FILE_NAME (
/* name */ 'example',
/* time_stamp */ '2005-05-20T04:00:12',
/* author */ ('Monjur Mourshed'),
/* organization */ ('University of Central Lancashire'));
FILE_SCHEMA (('IFC2X2_FINAL'));
ENDSEC;
DATA;
#1 = IFCORGANIZATION ('UCLAN', 'UCLAN', 'UCLAN’, $, $);
... ... ... ... ...
#7 = IFCSIUNIT (*, .LENGTHUNIT., $, .METRE.);
#8 = IFCSIUNIT (*, .AREAUNIT., $, .SQUARE_METRE.);
... ... ... ... ...
#1049 = IFCCARTESIANPOINT ((10.0, 2.7));
#1051 = IFCCARTESIANPOINT ((0.0 0.0);
#1052 = IFCPOLYLINE ((#1047, #1048, #1049, #1050, #1051));
... ... ... ... ...
ENDSEC;
END-ISO-10303-21;
```

Appendix F
IFC specification development process

This section contains a brief introduction to the IFC specification development process.

Specification development process

IFC specification development is based on a common view of AEC/FM data that can be shared by the AEC/FM industry. This common view is referred to as IFC Object Model, defined using a top-down approach. The development starts with a very general view of the AEC/FM industry; an overall model of a building can be defined and successively worked into a detailed model suitable for application development. Figure 1 shows an example of the development of usage scenario and Figure 2 shows process identification using diagrams. Following is a brief overview of the process:

- *Development of the usage scenarios*: Usage Scenarios are written descriptions of the processes that users perform, such as how a facilities manager maintains the various assets within a building. These usage scenarios capture the decisions and information that are used during each step of the process.
- *Identification of the process*: Process Diagrams give a visual representation of the process that is being defined. It is a diagrammatic representation of a usage scenario; e.g. the process of scheduling maintenance.
- *Development of the classes*: Classes are object-oriented templates to instantiate objects. They are designed to support the needs of the process and include concise definitions of the AEC/FM data objects.
- *Identification of the attributes*: Information about the class or its interface are called attributes; added to fully define an AEC/FM object. For example, *supply air volume* is an attribute for *centrifugal fan* class.
- *Establishing the relationships*: Relationships occur between classes; e.g. a space is related to wall(s) as a space is bounded by walls, and reciprocally walls bound a space. Relationships are important in defining object behaviour in ways that mimic the behaviour of the real world artefacts.
- *Defining the interfaces*: Interfaces are designed to support the AEC/FM processes and used to provide access to the corresponding object. It enables software developers to implement IFC based objects. For example, a fan object must support a variety of AEC/FM disciplines and include interfaces for costing, structural consideration, acoustic performance, among others.

- *Development of the object model*: The object model is used to represent the classes, their interfaces, attributes and relationships in a composite representation. The IAI uses Express-G for its model notation, which allows a graphical representation of the object model to be created.
- *Development of the Test cases*: Test cases are created to exercise the model using data from a predefined building and a specific usage matched to an industry practice. They are based on the usage scenarios developed along with the IFC information model and allow a software developer to test their application's ability to conform to the IFC Specification.

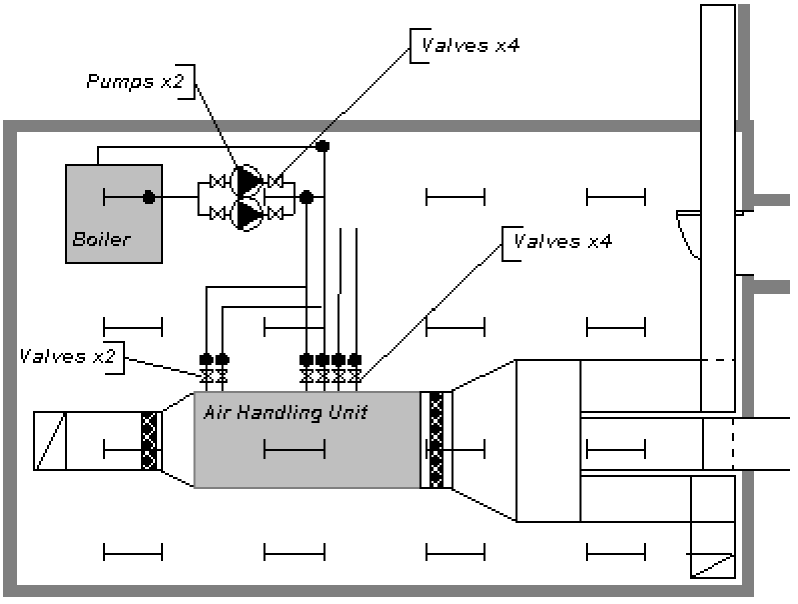


Figure F.1: An example of the development of usage scenario.

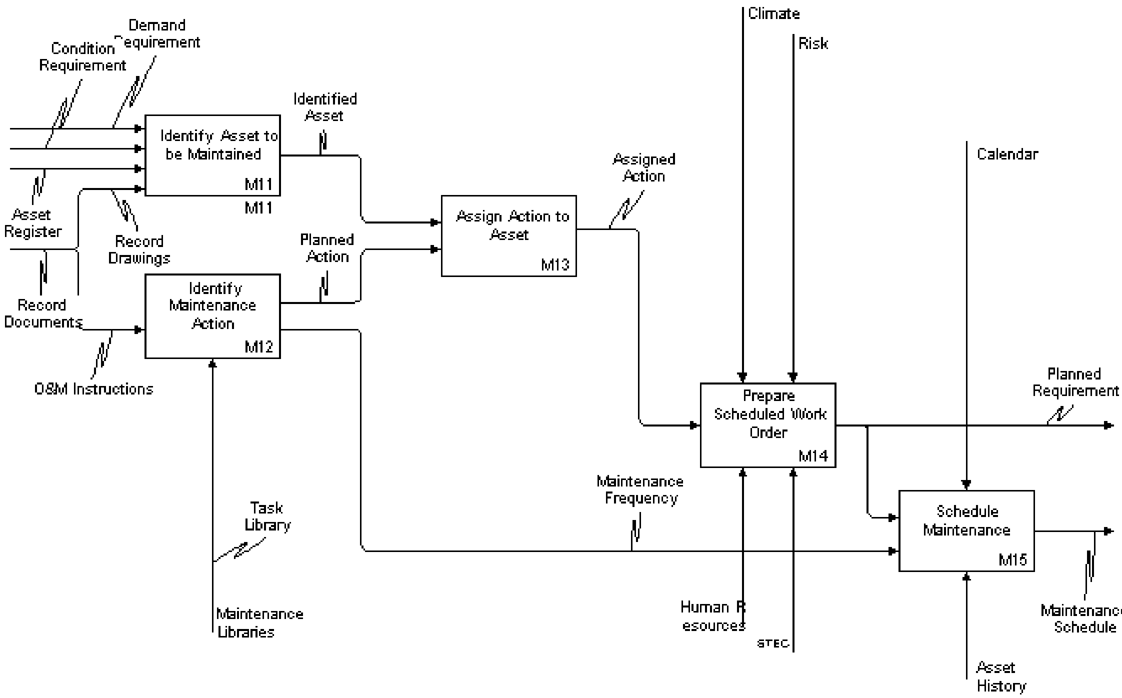


Figure F.2: Process identification using diagrams, source: http://www.bre.co.uk/.

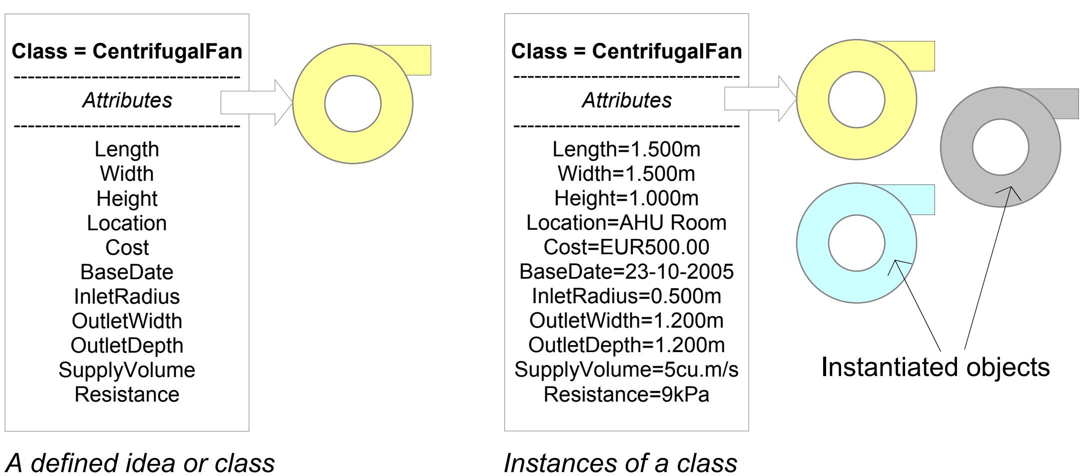


Figure F.3: IFC class and objects.

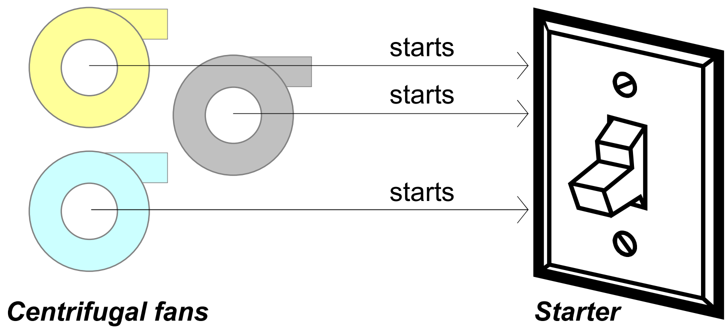


Figure F.4: Relationships between objects.

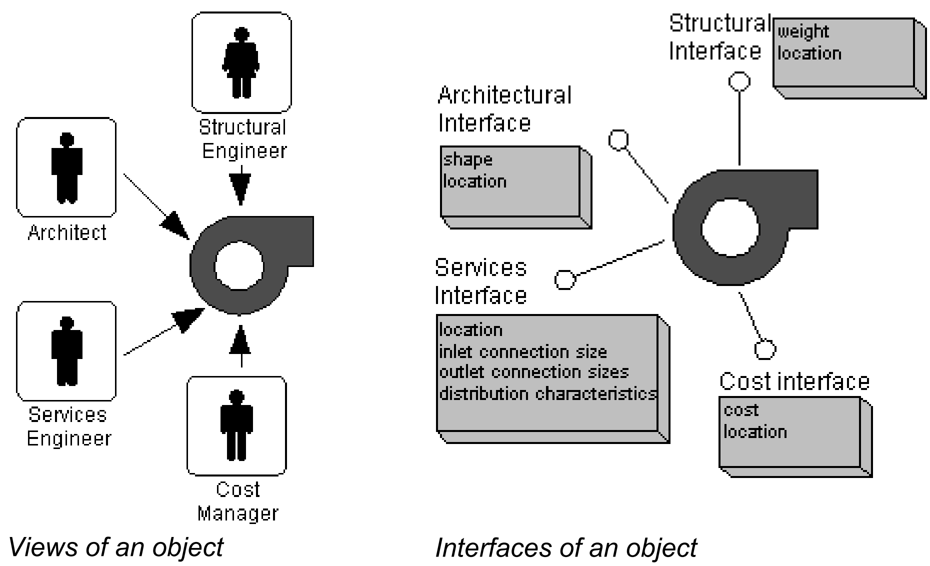


Figure F.5: Views and interfaces of an object.s

Appendix G
Climate analysis of locations

The following sections describe the methodology and analyses the climate characteristics of the chosen locations: Kilkenny, Ireland and Phoenix, USA.

Methodology

Pre-design climate analysis methodologies as described by Robinson (Robinson 2003) have been used to analyse the two chosen locations for this thesis. Some explanations are given in the following subsections.

Solar and daylighting parameters

The variation in Sun position throughout the year is visualised using a *sun-path diagram* to locate the position of the Sun at any time and day for a given location. The diagram is produced by plotting angular coordinates of the solar altitude (γ) and the azimuth (α) translated into planar x, y coordinates (Equations G.3 and G.4). Solar altitude (γ) at a given time and location is determined from the expression:

$$\sin\gamma = \sin\iota\sin\delta - \cos\iota\cos\delta\cos\omega \quad \text{(G.1)}$$

where: ι the site latitude, ω the solar hour angle and δ the solar declination angle.

The azimuth (α) is found from:

$$\cos\alpha = -\sin\iota\sin\gamma + \sin\delta / \cos\iota\cos\gamma; \omega < \pi \quad \text{(G.2)}$$

Conversion from angular to planar coordinates is expressed as:

$$x = (\pi/2 - \gamma)\sin\alpha \quad \text{(G.3)}$$

$$y = (\pi/2 - \gamma)\cos\alpha \quad \text{(G.4)}$$

The radial line in the diagram shows the solar azimuth and the concentric circles, the solar altitude.

Daylight availability curve is produced by charting the cumulative external horizontal illuminance distribution. An hour-centred time convention is used to prepare and archive the time-series data. By combining daylight factor and design illuminance, the daylight availability curve can assist in identifying measures necessary for solar design. *Irradiation surface plots* provides with the information on the variation of annual global solar irradiation (Whm^{-2}) over the range of receiving plane tilts (β) and orientations (α_w). The plots can inform designers on orientation and fenestration of the building as well as measures used for active solar energy conversion. Total irradiance, a combination of direct ($I_{b\theta}$), diffuse sky ($I_{d\theta}$) and diffuse ground ($I_{\rho g\theta}$) irradiances for the set of sun-up hours for 5 degree increments of collector tilt and orientation is plotted to achieve the surface plot.

$$I_{g\theta} = I_{b\theta} + I_{d\theta} + I_{\rho g\theta} \quad \text{(G.5)}$$

The beam ($I_{b\theta}$) or direct normal irradiance is determined by using the expression:

$$I_{b\theta} = (I_g - I_d)\cos\theta / \sin\gamma \quad \text{(G.6)}$$

where: I_g the global irradiance, I_d the diffuse irradiance, θ the angle of incidence and γ the solar altitude.

The cosine of the angle of incidence (θ) is expressed as:

$$\cos\theta = \cos\gamma\cos\alpha'\sin\beta + \sin\gamma\cos\beta \quad \text{(G.7)}$$

where: α' the collector-solar azimuth; $\alpha' = \alpha - \alpha_w - 2\pi$ for $\alpha - \alpha_w > \pi$, else $\alpha' = \alpha - \alpha_w + 2\pi$

The diffuse irradiance at the receiving plane ($I_{d\theta}$) from the sky vault follows Perez model (1990) which represents diffuse anisotropy due to circumsolar and zenith/horizon brightening:

$$I_{d\theta} = I_d\left([1 - F_1][1 + \cos\beta / 2] + F_1\cos\theta / \cos Z + F_2\sin\beta\right) \quad \text{(G.8)}$$

where: $F_{1,2}$ are circumsolar and horizon brightness coefficients.

Diffuse contribution to the ground ($I_{\rho g\theta}$) can be found as:

$$I_{\rho g\theta} = I_g\rho_g(1 - \cos\beta)/2 \quad \text{(G.9)}$$

To gain an understanding of the time varying nature of solar radiation as well as distribution of solar intensity, *solar radiation frequency distribution* graphs are used. It can be found useful in identifying the number of occurrences above or below some threshold.

Synoptic parameters

Synoptic parameters are related to temperature, humidity and wind. Wind roses1 are graphical representation of frequency of wind speed and direction divided into 16 segments of 22.5° each. The speed of the wind is represented as a distance from the centre of polar chart. This is done in 10 concentric rings for winds up to 50km.h. A colour scale indicates relative frequency. Average temperature, humidity and rainfall can also be displayed as roses to compare and contrast wind

[1] Wind roses used in this thesis are produced using Weather Tool, available from Square One research. http://www.squ1.com

data with other synoptic parameters to assist designers in deciding ventilation strategies. Bi-monthly ground temperature profiles can be used to explore the variation in ground temperature profile during the course of a year. The temperature at the soil surface t_s for the j th day can be found approximately using the expression:

$$t_s = \bar{t} - \tilde{t}\cos(2\pi[j - j']/365) \tag{G.10}$$

where: j' is the j day at which the minimum mean daily temperature occurred; $\bar{t}$ and $\tilde{t}$ are the annual mean and swing in mean daily temperature respectively.

Psychrometric chart reveals psychrometric extremities of synoptic climate data from which one can determine the need for and potential of some forms of evaporative cooling or dehumidication. The set of percentage saturation curves are defined at 10% increments (10→100) throughout the temperature range $-10 \le t \le 60^\circ$ C at 1° increments. Frequency and temporal distributions of temperature and humidity are also plotted in separate graphs.

Climate analysis of Kilkenny, Ireland

Kilkenny is located in the Republic of Ireland and is characterised as *Cfb: Marine Climate* of type *C: Moist Climate with Mild Winter* in Köppen classification. The sun-path diagram for Kilkenny in Figure G.6 shows typical moist mid-latitude climatic characteristics of the Northern hemisphere. The graph reveals that the solar exposure of north surface is less than that of south, east or west; hence passive solar design should consider reducing heat loss through openings on the north façade during winter months.

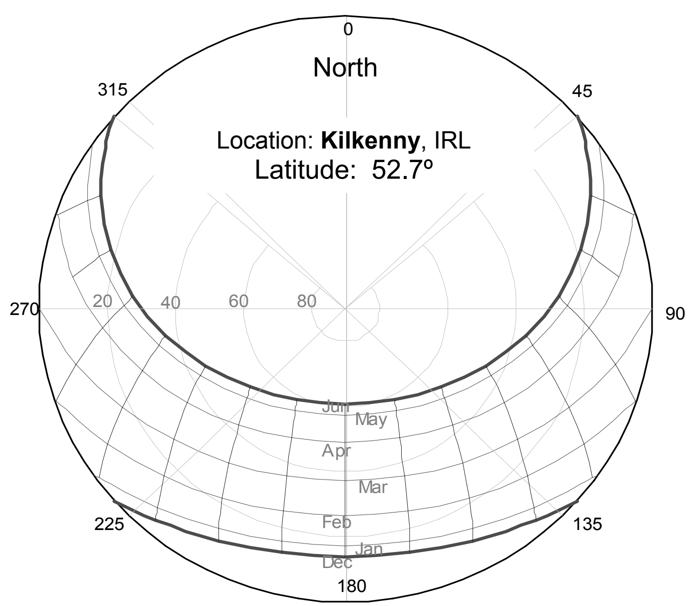


Figure G.6: Stereographic sun-path diagram for Kilkenny, latitude: 52.7º.

Solar availability contours in Figure G.7 backs up the assumption as global solar irradiation is the highest on the cardinal south and lowest on the cardinal north.

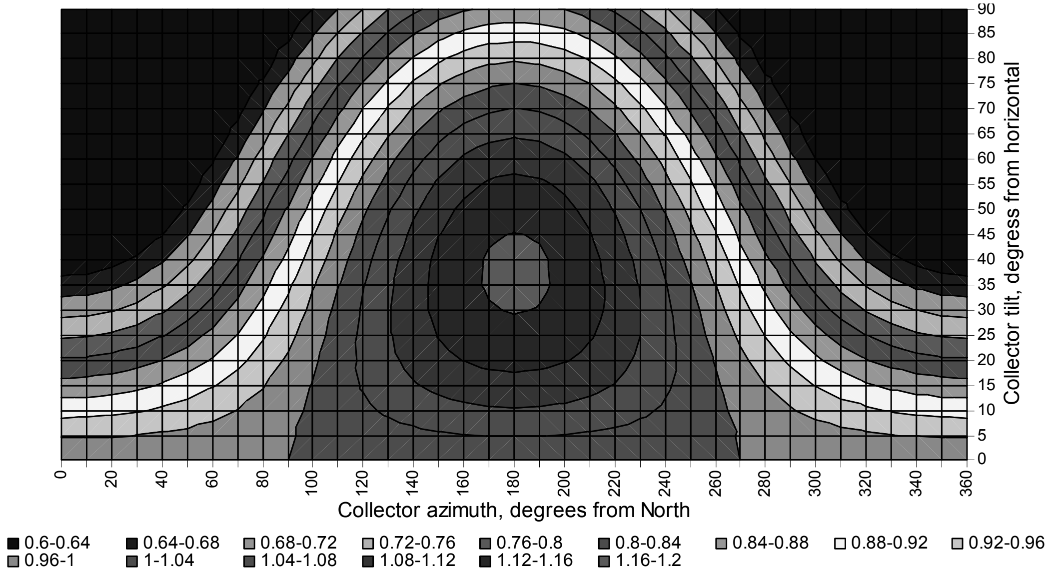


Figure G.7: Solar availability contours for Kilkenny.

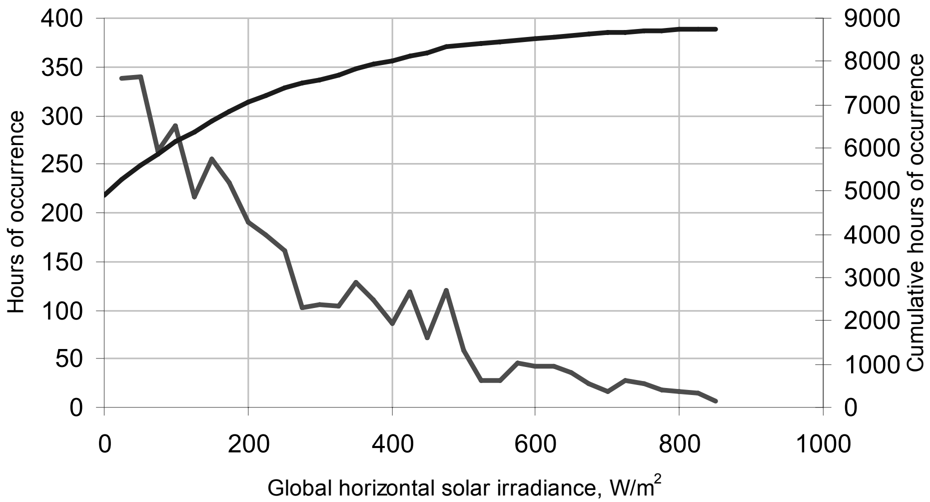


Figure G.8: Annual solar irradiance for Kilkenny.

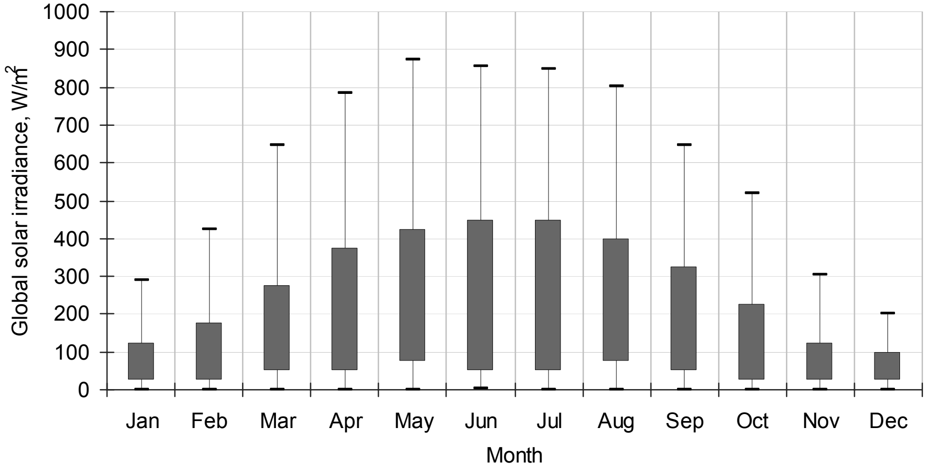


Figure G.9: Monthly solar minimum, maximum and upper/lower quartiles for Kilkenny.

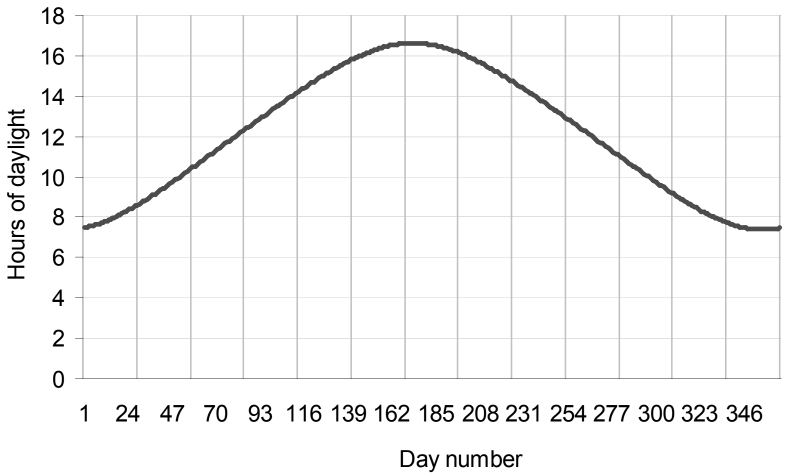


Figure G.10: Daily daylight hours for Kilkenny.

Solar irradiance is a vital parameter for Kilkenny as the climate is heating dominated with minimal cooling required during summer months. Figure G.8 shows hours of occurrence for different global horizontal solar irradiance values. Figure G.9 shows the monthly minimum and maximum as well as upper/lower quartiles. Solar irradiance up to 200 Wm^{-2} occurs for most of part of the year which indicates maximisation of glazing as a viable strategy for daylighting design. Figure G.10 shows daily daylight hours which can go below 8 hours during winter and can go above 16 hours during summer. This influences winter heat gain by building.

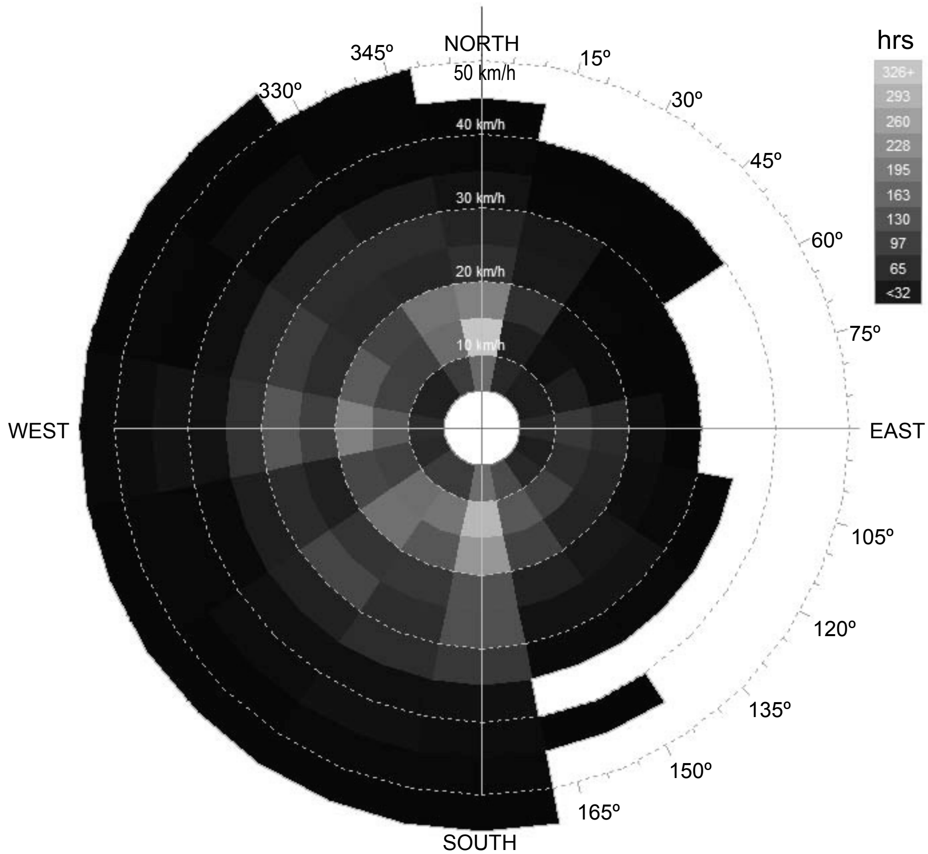


Figure G.11: Annual wind direction and frequency plot for Kilkenny.

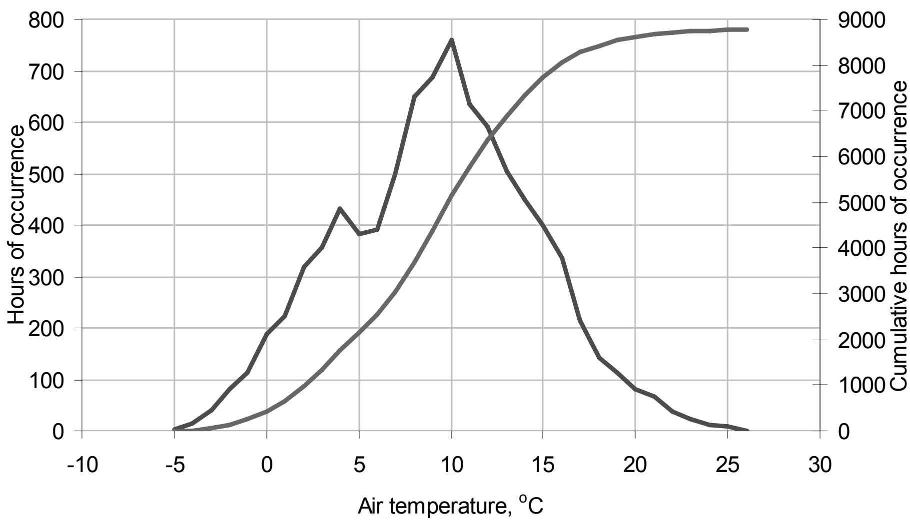


Figure G.12: Cumulative hours of occurrence of air temperature for Kilkenny.

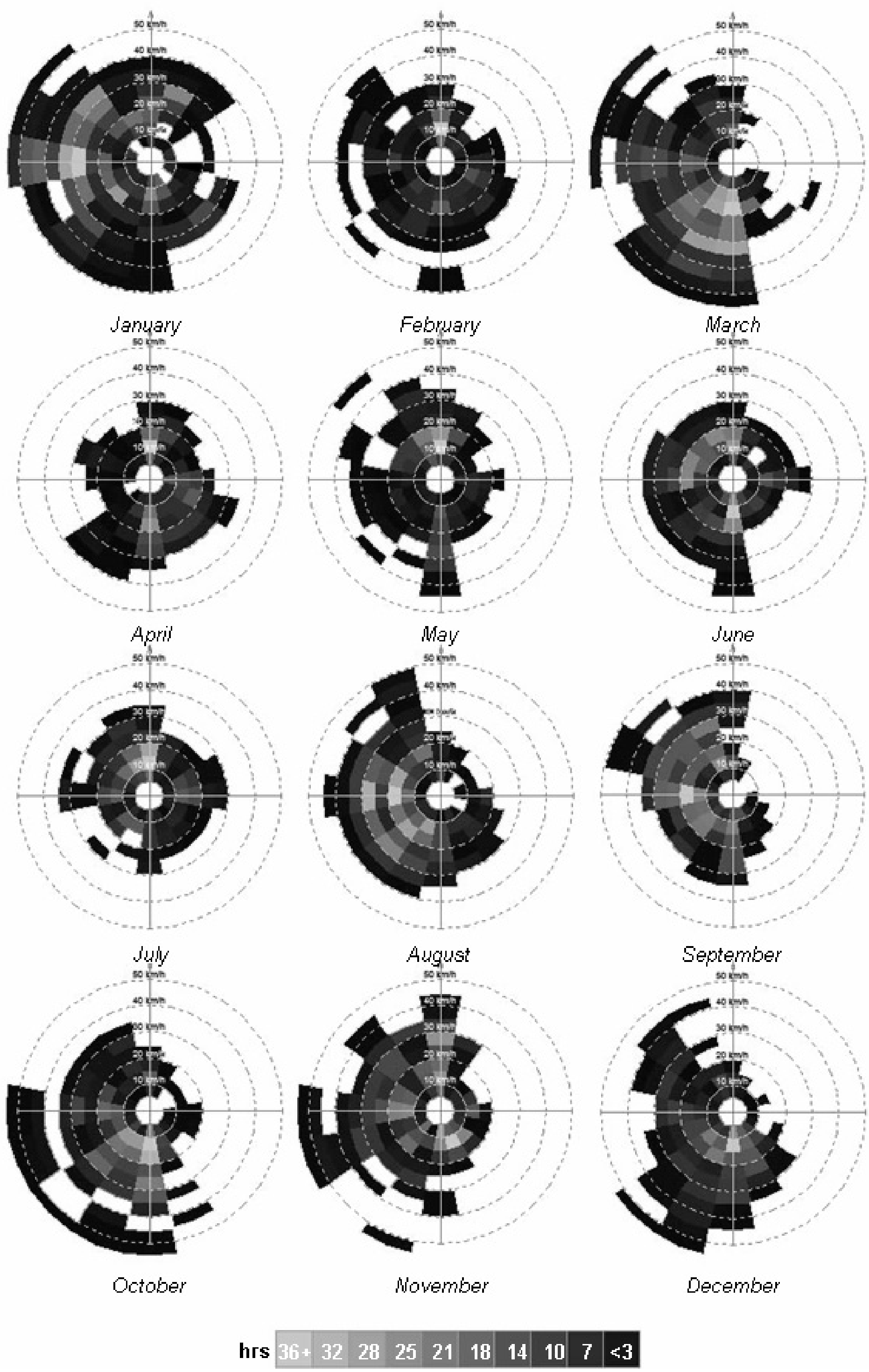


Figure G.13: Monthly wind direction and frequency plots for Kilkenny.

Figure G.11 shows annual wind direction and frequency for Kilkenny and Figure G.13 shows monthly distribution of wind direction and frequencies. Higher wind speeds above 50kmh^{-1} occur during winter months, usually for a shorter period of time. The prevailing wind directions are from the southwest, though all quadrants except north and east are generally well represented (Sweeny 1997).

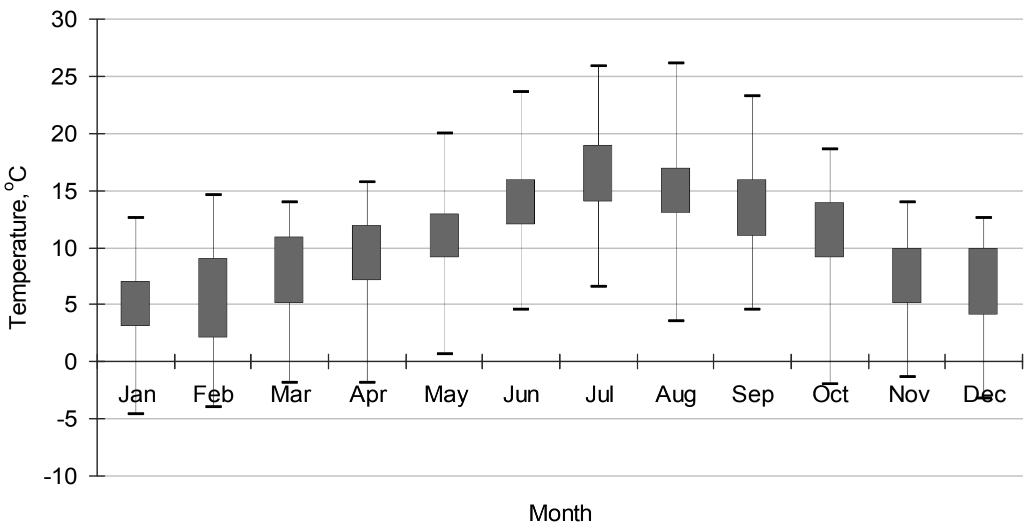


Figure G.14: Monthly air temperature - minimum, maximum, upper/lower quartiles for Kilkenny.

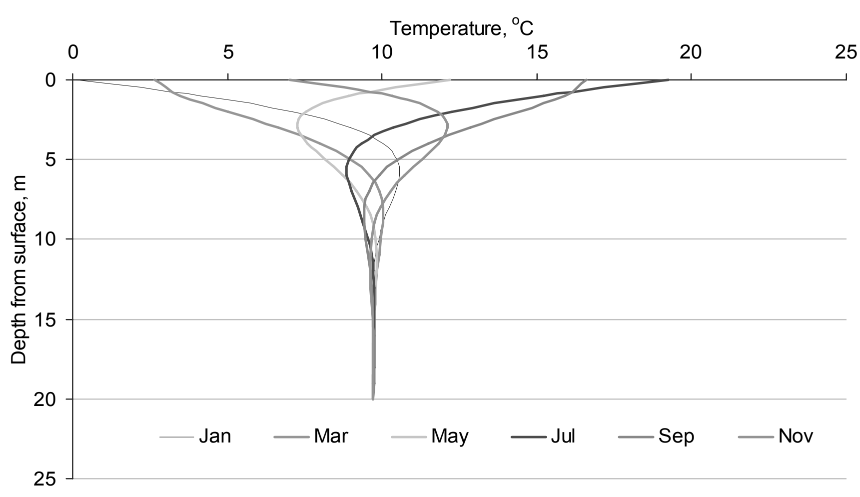


Figure G.15: Bi-monthly ground temperature profiles for Kilkenny.

Air temperature for Kilkenny varies from approximately -5°C during winter to just over 25°C during summer, shown in Figure G.14. Figure G.12 shows that air temperature between 5°C and 15°C occurs for two-thirds of the time in a year. The occurrence of temperatures over 25°C is comparatively rare and occurs during high summer months: July and August. Figure G.15, the Bi-monthly ground temperature profiles for Kilkenny shows the variation in ground temperature over the course of the design-year. Constant temperature of 10ºC at 10m depth from the ground level confirms the viability of *vertical loop* or *direct exchange* geothermal heating systems.

Psychrometric chart for Kilkenny in Figure G.16 reveals the synoptic characteristics of the climate and forms the basis for the selection of active/passive mechanisms to ensure occupant comfort. The chart shows that for a significant part of the year resultant plots lie close to the saturation humidity line. Considering the comfort for a normal sedentary adult @1.0 MET wearing light office wear (0.6 clo), it can be said that some forms of adiabatic dehumidification (usually by sorbents) or mixing of two air streams are required to keep the spaces comfortable.

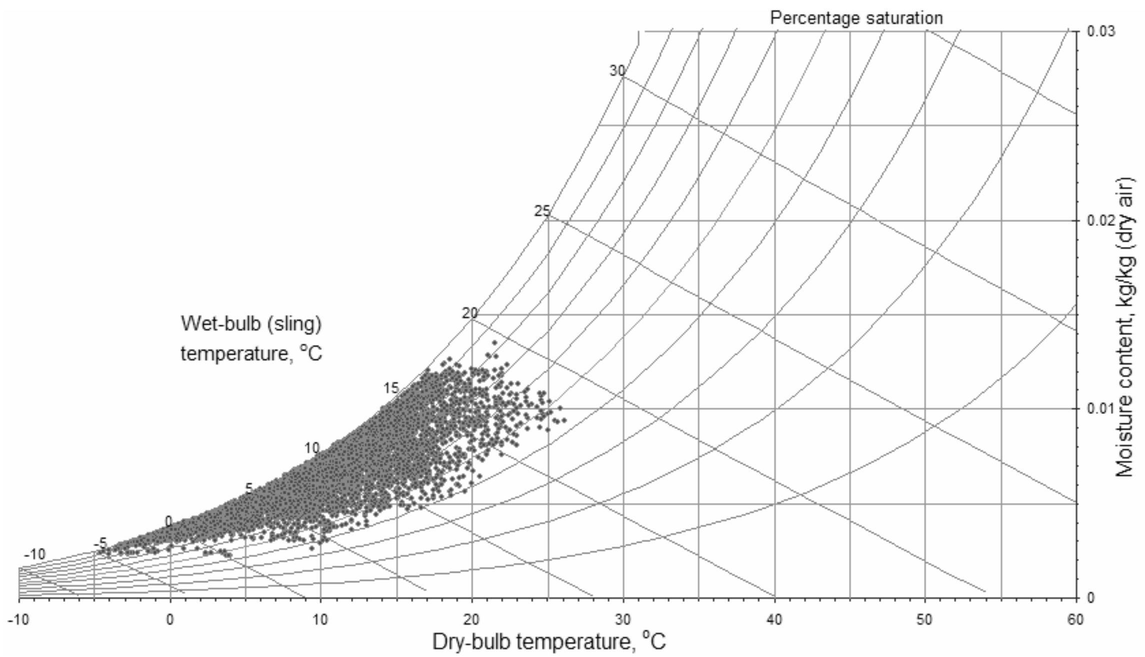


Figure G.16: Psychrometric chart for Kilkenny.

Climate analysis for Phoenix, USA

Phoenix is located in Arizona, United States. The climate is classified as of type *BWh: arid climate*, according to Köppen climate classification (Köppen 1884; Trewartha 1968). The climate is hot and dry with marked seasonal temperature extremes. It is located in the rain shadow of high mountain ranges and subject to orographic precipitation (see Park 2001). Precipitation is usually deficient and irregular. Average annual precipitation is about 19.6mm. Temperatures are characterised by hot, dry summers and mild winters. Maximum daytime temperature can reach up to 50°C.

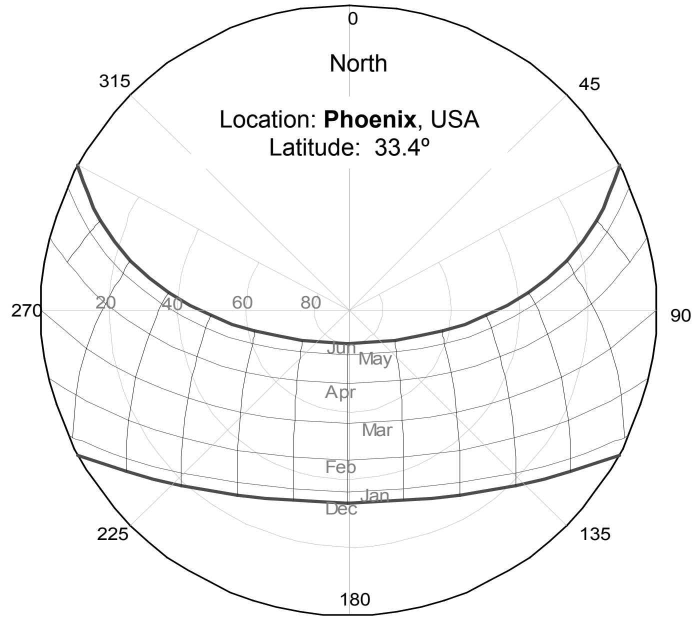


Figure G.17: Stereographic sunpath diagram for Phoenix, Latitude: 33.4º.

The stereographic sunpath diagram in Figure G.17 along with the solar availability contours in Figure G.18 reveals the extremities of solar irradiation in Phoenix, latitude: 33.4º.

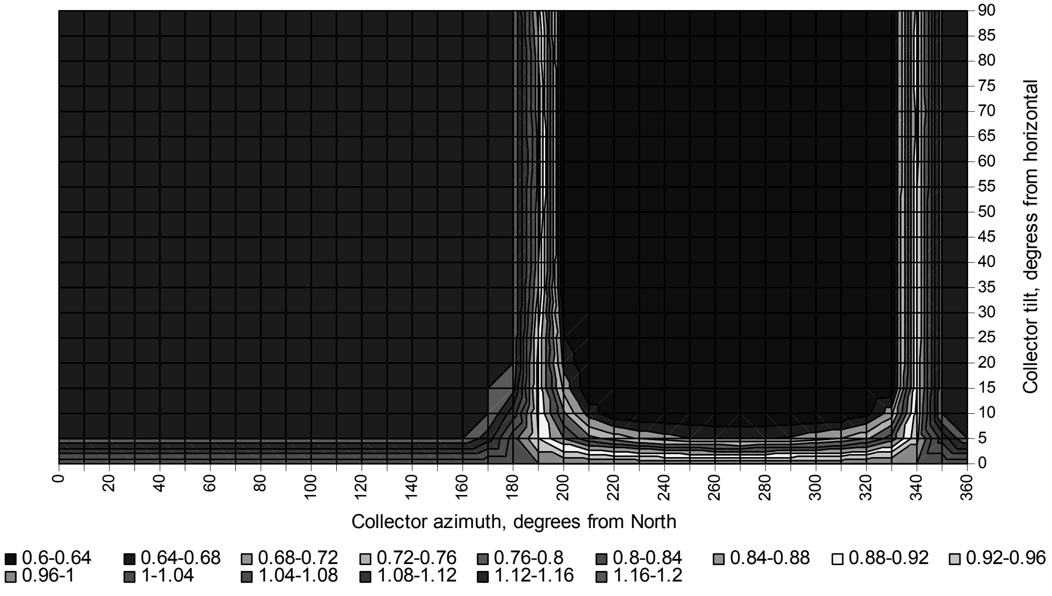


Figure G.18: Solar availability contours for Phoenix.

Collectors with an azimuth of 180-200º from north; i.e. collectors on the cardinal south, receive highest solar irradiation in Whm-2. The variation in collector tilt from horizontal does not alter solar irradiance received by the collector significantly. Passive design strategies in this climate need to consider minimising insolation. There is a significant difference between the solar availability contours of Kilkenny and Phoenix which is prominent in the distribution of curves with respect to collector tilts.

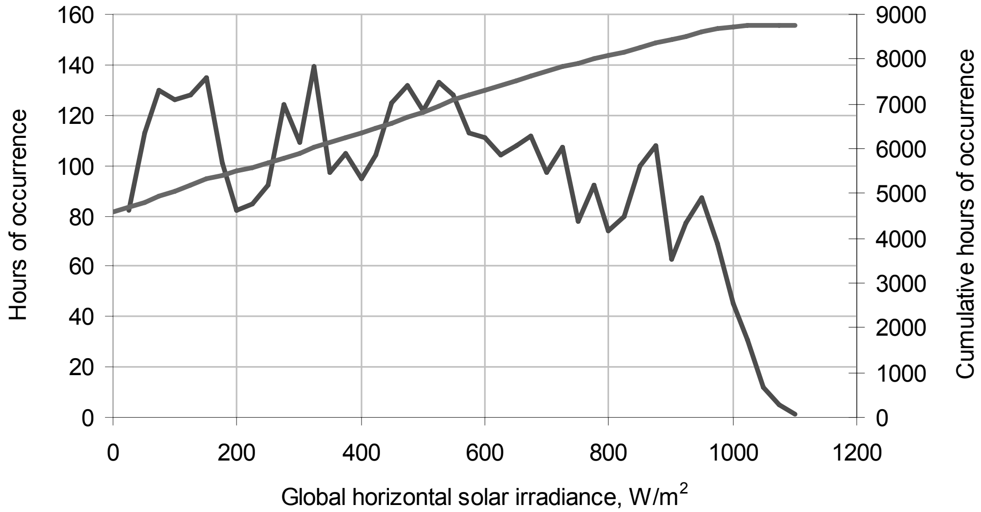


Figure G.19: Annual solar irradiance for Phoenix.

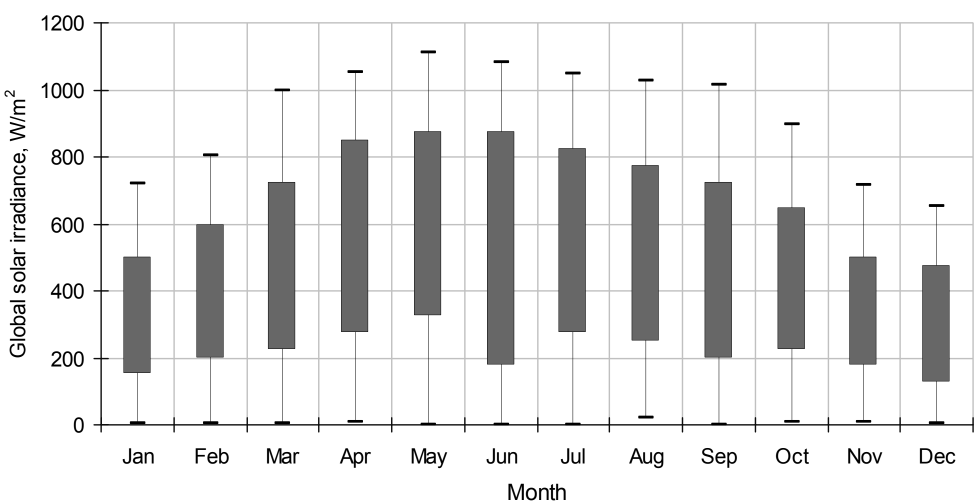


Figure G.20: Monthly solar minimum, maximum and upper/lower quartiles.

The chart on cumulative hours of occurrence of global horizontal solar irradiance for Phoenix in Figure G.19 is significantly different from that of Kilkenny in Figure G.8. The chart in Figure G.19, together with the upper/lower quartiles chart in Figure G.20 reveals the high availability of solar irradiance and cooling dominated nature of the climate. Average solar irradiance values of >300

Wm^{-2} during winter and >500 Wm^{-2} during summer indicate shading as an effective strategy for minimising insolation. Figure G.21, daily daylight hours for Phoenix, shows a much flatter curve than that of Kilkenny in Figure G.10. Winter values are around 10 for Phoenix, where as for Kilkenny it is around 8. Summer values for Phoenix is around 14 where as for Kilkenny it is around 16.

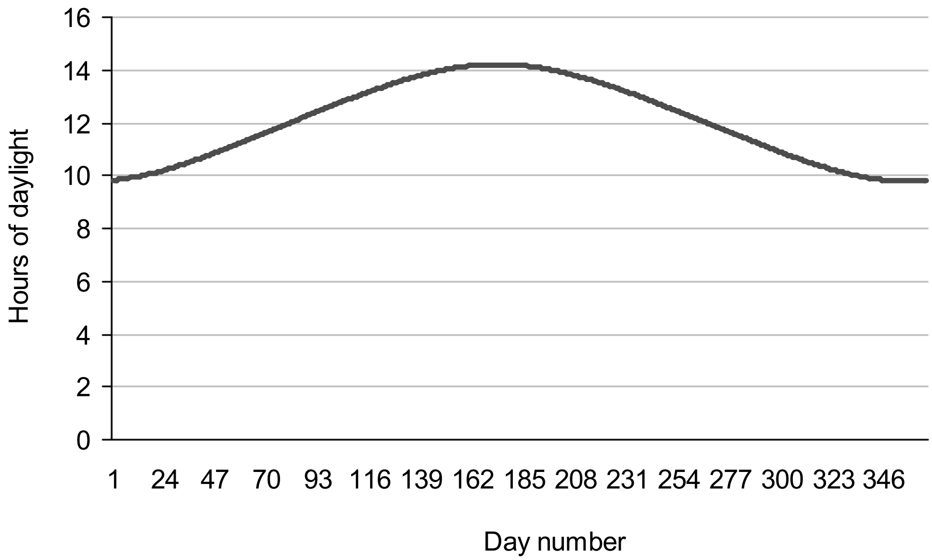


Figure G.21: Daily daylight hours for Phoenix.

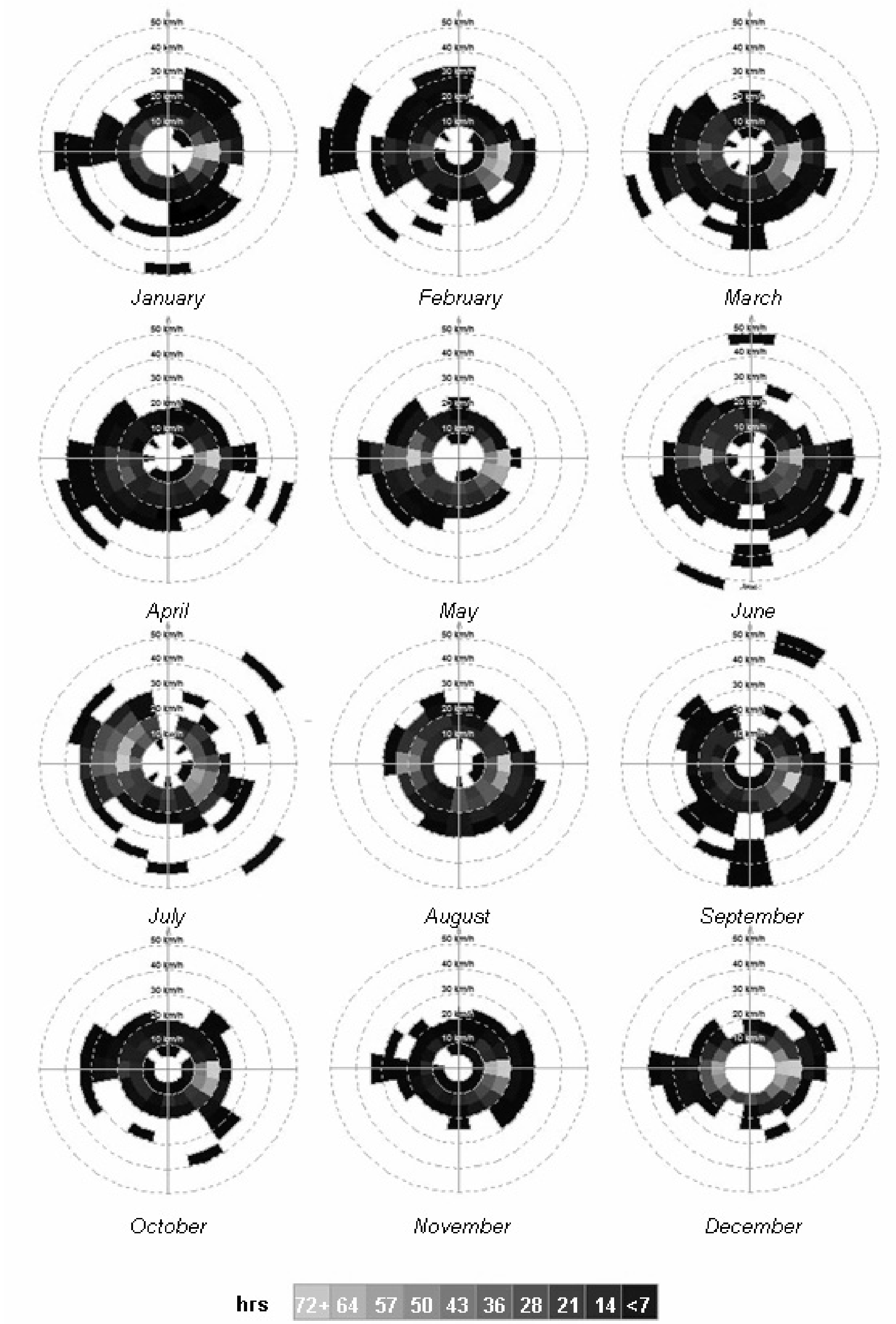


Figure G.22: Monthly wind direction and frequency plots for Phoenix.

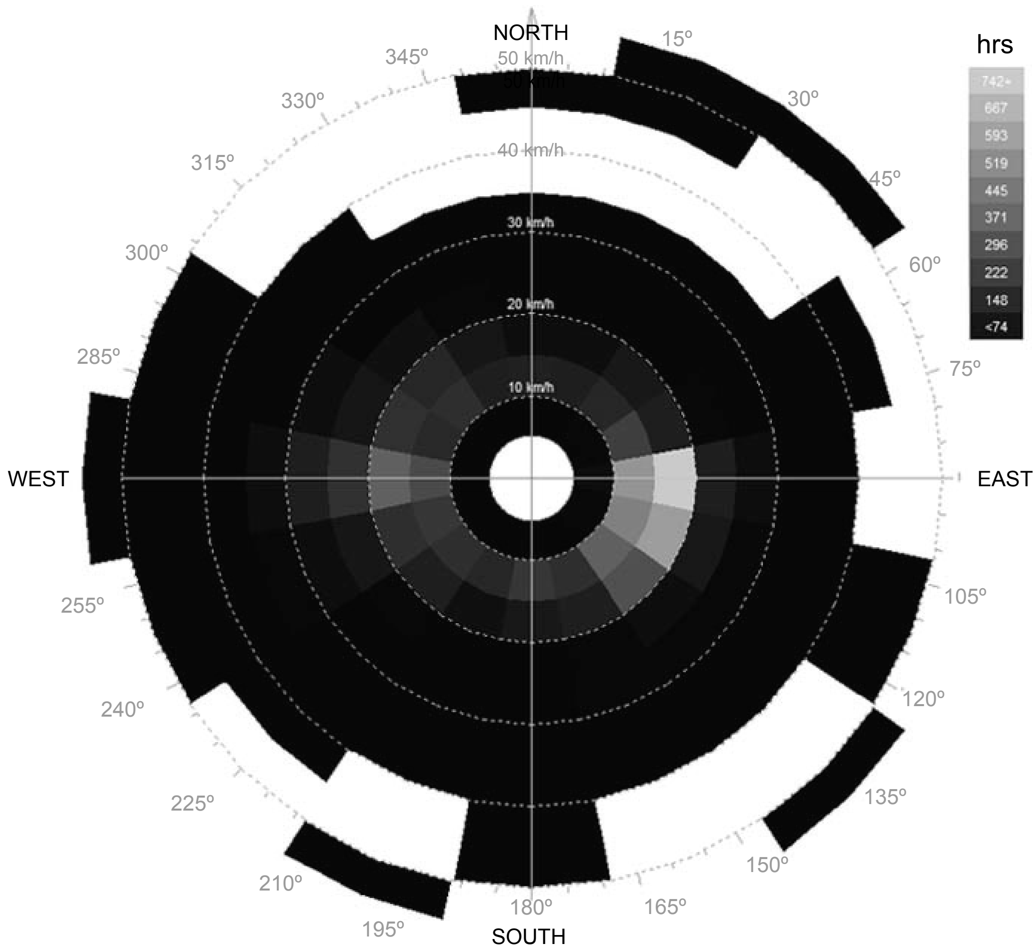


Figure G.23: Annual wind direction and frequency for Phoenix.

Figure G.23 shows annual wind distribution and frequency. Wind speed of 20 kmh^{-1} (approx.) occurs for most part of the winter predominantly from east. During summer wind tends to flow from west as well as from east. Mean value for wind speed is 11.52 kmh^{-1} from a direction of 141° from the north. To maintain comfortable indoor conditions, natural ventilation strategies could be adopted by exposing the building to prevailing breezes.

Monthly mean temperature varies from 11.3°C in December to 33.3°C in May. More importantly, minimum and maximum temperatures are above 19.8°C and 43.7°C respectively during summer as shown in Figure G.24. Minimum temperature occurs around 0600 hrs and rises to the maximum around 1600 hrs, which is vital for thermal design of commercial buildings with regular office occupancy. Figure G.25 shows air temperature between 20°C and 40°C occurring for half of the time in a year. A significant part of it occurs during daytime. Passive design strategies, in particular building orientation and form should consider the heating gain over summer periods. Figure G.26, the Bi-monthly ground temperature profiles for Phoenix shows the variation in ground temperature over the course of the design-year.

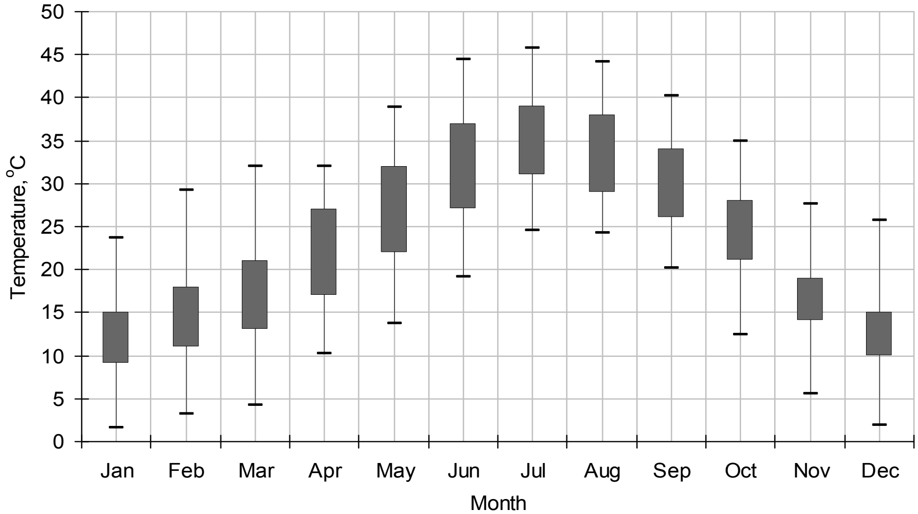


Figure G.24: Monthly air temperature - minimum, maximum and upper/lower quartiles.

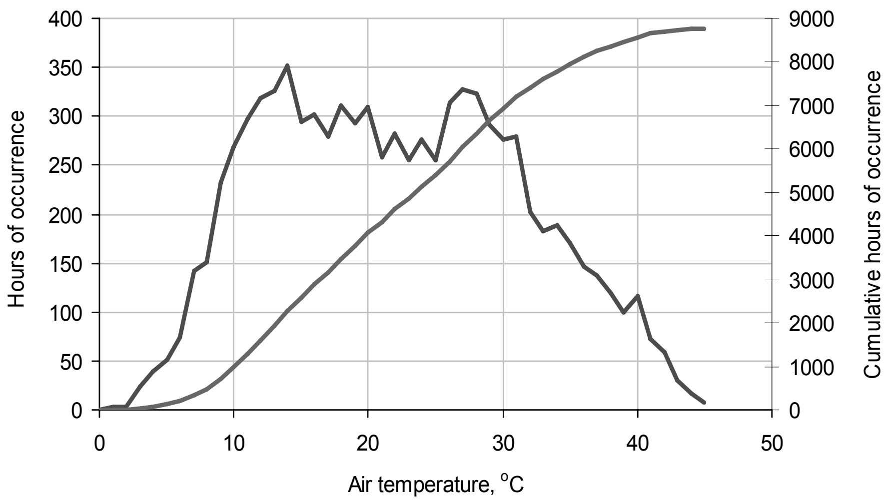


Figure G.25: Cumulative hours of occurrence of air temperature for Phoenix.

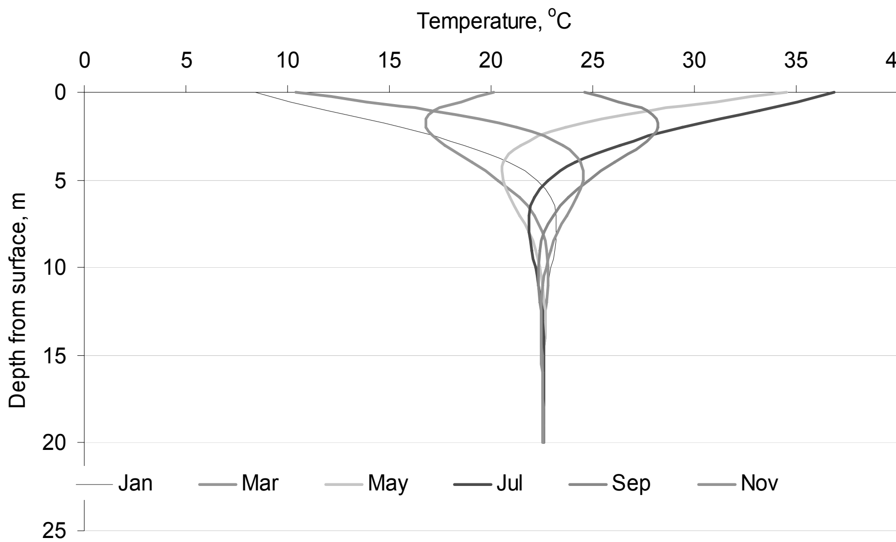


Figure G.26: Bi-monthly ground temperature profiles for Phoenix.

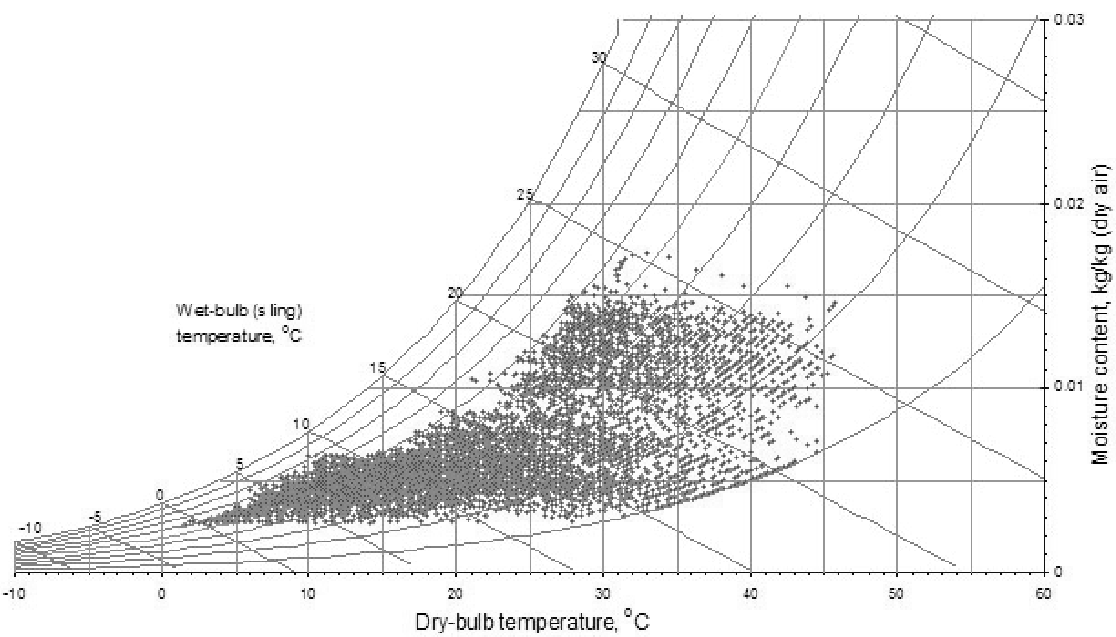


Figure G.27: Psychrometric chart for Phoenix.

Psychrometric chart Figure G.27 shows corresponding relative humidity is higher around 0600 hrs throughout the year and is lower around 1600±0100, depending on the time of the year. In summer time, relative humidity goes down as low as 4% and on average stays below 12% during regular office hours. This suggests evaporative cooling as an efficient strategy for occupant comfort. January to April, November and December are months substantially or less significantly under-heated. Daytime requirements during these months can well be taken care of by solar gains.

References

Köppen, W.P. (1884). Die Warmezonen der Erde, nach der Dauer der heissen, gemassigten und kaltern Zeit und nach der Wirkung der Warme auf die organische Welt betrachtet. *Meteorologische Zeitschrift*, 215-226.

Park, C. (2001). *The Environment: Principles and Applications*. London: Routledge.

Perez, R., Ineichen, P., Seals, R., Michalsky, J. and Stewart, R. (1990). Modeling daylight availability and irradiance components from direct and global irradiance. *Solar Energy*, **44**(5), 271-289.

Robinson, D. (2003). Climate as a pre-design tool. In: *BS2003: Eighth International IBPSA Conference*, Eindhoven, Netherlands, 1109-1116.

Sweeny, J. (1997). IRELAND. *Regional Climates of the British Isles*. Wheeler, D. and Mayes, J. London: Routledge**:** 253-275.

Trewartha, G.T. (1968). *An introduction to climate*. New York: McGraw-Hill.

This page is intentionally left blank

This page is intentionally left blank