The Electronic Building Simulation Assistant - constructing building descriptions that computers comprehend

Michael Donn
School of Architecture, Victoria University of Wellington
michael.donn@vuw.ac.nz www.aecsimqa.net

SUMMARY

This paper describes the principal features of a design decision support tool to be used by building designers. These features form the detailed conclusions to case study research of practitioners' and consultants' use of simulation. It posits a test that might be used for Quality Control in digital simulation of building performance as a necessary adjunct to the widespread use of simulation in design, particularly if it is to be used early in the design process.

the nature of design simulation

The crucial question posed by this research is about the nature of the information sought by building designers when they want environmental design decision support. This implies a fundamental assumption that improved building performance for the individual building owner or occupier is the goal of all designers.

This starting premise requires that merely asserting that architecture has *profound significance* (Baudrillard, 1999) or *embodies timeless laws* (Jackson, 1995) requiring architects to understand and acknowledge significant architectural *precedents* (Clark and Pause, 1996) does not equip these architects to adapt those precedents for the specific locations and uses of new buildings. What is required is a numerically quantified understanding of the relationship of those precedents to the environment.

Ultimately the research poses the issue of the role of the building designer and especially the degree to which a 'consultant' might provide the analytical input necessary for numerically based performance prediction? The conventional architectural notion (Schön, 1983) is of the architect as team leader. Building science researchers frequently conclude that a design approach compatible with these conventions (Gordon, Kantrowitz and Estoque, 1987) that is also to deliver quality environments should place central importance on *early design decision making* if their environmental design advice is to be effective. Indeed, many spend long hours developing design tools that are designed to improve the effectiveness of architectural decision making in the early stages of design (Balcomb, 1997).

This paper posits a form of 'consultant' for the designer which is akin to the digital butler suggested by Nicholas Negroponte. (Negroponte, 1995) A digital agent like this butler working for the building designer will watch the design as it progresses, prompting action by asking questions and showing the effects of design decisions by simulating the environmental performance.

The problem at present with performance prediction tools and software for these architects is that there is no simple to understand, independent measure of the reliability of the performance predictions for the e-building. Neither the expert consultant nor the building designer have a means of independently checking the data (Donn, 2001). Partly this is because the language of the design tool is unfamiliar to the designer. This problem, however, is being addressed by developers of a new generation of computer programs for environmental design analysis (Balcomb, 1997 and Hand et al, 1999, Crawley et al, 2001). Mostly it is that users of this type of software have no independent test of the quality of the performance prediction.

In order to build a design decision support tool that helps the designer to formulate a building that performs well, tools must be constructed that can function early in the design process when the building description is incomplete. Architects are strongly interested in the qualities of the environment that design tools describe (Donn, 1999). Where this interest has been noted in the past, design tools have been developed to provide beneficial 'output' from architects' 'inputs'. Examples of such developments include: Waldram diagrams (Waldram, 1923) R value calculations (Péclet, 1861) and daylight nomograms (Moore, 1985). These tools often require training intended to assist the architects to understand their application.

The reluctance of many architecture firms to get involved in the design performance prediction business (Donn, 2001) apparently has its origin in the belief of the senior members of architecture firm that numeracy skills are not part of the core business of an architecture firm. They also have no confidence in the skills of junior staff who may have received training in numerical building performance evaluation techniques because they have no way that they personally can determine the quality of the work done by this junior. What is attractive to these same senior members of the architecture firm about the recent availability of rendering software, which simulates lighting, is that, at least superficially, they can use their own traditional 'architectural' skills to assess the quality of the 'output' because this output is often published in the form of pictures. The issue is not the precision of the numbers - the number of decimal places in the 'answer' - but determining the accuracy of the relationship between the numbers and the reality they represent.

quality control - simulation and the real world

Design simulation requires a building designer to develop a mental model of the relationship between the real world and the information they are feeding into and getting back from the simulation. The quality of this mental model determines the quality of the information that they can obtain from the simulation. If a person does not understand the simulation process, they cannot easily use the simulation results to inform their design. Rather, the conscientious but uninformed user will have a series of numbers and a set of concerns about their meaning and reliability. There is an associated danger that the casual but uninformed user will have a series of numbers they trust unreservedly. This is most obvious in "Rules of Thumb" developed by building scientists for designers who do not want to deal with what they characterise as complex mathematics. At first the simplicity of a proposal in solar heated buildings to allocate an area equivalent to say 20% of the floor area to windows facing solar noon is seductively simple. It apparently fits that old truth test of scientific laws that they be elegant mathematically. In reality it is absolutely critical that users understand much more. They need to know far more about the origin of the Rule. Is it for houses alone? Is it dependent on the behaviour and expectations of people (how and when they heat for example)? Is it relevant to high and low thermal storage houses? A rule such as this based on research in 100 square meter houses would be irrelevant for 300 square meter houses.

With simulation software the problem is the same as it is with more simplistic algorithms. In order to obtain a calculated result in a time period that the design team finds acceptable, many mathematical tricks have been used to generate a simulation program that produces answers in hours or days not weeks or months. These tricks follow well-accepted mathematical methods for the solution of differential equations. However, they add artefacts to the calculation process that can confuse or undermine confidence in the output. In some cases they can place limitations on the degree to which reality can actually be modelled.

As McCullough (1996) has suggested, knowledge of the *affordances* of a computer tool is a skill that must be expected of the craftsperson of the 21st Century. He draws analogies with the way that a person skilled in the craft of wood carving knows the *affordances* that govern the relationship between chisel, grain and finish.

In the near future when design tools are more readily available because they are being incorporated into user-friendly computer analysis programs, and where clients are asking for more responsive building environments it is likely that many more people will be looking to use design decision support tools that analyse building performance. Many of these people will be untrained in their use and will trust the *black-box* (Tóth and Mayer, 1998) computer analysis program they have bought to analyse the thermal, visual or acoustic properties of their building. The purpose of Quality Assurance instruments in this situation is ultimately to provide people with the intuition for the application of their design decision support tool that marks genuine expertise - to help them to understand the affordances of each tool. While nothing can replace practice as a means of training a user, the goal of QA instruments has to be to ensure that the training is reinforced and strengthened every time the design tool is used.

This is ultimately reduced to a single issue: **how to ensure that the e-building constructed digitally with the simulation program is in fact the building we want it to be.** The digital simulation of the e-building's performance may well be valid. But how do we confirm that what we have described to the computer is first constructable and second is the building we imagine. QA processes in digital simulation should allow the user to understand and trust the relationship between performance predictions and building design. With clear feedback on the relationship between building and performance, architects would be much more likely to use simulation software. We need a means of calibration of the user and the software in combination so that the predictions are sufficiently well understood that they can become trusted design decision support.

What is proposed is a test for the output from a simulation program that is a Quality Assurance (QA) test that is like the *Turing Test* (Turing, 1950) for the 'existence' of computer-based (so-called artificial) intelligence. This test of the output of simulations programs would contain a number of practical assessment procedures that could be used by the program vendor to demonstrate the validity of their simulation process to their users. It could also be used to develop calibration procedures for these programs to enable the users to ensure that their use of the program produces reliable predictions of building environmental performance.

quality assurance - the reality test

The following statement is intended to be the type of truism in digital simulation that the Turing test is in artificial intelligence. Its careful application to digital simulation processes should generate Quality Control tests that convince the sceptics (Donn, 1999) that any of their design decisions supported by these simulation processes are dependable.

> ***Changes in the predictions of a simulation program with changes in building design should always be of the same scale and nature as those perturbations in performance observed in reality.***

This is an obvious 'truth' that most simulations, whether calculator or computer based, would claim to match. The claim for the value of many digital simulation tools for energy performance analysis has been that although the absolute numbers may not be completely reliable because users will operate the building differently than the computer does, the relative size of the changes in performance due to changes in e-building design is accurate. Equipment purchase or building design decisions are made on the basis of the predicted differences in energy performance.

The simulation QA test operates in the same manner as the Turing Test. It requires three "participants". The first is the simulation program; the second is the user of the program and the third is an external auditor. Simply, the "reality" test compares the e-building performance predicted by the program with the performance of other buildings. The building passes the QA test when the auditor cannot tell the difference in behaviour between the e-building and the standard building. Ideally the standard building performance data derives from a real building. However, the test requires the auditor to compare "*changes in the predictions .. with changes in building design …*". It is difficult to find measured data for buildings where changes in design have been made and the performance measured before and after. It is far easier to establish a test with other e-building data. Here the computer as agent would run the QA audit by finding a suitable standard building model via internet search and running a set of standard changes on it and the simulation program users' e-building and then comparing their performance. The most that the test could rely on as real performance data would be the measurements in laboratory test cells that form the basis of many validation exercises.

The QA test would be part of a design agent's role. It would have other roles. Even building scientists who are experts in digital thermal simulation need the agent because:

- ! they need a means of educating junior staff in the intricacies of thermal simulation - particularly in making relevant assumptions about the aspects of the design that are and are not important to model.
- ! they need to develop in new staff that healthy suspicion of the predictions of the computer that they have acquired through long years of experience.
- ! they need a means of simply guaranteeing the reliability of their conclusions - everyone has a horror of the misplaced decimal place deep within the intricate

melee of data that is a normal simulation input file.

- ! they need a way of finding standard output graphs which contain base cases (Gordon, Kantrowitz and Estoque, 1987) which allow the output to be measured consistently against well-characterised buildings.

The base cases would be described in detailed case notes and would represent relevant situations. They may even be generated by the software based on the user's choices when setting up the model of their building (e.g. it may be a standard building operated as the proposed building is modelled).

A new breed of architects are also in great need of an agent running a QA system that assists them to trust the design decision advice of digital simulation. They are the people who are being encouraged to look more carefully at the likely performance of their designs by their education, by the burgeoning market for solar and environmentally responsive design, and by the availability of 'user-friendly' software for thermal analysis of building performance.

This new breed needs:

- ! to learn how to use the digital simulation in a manner that does not place them open to litigation;
- ! to learn to trust the output of the digital simulation to the extent that they feel comfortable making design decisions weighing up say the thermal performance of the building against other criteria like view, aesthetic appearance and access;
- ! to produce evidence of building performance to support design decisions that is convincing for the client as well as themselves because it is described in language that can be readily understood.

To me the most interesting result of digital simulation design decision support is the likelihood that the client can become much more intimately involved in decisions about their future comfort *if the QA procedures are available to assist them to understand the performance predictions.*

In order to address the issues raised by the analyses in this paper, it will be necessary to develop a set of pre- and post-simulation tools that are incorporated into the Graphic User Interface of the digital simulation program. Ideally, they will be operated by "Agents" or "Bots" working with the simulation software on the users behalf. Thus, when a thermal simulation of a new school in a Temperate climate is planned, the designer asks the agent through their CAD program to assist.:

1) The agent finds similar buildings in local databases such as the Building Design Advisor (Papamichael, et al, 1997) datasets.
2) Using the internet, the agent searches the Building Performance Database for buildings in a similar climate with a similar function and with thermal data available.
3) The agent also searches these same databases for input data for the simulation program used by the designer.
4) The agent presents the designer with the precedents it has gathered and any associated performance analysis data .
5) The agent responds to the designer's query by suggesting a close match building as a design reference which will form the measure against which the new building design will be measured. Whether this is one of the buildings from the datasets or is a hybrid of one of them with local code minima for thermal insulation applied will be up to the designer to decide.
6) The agent will offer the designer a thermal simulation program input file based on the design reference. Some designers may use this as the starting point for constructing the input data file for their own design. Others may want the agent to provide an input file that matches the CAD data they have entered. This latter option will probably have to await the introduction of a fully functional Building Product Model.
7) The designer will "run" the simulation for their building. However, this command to "run" the simulation program is actually stated as an instruction to answer a question or series of questions. Rather than "run" an annual calculation of the energy required to heat and cool the building, the Agent will be instructed to:
    1) calculate the seasonal suitability of the building to the activities planned; e.g. if it is a house in a temperate climate it might explore how cold the house gets on a

winter morning when the heating system is turned off overnight.

2) calculate the size of the heating or cooling or ventilation plant required in suitably understandable increments: e.g. in a school in New Zealand, this might be how many of the designer specified opening windows would be needed to cope with the excesses of February sunshine, pupils and days with no wind.

3) perform a cost benefit analysis on one or other particular element in the building: e.g. in a tall deep plan commercial building in a cool climate, contrast the heating energy reduction against the cooling energy increase as a result of installation of wall insulation.

8) The Agent will report the simulation results in a format that identifies not only the simple answer to the question of veracity but also a list of which of the myriad input parameters has a significant influence on the answer. This will require the Agent to create a parametric input file and to test reactions to variations in the input parameters in order to be able to report which have the greatest influence on the building's performance.

9) The Agent will typically report all these results measured against the precedent(s) identified by the designer at the start of the analysis process.

10) Graphing reporting and data export functions will be required of all analysis agents at this point. But they will also be asked to conduct a "veracity" test - to use the internet to match the input and out put files to establish that this new simulation is behaving in a manner that is consistent with reality. Without this final Turing-style test of the reality of the simulation, and the internet data to make it happen, the simulation will have little credibility.

The crucial part of this process is to have building performance data against which to compare one's e-building performance.

Finding data on the web

In order to provide an internet based resource for the storage of building performance information that is accessible through familiar computer technologies, it will be necessary to respect the overarching goal for the most widely used part of the internet - the World Wide Web (Berners-Lee, 1999c) "*The Web was designed as an information space, with the goal that it should be useful not only for human-human communication, but also that machines would be able to participate and help*." (Berners-Lee,1998).

The most important feature of an internet-based building performance data resource is that it make the data somehow accessible. This requires use of leading edge internet technologies which make possible intelligent machine searches. The process could not work if one were to look for a daylit school in a temperate climate via standard internet key word searches. The computer must "know" what daylight, school and temperate mean.

the semantic web

The following analysis is drawn directly from the principles (highlighted in the text) espoused by Burners-Lee in his paper *Principles of Design* (Burners-Lee, 1999a):

Adoption of **XML** for documents describing building performance allows for mixing CAD and thermal simulation files in the one "document" describing a building's performance. Unless the system is tolerant of various ways in which files can be stored and delivered it will not work. It must be possible, for example, to store DOE2 data as ASCII or Binary. Or, for a 3D CAD file of a building's geometry to be in one of the many different proprietary formats for Computer Aided Design programs. As shown by the proliferation of non-standard HTTP which has lead to web pages that can be read by one web browser and not by another, the principle of **tolerance** has an inherent weakness: it can encourage a too liberal attitude on the part of the creators of building performance datasets.

The attraction of the XML format, is that the structured database format of input data describing a building for a thermal (or lighting or acoustic or air flow) simulation program is readily translated into the structured XML format. As noted above, the essential requirement of a computer program that performs the role of being the intelligent "agent" advising the designer about each step in the design process is that the agent understand the data it is working on. In the "semantic web" *most databases in daily use are relational databases - databases with columns of information that relate to each other, such as the temperature, barometric pressure, and location entries in a weather* *database. The relationships between the columns are the semantics - the meaning - of the data. These data are ripe for publication as a semantic web page ... the Resource Description Framework (RDF) which... is based on XML ... allows computers to represent and share data just as HTML allows computer to represent and share hypertext. ... In fact it is just XML with some tips about which bits are data and how to find the meaning of the data.* (Berners-Lee, 1999c)

The key with the *semantic* web in this proposal is that a document contains not only the data but the links or references to the places on the web where a computer program can find how to convert each term in the document it doesn't understand into a term it does understand. With the appropriate RDF's an XML document describing lighting performance measurements in an office building in Los Angeles might be used to create a realistic Radiance daylight simulation for San Diego this week; and next week it might form the basis of a DOE2 analysis of the impact of daylight on cooling equipment energy use in a LA doctors' surgery.

If each dataset is placed in cyber space with its own built-in RDF definitions, in an XML language document, then useful searches by a pre-processor could be constructed such as: "find all the mild climate office buildings monitored in the past 10 years for which lighting measurement and energy consumption figures are available". To find information to calibrate its predictions the agent doing energy performance simulation post processing uses a similar search concentrating only on buildings for which energy use data is stored. The simulation package authors do not need to have done a complete analysis of the knowledge representation required to construct a computer-based *'product' model of a building* (COMBINE, 1997) and hence of the translation of their input data into that model format. Rather, they need to provide a link from the program user to the RDF for their program. Inference engines (Berners-Lee, 1999d) developed by them or by others will provide the link to relevant data in other people's data formats.

To paraphrase Berners-Lee: machines can give the appearance of thinking by answering questions that cause it to follow the links in a large database. The database of relationships might be structured like: *a building is a thing, a house is a building, a door is a thing, a building has at least one door*. To create a useful database of this type is a huge and typically *has room for only one conceptual definition of* a house. The goal of the semantic web is to allow different sites to have their own definition of "house" and to develop an "inference layer" to allow machines to link definitions. RDF's are the inference layer.

A major further advantage of this approach is, in Berners-Lee's terms: *evolvability.* If an RDF exists for the input files for a program like DOE2 (Crawley et al, 1999) then when an old version encounters a file from a newer version it can look up the relevant RDF for the new version to find the parts of the new file it can "understand". The process of expanding the use of these QA tools then is one of evolution, and requires very little in the way of international or inter-disciplinary standardisation. It carries within it the RDF tools that permit adaptation and machine learning.

Burners-Lee (1999c) notes we may assume we will be smarter in the future and thus should ensure that we work by a principle that version 4 of our system should always be able to read the data from version 5, even if it misses some of it. He then points out we cannot assume we will be the smartest. Someone else may well devise a system that is even better. What is important and indeed, essential about the proposed database of building performance is that all its pieces could be re-used by other better or different systems. You need well-designed meta-data to enable a subsequent translation of the meaning of entry 'x' in one system to be (machine) translatable into the same meaning in entry 'y' in another system.

The rationale for this principle in web design is that the less powerful you make the language in which data is stored, the more each individual can do with the data stored in the language. I leave to others at this point the decision about the relationship between Building Product Models and the W3 organisation's published work on a Resource Description Framework (RDF) for Metadata. (Lassila and Swick, 1999a and 1999b)

summary

Advantage arises from the XML/RDF split in the presentation of data - on the web or anywhere else. This is the reasoning - the rules that define the relationships between parts of a building are explicitly removed from the simulation program revealing the reasoning behind the analysis very clearly. This separation has several benefits when seeking to apply a QA process in simulation.

First, the new simulationist often finds it puzzling how to determine the appropriate external environment to "apply" in a simulation. What analysts debate is how to characterise the 'typical' external environment. Is it an average day/week/year? What might the risk to the building owner or operator be if the normally expected variations around the average occur from year to year?

Stochastically valid risk analysis is essential in all QA procedures related to building performance simulation. In an XML system the weather data for thermal or lighting simulation would contain an RDF definition of its terms. This would enable a different XML-aware simulation to translate the columns of weather information to a format compatible with its own views of the world. It would also mean that each weather file would contain synoptic information on how typical it was which could then be used by the simulation package to construct atypical weather scenarios.

A second and often-overlooked aspect of the external environment is the operational environment. The designer needs to know just how vulnerable the simulated performance will be to variations in the way we occupy or operate the building. If we no longer operate the building as we assumed it would be, what might the performance consequences be? XML format data on the energy performance of other real or simulated buildings would contain data about the data (Metadata) in the file. This would describe the context for the measurements and hence permit the agent, the XML front end of the simulation package, to *infer* how "typical" the usage patterns are and hence how they might be tweaked to test the sensitivity of the simulation output to usage patterns.

Finally, the increased complexity of modern computer-based building performance simulation tools has not rid the design profession of its traditional problem with these tools: that they evaluate completed designs. Guidance about how to move forward in improving a design typically only comes only from the informed user looking backwards at how the existing design performs. Our XML based agent working in CAD with the design team would look up Post Occupancy Evaluation contributions to the Internet database. It might even generate initial design ideas based on successful precedents.

This research defines a development path for the next generation of design tools. It assumes this next generation of design tool will be more detailed computer programs. It also assumes that simulation programs like DOE2 and RADIANCE that a few expert "simulationists" currently use will increasingly be a part of the building designer's repertoire. QA based agents built around XML will be a significant part of that future. What is required is a store of data that can be drawn upon by building performance analysis programs around the world. We need only a central repository for pointers to the myriad locations of the actual data, not a single massive data base. The data is machine readable, and the pointers are machine-readable. The goal is an ever expanding network of data that machines understand and can hence search on our behalf knowingly.

References

Baudrillard, Jean *Truth or Radicality? The Future of Architecture.* Blueprint, January 1999

Balcomb, *D Energy 10* in Proc 1997 IBPSA Conf, Prague

Burners-Lee, Tim *The Semantic Web* http://www.w3.org:80/DesignIssues/Semantic September 1998

Burners-Lee, Tim *Principles of Design* http://www.w3.org:80/DesignIssues/Principles September 1999a.

"The World Wide Web is the universe of network-accessible information, an embodiment of human knowledge." Tim Burners-Lee quoted in www.whatis.com/www.htm November, 1999c

Berners-Lee, Tim *Weaving the Web* HarperCollins, New York 1999d.

Clark, Roger H. and Michael Pause *Precedents in Architecture* John Wiley & Sons; (1996)

Crawley, Drury, Curtis O. Pedersen, Richard J. Liesen, Daniel E. Fisher, Richard K. Strand and Russell D. Taylor, Linda K. Lawrie, Frederick C. Winkelmann, W. F. Buhl, A. Ender Erdem, and Y. Joe Huang *Energyplus, a New-generation Building Energy Simulation Program*", Proc Sixth Intl IBPSA Conference, Kyoto, 1999.

Crawley, Drury, Curtis O. Pedersen, Linda K. Lawrie, Frederick C. Winkelmann Energyplus: New Capabilities In A Whole-Building Energy Simulation Program Proc Seventh Intl IBPSA Conference, Rio de Janeiro, 2001

Donn, Michael Quality Assurance - Simulation And The Real World Proc Sixth Intl IBPSA Conference, Kyoto, 1999

Donn, Michael, Robert Amor and David Harrison *A design proposal for an internet based Simulation Quality Control Tool* Proc Seventh Intl IBPSA Conference, Rio de Janeiro, 2001

Gordon, Harry, Min Kantrowitz and Justin Estoque. *Commercial Building Design. Integrating Climate, Comfort and Cost* van Nostrand Reinhold, Co., Inc. New York, 1987.

Hand, Jon, John Clarke and Paul Strachan *Deployment of Simulation within Design Practice* Proc. Sixth IBPSA Conf. Kyoto, 1999.

Jackson, Anthony. Reconstructing Architecture for the Twenty-First Century Univ of Toronto Press, 1995. *"It is a telling commentary on the current situation that architects must now be convinced that it is no mean achievement to design buildings that function well, and that allow people to carry on their social life in a practical way".*

Lassila, Ora and Ralph R. Swick *Resource Description Framework (RDF) Model and Syntax Specification* Specification http://www.w3.org/TR/WD-rdf-syntax/ *January 1999.*

McCullough, Malcolm. *Abstracting craft : the practiced digital hand* MIT Press, Cambridge, Mass. 1996.

Moore, Fuller. *Concepts and Practice of Architectural Daylighting*. Van Nostrand Reinhold, New York, p 141, 1985.

Negroponte, Nicholas. *Being Digital.* Alfred A Knopf Inc. New York, 1995.

Papamichael, Konstantinos John La Porta & Hannah Chauvet *Decision Making Through Use of Interoperable Simulation Software* Proc. IBPSA Conf., Prague 1997

Péclet, E. Traité de la Chaleur. (3rd Edition.) Massone, Paris, 1861. Cited in Building Services Engineering - a review of its development, Neville S. Billington, Pergamon Press, Oxford, Ch. 13 pp482-526, 1982.

Schön, Donald. The Reflective Practitioner. Basic Books, Inc., New York, 1983.

Tóth, Gábor J.and Paul Mayer 'black box' <jargon> *An abstraction of a device or system in which only its extenally visible behaviour is considered and not its implementation or 'inner workings'.* From: *Free On-line Dictionary of Computing* by, http://wfn-shop.Princeton.EDU/foldoc/ , as at June 1 1998.

Turing, Alan M. *The Turing Test was introduced by Alan M. Turing (1912-1954) as "the imitation game" in his 1950 article (now available online at* http://www.sscf.ucsb.edu/~sung/comm115/writing-define-computing/Computing-machinery.html *) Computing Machinery and Intelligence (Mind, Vol. 59, No. 236, pp. 433-460) which he so boldly began by the following sentence: I propose to consider the question 'Can machines think?' This should begin with definitions of the meaning of the terms 'machine' and 'think.' Turing Test is meant to determine if a computer program has intelligence.*

Waldram, P.J. and J.M. Waldram. Window design and the measurement and predetermination of daylight illumination. The Illuminating Engineer, April-May, 1923. Cited by Dean Hawkes in A History of models of the environment in buildings, Working paper 34 of the Centre for Land Use and Built Form Studies, University of Cambridge, 1970.