The concept of Intelligent Simulation Environment

Dr. Roger Pelletret, Dr. Souheil Soubra,
Werner Keilholz, Zakaria Belghali
Centre Scientifique et Technique du Bâtiment
Service Informatique et Bâtiment
Sophia Antipolis, FRANCE

Abstract
The aim of this paper is to describe the work- currently under way at CSTB in the frame of the project 'Intelligent Simulation Environments'. The paper presents the underlying concepts of this project and then discusses the main features of an ISE (data sharing, coupling of simulation tools, reasoning facilities, model documentation and graphical front-end). Special emphasis is put on the model documentation with a proposal of a Standard Model Documentation which would combine the advantages of the existing "Proforma" and Neutral Model Format. The application of the ISE concepty to existing simulation codes is then illustrated.

Introduction

The need for integrated tools is growing among the building practitioners. This is the case for the designers as well as for the owners, the project managers, the constructors, the operators or the property managers.

There are many aspects in which it is possible to improve the professional practice, from the design process to the construction and the operation. The main one is to share the data describing the building and its equipment (for instance, HVAC systems); this data sharing will save a lot of time and insure a great consistency among the data used for various purposes. To enable the data sharing, an Integrated Data Model of the building has to be developed together with specialised data models which will add to the Integrated Data Model specific information about the structure, the HVAC system, the lighting, etc., related to the building.

CSTB
BP 209
06 904 Sophia Antipolis Cedex
FRANCE
Tel.: + 33 93 956 711
Fax: +33 93 956 733
e-mail: roger@cstbsop.sophia.cstb.fr

The CSTB's R&D project, entitled "Intelligent Simulation Environments", contributes to this wide R&D program. It is important to understand that the ISE research program is linked to other research programs. For instance, the development of the building Integrated Data Model is a task parallel to the ISE task.

The second main aspect to improve the professional practice is to facilitate access to the Integrated Data Model and to various tools (experts systems, diagnosis tools, design tools, documentation access tools, etc.) that can provide valuable information; among these tools, simulation codes are of primary importance, especially in the design phase.

The ISE program [Pelletret 90.06, Soubra 91.11] aims to facilitate access to the simulation tools used for building performance evaluation; this covers various fields: structure analysis, thermal analysis, acoustics, lighting, ventilation systems, etc. This R&D program started with the idea that simulation will be more and more used by the consulting engineers in order to better design and operate the buildings; but, for that, they need powerful simulation tools, as easy to use as simplified design methods. These powerful simulation tools already exist or are under development in the research laboratories. But their use is very often difficult and limited to specialists. The ISE project aims to develop a simulation environment, called IISIBât, which brings answers to most of the problems related to the use of advanced simulation tools. In other words, this ISE project will help in transferring technologies from the research labs to the consulting engineers.

ISE main features

On the first glance, IISIBât might look like just another attempt to build a graphical front-end around existing Building Performance Evaluators. It is indeed a major goal of the ISE to provide a sophisticated graphical environment to allow the user to enter information in a straightforward way, instead of dealing with 'macro-languages' or other highly confusing forms of input files needed by common building simulation tools. So, the ISE is also a graphical front-end. But is more.

The ISE addresses three kinds of users:
• the models developers;
• the projects developers;
• the analysts.

The models developers are those who invent and produce new components stored in model libraries; the project developers are those who model real buildings, using for that the components; the analysts are those who make the economical and technical studies (i.e. parametric analyses).

For each of these users, the ISE proposes specialised simulation environments in order to increase the productivity of their work.

For the model developers, object oriented libraries, automatic generation of code and embedded documentation of models are the basic features that are needed; for the model documentation, we use at the moment the "Proforma". Information stored in the "Proforma" files can be used to build knowledge bases; these knowledge bases are then processed by expert systems to help the project developers: the role of the project developers is to create project libraries; to do so they have to link components together in order to develop networks which represent the building from the desired "point of view" (structure, HVAC system, etc.); knowledge bases are developed to propose, if any, already defined networks which fit the main requirements or to help in connecting the models (compatibility rules) or to help in connecting the variables or to help in choosing the parameters values, etc. Other functions are developed to assist the analysts when performing the simulations; the most interesting one is the optimisation function.

Furthermore, procedures are under development to make automatic requests onto the Integrated Data Model in order to retrieve all the possible information [Soubra 92.07]. Functions are under development to make possible the coupling of various simulation tools in order to enable, for instance, building energy performance and related pollutant transfer evaluations. There are also embedded functions to help the users whatever their objectives are.

We describe hereafter, into detail, the main features of the ISE and, in particular, we focus on the front-end for the project developers; this front-end is based on the principle of assembling block-schemes. But we think that, in the future, semantic CAD tools, built with an object oriented approach could be widely used to describe the building and its equipment and then to evaluate the building performances. When semantic CAD tools will be available, the way of working of the project developers will radically change: rather than to describe an assembly they will draw the building and its equipment, an expert system will infer the corresponding networks in order to run simulations; to reach this goal, new knowledge bases will have be developed.

Data Sharing

The ISE will enable simulation tools to communicate on different levels (sharing data and sharing part of assemblies).

Sharing data is the simplest way of communication. The starting point for this concept is the trivial observation that all the simulation tools involved operate the same 'object': a building.

It is an obvious demand of users who apply a set of tools that information commonly needed by several tools should not have to be entered more than once. This requirement is met by the Integrated Data Model (IDM) [Dubois 92.01]. The Integrated Data Model is a description of a building at such a high level of abstraction that it can be used as a common base for all Building Simulation Evaluators that are operated through the ISE; that means, among others, that the sum of all fields of all classes of its internal object oriented representation contains all information ever needed by any of the tools. Adding new tools to the ISE means, among others, adding new fields and/or classes to the Integrated Data Model used by the ISE.

The sharing of common data between various simulation tools must take place through an integrated data model of the building to insure efficiency and consistency of the data used. The integrated data model comprises all data that is needed for a complete description of the building in its different stages and hence supports the extraction of all kinds of different views. Conceptually, the integrated data model can be regarded as the central and common core of different clients. No direct links between clients are allowed.

The design of an integrated data model is a task parallel to the ISE development; a first attempt to develop such a model (energy oriented) has been done in an EC-funded R&D project (COMBINE 1 research project -JOULE programme of the CEC Directory-General XII-)..

The IDM, where energy aspects of buildings design have been given priority, reflecting the project's origins in the JOULE programme, could be replaced at a later phase by a domain independent data model.

Data sharing could ideally allow to translate an assembly of models describing one aspect of a system (such as thermal aspects of a building described in terms of TRNSYS[1] models) into an assembly describing another aspect of the same system (e.g. a description of the same building in terms of COMIS[2] objects). This problem is non-trivial. In the case of some simple, straightforward assemblies, an automatic translation of both the models and the logical links between them may be feasible. In most cases, however, the best we can hope for will be a partially automated translation: many models used in one view of a given problem (e.g. a solar collector in the thermal model of a TRNSYS point of view) do not have counterparts in other views (e.g., no solar collectors in COMIS). In such cases, the assembly yielded by the translation will be a rather sparse network of models because of missing equivalencies between the components involved. Missing components will have to be transferred manually, and missing links will have to be defined.

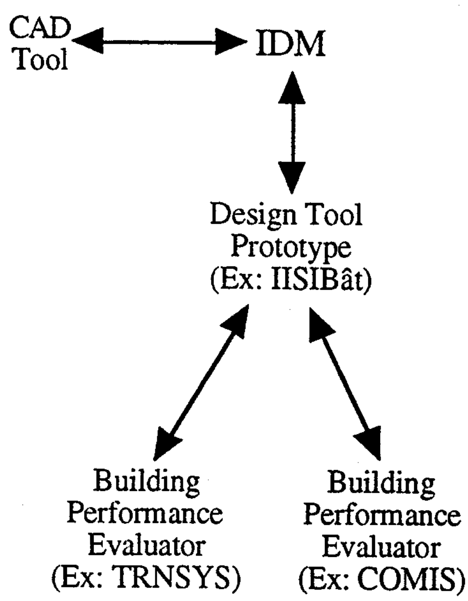


Fig.1: Sharing data via the Integrated Data Model (general principle).

Coupling of Simulation Tools.

Data sharing will also allow coupling of various simulation tools. In general, the coupling of simulations tools aims to enlarge the range of possible applications and to increase the accuracy of the results. For instance, the coupling of an Air Flow simulation code with a Building Energy Performance simulation code will enable accurate performance assessment of active

[1] TRNSYS is a thermal simulation code developed by the University of Wisconsin - Madison.

[2] COMIS is Multizone Air Flow and Pollutant Transport model developed in the frame of Annex 23 of IEA.

air heating and cooling systems (i.e. calculation of the heat gains and losses of the building and of the cooling or heating power provided by the HVAC system), passive cooling (i.e. determination of the energy released by passive cooling during the night), heat transport between zones (i.e. calculation of the heat transported by air circulating between zones: accurate computations are useful for passive solar buildings, where excess heat in irradiated part should be brought to the shaded part of the building by natural convection), ventilation heat losses, etc.

Different coupling strategies could be considered as for instance :

- Sequential coupling is the most straightforward and also most time-efficient coupling method. It simply consists of invoking a first simulation tool, yielding a huge matrix where one dimension is time. This result is fed into another simulation run, invoking a second simulation tool. This coupling method, too crude for many problems, is sometimes sufficient.

- 'Ping-pong coupling' consists of alternately starting two simulation tools, each using variables calculated by the other. A supervisor should be built in the ISE; the supervisor should generate a time-loop and alternately starts a one-step simulation run with the two codes involved. Each tool solves its own problem with its own solving method. At each time step, a decision has to be made, whether or not the simulation converges. In this method, the supervisor compares, at each time step, the results of iteration i with those of iteration i-1 and decides whether or not the results are valid. The advantage of this method is that when a generic coupling environment exists, various codes can be coupled without having to rewrite specific subroutines.

- Another method should yield the most accurate results; it consists of extracting the knowledge contained in various Building Performance Evaluators and translating this physical knowledge into models of a generic solver (domain independent). This method might be pursued in future versions of the ISE using either SPARK (developed at the Lawrence Berkeley Laboratory -USA-), or ESACAP (developed by the European Space Agency), or ZOOM (developed by the RAMSES group of CNRS -France-).

Reasoning facilities

Expert rules can be used by the ISE to assist the user in the process of modelling/simulating complex systems. The user generally has a good idea about his simulation goal. He begins by searching for an existing project that might fit his goals with minor modifications. If he finds one, he goes directly to the phase of parametric study. If not, he will choose existing components in the libraries in order to link them in an assembly. He will perform afterwards a parametric study consisting on giving to the parameters of the assembly some values in order to run the simulation. A parametric study is an iterating process : the user might change the values of the parameters in relation with the results of the previous simulation.

Therefore, expert rules aiming at assisting the user consist of :

- Simulation goal rules
  Starting from simulation goals specified by the user, the system could ease the assembly of models in several ways:

  - by trying to find in the project library an assembly that fits the posted simulation goal. A list of possible criteria of the search (e.g. type of building, HVAC system description...) will be supplied. If found, this project will be a starting point for the user's parametric study;

  - by not allowing to use a certain number of models which are not in line with the simulation goal;

  - by helping the user to choose modules adapted to his simulation goal;

  - by assisting in the choice of the solving method for the numerical problems (e.g., the system could provide answers to questions like "Is sequential coupling sufficient in the given assembly / simulation goal combination ? Which solver should be used ?").

- Input data rules
  Expert knowledge could be used to help the user in choosing adapted values considering the simulation goal and the default values stored in the Model Documentation File.

- Linking rules
  The goal is to assist the user in correctly linking models and choosing models that can be linked to a given model. This information should be taken from the Model Documentation File attached to each model. Linking rules can be considered on various levels. The simplest approach would be two lists of models attached to each model, one consisting of possible precursors (i.e., upstream models) and the other one listing possible successors (this information could be generated by an inference engine from the lists of upstream models). Furthermore, rules for linking variables (for instance, in the TRNSYS style) between models could be automated once a link between two models is established.

- Consistency checking
  An expert system could apply rules to a given assembly which are to ensure the consistency of the links the user defined between models. This would go beyond the simple rules used to check if a link between two given models is allowed. More complex rules, taking several hops in a network (or the entire network) into consideration could exist.

Model documentation

For the model documentation, IISIBât uses the "Proforma".

The "Proforma" is a standard way of documenting models; the idea was born in 1984 from discussions between Louis Laret (University of Liège -B-) and Joe Clarke (ABACUS group, Strathclyde University -Glasgow, Scotland-). Then a first version of Proforma, defined with CSTB, has been applied in Annex 10 of IEA. A "Proforma" club is now established in France and the concept of "Proforma" has been shared with other labs in Europe and used in Annex 17, 21 and 23 of IEA. The main aims of "Proforma" were to document models, to ease the sharing of models between researchers, to ease the translation of models into subroutines.

On the other hand, Axel Bring, Per Sahlin (Royal Institute of Technology, Sweden) and Ed. Sowell (California State University) proposed in 1989 at the first BS'conference the concept of Neutral Model Format (NMF) [Sahlin 89.06]. The basic objective of the NMF is to provide a common format of model expression for various simulation environments (i.e. independently of the Simulation Environments). The target was to be able to automate the translation of a set of equations into a library component for these various Simulation Environments.

To summarize, the "Proforma" group put efforts into the model documentation (not only the way of writing the equations but also the way of defining all what is needed to understand the limits and the domain of application of a model), the NMF group put effort on defining a syntax which ease the development of automatic translators. This summary is for sure a little bit short but I think it focuses on the respective advantages of both formats.

In the future we propose to use a Standard Model Documentation (SMD) which will combine the advantages of the Proforma and of the Neutral Model Format. In the frame of an ISE, the information stored in the SMD will be used to built knowledge bases; these knowledge bases will then be processed by expert systems to help, mainly, the project developers. Moreover automatic translators could be incorporated in ISEs to enable model developers to easily increase the model libraries.

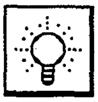
The graphical front-end

A brief demonstration on computer would be better than a long description. Nevertheless it is possible in few words (and with the help of figure 2) to give an overview of the human/machine dialogue through IISIBât.

The graphical front-end can be relatively easily modified due to the way of programming; so the front-end does not look the same for the models developers or for the projects developers or for the analysts. More easy is the work to do (i.e. analysts), more simple is the front-end.

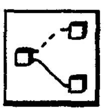
For the model developers, IISIBât is composed of three main windows: a model library, an assembly window and an account operating window.

Various tools are attached to each window. For instance, in a model library it is possible to create new models with the following tool:



To create a new model, the user has to define all the information needed by IISIBât (e.g. in the case of TRNSYS, the list of inputs, parameters, outputs and derivatives with all their attributes). This information is entered in a Proforma file. The code (i.e. Fortran subroutine, C file, etc.) corresponding to the new model has to be entered independently and till now we did not developed help functions for this purpose.

An other important tool of the model library is the following:



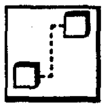
This tool enables the user to create new classes in an object oriented tree, where the models are stored as leaves. Till now we do not take advantage of the way of storing the models, but the idea is that it should be possible to attach information to classes and then, when a model will be stored as a son of a class, it will inherit of all the methods and attributes defined at the class level.

The main tools attached to the assembly window are:

- 
  multifonctionnal tool enabling the user to select and to create new instances of components on the assembly window, to define the parameters values, to check the inputs or the outputs, to move the component on the window (by dragging it), etc.

- 

  to get information about the model (i.e. to open a "Proforma" file);

- 
  to draw links between two models and to specify the contents of the links (i.e. to connect outputs to inputs);

- 

  to define macro-models (i.e. components gathered to be used as a single component);

- 
  a tool to run the simulation (for instance, automatic writing of a COMIS Input File and run of the program).

From the user's point of view, the most interesting part of a simulation is of course the results. IISIBât involves a graphical tool to draw curves from the output files.

From a general point of view, the standard working procedure is to run a sequence of simulations, varying a couple of parameters to reach a desired configuration. Functions to ease sensitivity analysis could be involved (for instance, a function could take in charge the preparation of the sets of parameters needed to do an optimization; for that the concept of Multi-run Interface for Sensitivity Analysis developed in the frame of Annex 23 of IEA will be reused).

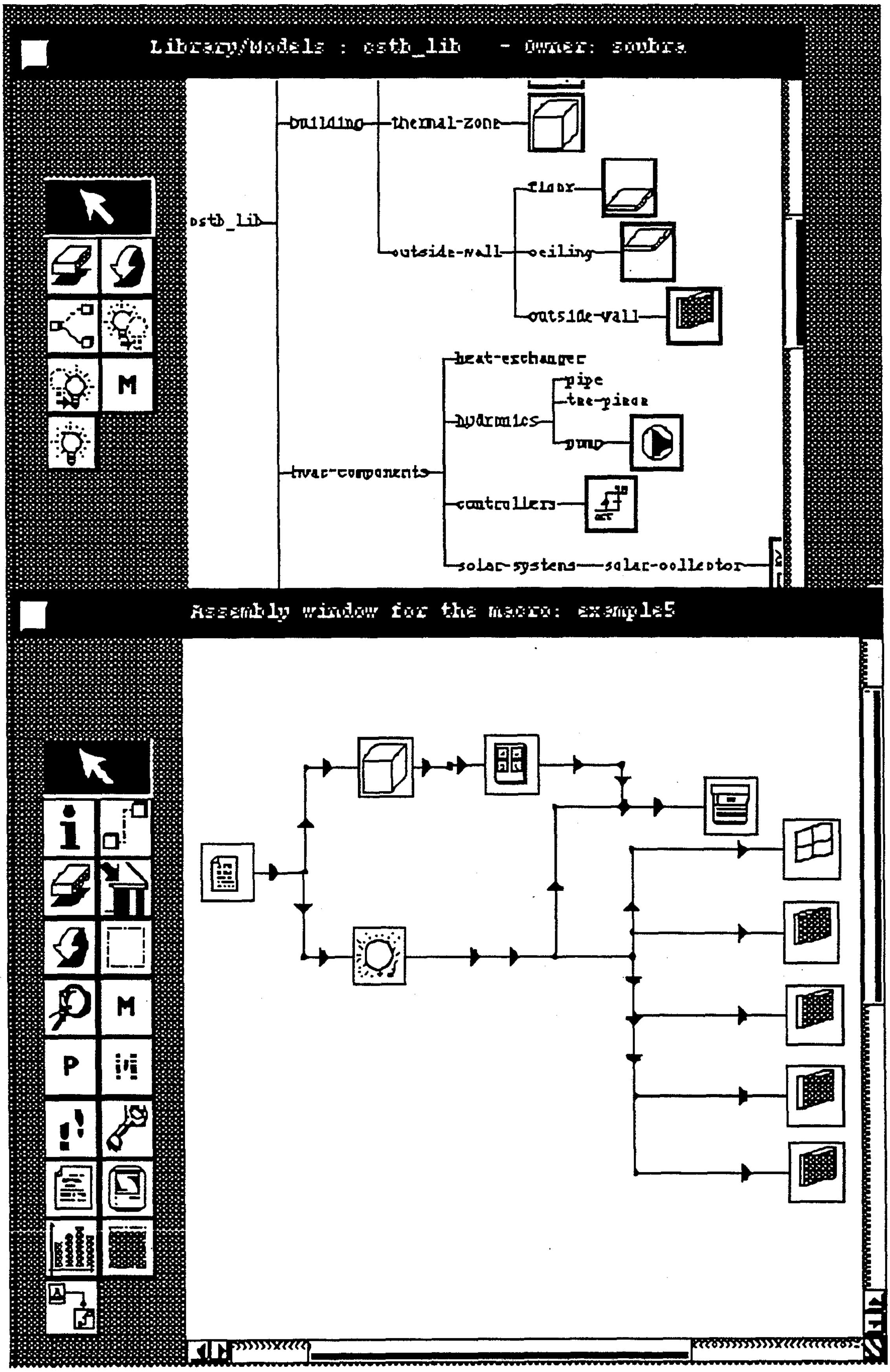


Figure 2: The graphical front-end.

Implementation issues

From the computing point of view, the development of the ISE is based on an object oriented approach: we use Le_Lisp[1] which includes the object oriented Language MicroCeyx[1] and the graphic interface generator Aïda[1].

The description of the conceptual model is done using the HBDS formalism.

A specific ISE is an instance of an internal object oriented model of a generic ISE. This feature is very important for the ISE developers to increase their productivity.

The ISE runs on machines with UNIX as Operating System and X11 as window manager; it runs also on PC/DOS under Windows; it needs Le_Lisp and Aïda run-time licences.

Conclusion

The ISE project aims to help in transferring technologies (i.e. simulation codes) from research labs to consulting engineers.

CSTB develops an ISE called IISIBât. IISIBât is still evolving as the concepts themselves evolve and as the already defined concepts are implemented.

Till now, the ISE concepts have been applied to some existing simulation tools and new projects are underway.

The first application of IISIBât has been to develop an ISE for TRNSYS. From there, an ISE for CA-SIS[3] has been developed, this software is being tested in a consulting engineering office. An other application is developed in the frame of Annex 23 of International Energy Agency ; this application is intended to encapsulate the COMIS software; it is also intended to test the feasibility of a 'ping-pong' coupling between COMIS and TRNSYS. Other applications are being initiated : SPARK (developed at the Lawrence Berkeley Laboratory -USA-) [Nataf 92.09], ESACAP (developed by the European Space Agency), or ZOOM (developed by the RAMSES group of CNRS -France-).

[3] CA-SIS is a software developed by French Electricity (Electricité de France) to compute the thermal performances of active cooling systems. CA-SIS uses TRNSYS solver and syntax.

References

[Dubois 92.01]
COMBINE IDM / V3.3
Anne-Marie Dubois & al.,
January 1992
CSTB's report: MGL/92-1264

[Nataf 92.09]
Automatic Code Generation in SPARK. : Applications of Computer Algebra and Compiler-Compilers.
Nataf J-M. and Winkelmann F.
Research Report.
Simulation Research Group.
Lawrence Berkeley Laboratory.

[Pelletret 90.06]
The IISIBât project :
Intelligent Front-End Shell for Building Simulation Codes.
R. Pelletret & S. Soubra
2nd French-Finnish Seminar.
Espoo. Finland. June 1990.

[Sahlin 89.06]
A Neutral Model Format for Systems in Building Energy Simulation
P. Sahlin, E. Sowell
Building Simulation'89, IBPSA.
Vancouver, Canada. June 1989.

[Soubra 91.11]
Intelligent Simulation Environments for Building Modelling and Simulation.
S. Soubra & R. Pelletret
EuropIA'91.
Athens. Greece. November 1991.

[Soubra 92.07]
Concepts et méthodes pour le développement d'Environnements de Simulation Intelligents.
Ph-D thesis.
Ecole Nationale des Ponts et Chaussées.
Juillet 1992