FORGET THE TOOL WHEN TRAINING NEW SIMULATION USERS

Jon W. Hand
University of Strathclyde
Glasgow G1 1XJ Scotland
Jon@esru.strath.ac.uk

Drury B. Crawley
U. S. Department of Energy
Washington, DC 20585 USA
Drury.Crawley@ee.doe.gov

ABSTRACT

Training new users of simulation programs typically focuses on a single tool–specific techniques for interacting with, creating models, and assessing performance. Unfortunately, this tends to produce users limited by a particular tool's capabilities—not users that can easily decide how best to tackle a simulation problem, regardless of tool. Additionally, tool-specific introductions to simulation tend to encourage novices to be 'tool driven'—everything begins to look like a nail when your only tool is a hammer.

This paper argues for a tool-independent approach to training first-time simulation program users. A suggested curriculum is presented for generalized simulation techniques and assessment methods independent of any tool. How this approach has worked with graduate students and researchers with and without prior simulation experience is also discussed.

INTRODUCTION

Learning how to interact with a tool can get in the way of understanding basic simulation techniques—how best to represent, for a particular problem, the essential building characteristics and relationships between zones and plant components. Learning simulation methods based on the capabilities of a specific tool then drives how users create simulation models and assess performance. This may not be appropriate for the needs of a design process that relies on working with a palette of tools and a mix of design and assessment skills.

By learning the fundamentals skills of simulation before starting with a particular tool, novices can better understand the link between what they see on the screen and the descriptive entities and relationships that they must, ultimately, control. With tool-independent training, how a particular program works (syntax) becomes secondary to learning the (universal) semantics of modeling.

PROBLEMS WITH TOOL-CENTRIC APPROACHES

Most training programs for new simulation users focus on tool interaction and production techniques. This excludes the most important aspects of simulation—understanding how to abstract a building model that represents the important and appropriate characteristics for the performance assessment methods chosen in the particular project. After all, simulation is "to assume a form resembling that of something else" (OUP 1971)—in this case a building—not to create an electronic duplicate of the building down to the finest detail. It does involve abstraction and simplification—it is not learning how to force a building into a tool.

Without such direction, the use of simulation is often far from optimal, either delayed by overly complex models or lacking credibility by the use of inappropriate simplifications. Those constrained by the capabilities of one tool are not well placed to select appropriate tools for projects, which increasingly rely on the cooperative working of various assessment tools.

How does it come about that initial training tends to exclude potentially important aspects of simulation work? Perhaps software developers, vendors, and trainers:

a) view those issues as irrelevant to their tools,

b) assume that such topics detract from the acquisition of keyboard skills,

c) assume that those being trained already possess an adequate background,

d) recognize the difficulty in presenting such topics to groups of mixed backgrounds and interests,

e) learned simulation through use of a single tool and are still limited by that tool, or

f) focused on a specific component or system rather than the underlying simulation techniques.

Simulation ease-of-use is usually gained by constraining the descriptive process (simplified methods) or through removing the tedium associated with a difficult task (graphic data entry of geometry). But ease-of-use does not alter the difficulty of understanding the complex thermophysical processes and interactions within building and environmental control systems. It is not a trivial task to decide if a design is well represented by two rectilinear zones or if a default gridding scheme is appropriate for a computational fluid dynamics (CFD) problem. Those with a tool-centric introduction to simulation are not well equipped to answer such questions. Their subsequent misapplication of a tool will be frustrating for them and hardly in the interest of either their client or the tool vendor.

Because simulation tool interfaces today increasingly look like computer-aided drafting (CAD) tools, novices are inclined to judge them in a visual context. Including metal roof system profiles or doorframes is rarely justified in a thermal context. If users appreciate that simulation models are composed of descriptive entities related to site context, thermal zones, surfaces, controls schemes and schedules of occupancy and the like, then the information content of the display is more likely to be recognized.

Further, consider that novices often acquire interaction (keyboard) skills without appreciating what it is they see on the screen or the implications of the choices they are being asked to make. Thus, they rely on syntactic information (which is difficult to acquire and remember) rather than drawing on semantic knowledge of the underlying entities and interactions. For example, once a novice learns that ground temperatures are associated with site conditions and apply to below-grade surfaces, they can direct their search for relevant descriptive facilities regardless of the interface. Delaying the introduction of keyboard skills is thus a useful technique.

The assumption that those being trained need only to acquire production skills (and that prerequisite skills can be discounted) is also suspect. Even those who have been active in assessment tasks may not appreciate that the marginal costs of incremental refinements to a model is low or that a range of issues can be covered quickly if the initial model is well planned and designed. Practitioners used to traditional engineering practice may find that moving to simulation is a bit like acquiring a new Swiss Army knife. "So many tools. Which are most appropriate for my current needs?"

As tool vendors lower the entry requirements via ease-of-use, they also increase the complexity of their tools and support more integrated assessments. There are several implications: novices may not have the skills to judge which facilities ought to be used and domain experts must broaden their skills to 'adjacent' domains or learn to work with other experts to successfully support integrated assessments. Clearly, mastering such skills may be out with the remit of initial training. However, the implications of this can be covered in multi-tool integration training.

CLUES FROM THOSE WITH PRIOR EXPERIENCE

Those who have prior experience in simulation and who are exploring a new tool are often quite adept at learning a new tool (with little instructor intervention). More importantly, experienced users often display considerable wisdom in use of an unfamiliar tool. The authors have observed that such users:

- begin by asking questions which allow them to confirm how particular processes are treated and how facets of a design are represented (semantics),
- tend to discount syntactic knowledge and the particular form of interaction (unless it gets in their way),
- are skeptical of predictions, their initial models are usually created to confirm the response of the tool to particular assessments, and
- do not rely on a single simulation but on a series to test the sensitivity of design features and to understand performance.

These attributes of experienced users are important to an initial training regime. Novices tend to accept initial predictions and have little or no concept of model calibration or the need to confirm predictions based on results at a finer level of detail. Initial training can and should point out the place of calibration, quality assurance, and uncertainty analysis.

Experienced users spend considerable time planning their simulations, starting with a sketch of the geometry of their models (and likely future variants) in order to identify where detail is required, where parametric variations may be applied or where systems and flow networks will be attached. Novices inevitably rush to the keyboard, compose models on the fly, and just as often paint themselves into corners with ill-considered complexity.

The evolution of tools to support the rapid creation of models may be problematic for the tool-led. The authors are aware of many cases where a user has insisted on simulating the minute details of complex geometry that have little to do with the system evaluation intended. Training can and should address this issue so that when users explore tools they appreciate the value of reasonable abstraction. No doubt, some developers and analysts will take exception to this. After all they have spent tens, if not hundreds of thousands of dollars in the name of ease of use and will be tempted to confuse this with enabling users to make appropriate use of their tools.

Vendors and developers often include example problems with their distribution disks or as part of their training. But novices don't really know what to look for when they browse such problems or what is required to appreciate what an example problem represents. Clearly, knowing more about the semantics of tools will ease the chaos that confronts first time users. To remind the novice about this, an introductory course should supply a checklist. For instance:

- Start with the context of the problem. Where is it located, what are the attributes related to site that you could find in the tool? Are such attributes held together or multiple displays?
- Look in the model for documentation. If the author of the model has included a description of what the model is trying to represent then see how much of the explanation you understand. Any confusion you might have should be the subject of questions to the instructor.
- Next look at the geometry of the problem. What is the relationship between what you see and the "building blocks" of models introduced in the course? How has the building been subdivided into zones? Do these zones match rooms or floors in the building?
- With this as a background, run simulations for a few days in winter and summer and look at the predictions. Are the patterns of room temperatures or heating demands as expected? If not, how might you go looking for further information or clarification? Are there alternative ways of looking at performance, different levels of detail or even different performance criteria?

Such a progression of questions and checklists can remind the novice to occasionally shift attention away from rote learning of keystrokes to allow them to begin sorting out the semantics underlying the tool. It also puts pressure on tool vendors to improve the documentation of their example problems and their training staff's ability to support questions not related to keyboard skills.

Many perceive the planning of simulation tasks, the abstraction of a design into a model and decisions as to the nature of the assessments required are inappropriate outside the context of a specific tool. However, consider that the *CIBSE Applications Manual: Building Energy and Environmental Modeling* (Bartholomew et al. 1997) lists these as necessary skills: selecting assessment tools, determining appropriate assessment metrics, and proper application of the appropriate tools. The assumption that such skills are only necessary as practitioners attempt to go beyond routine assessments is thus called into question. An introductory course should confirm the importance of planning the nature of the model, assessment tasks and metrics.

It is possible to view the skills associated with abstracting the essential characteristics of a design into a model as generic rather than exclusively specific to the tool. It should be reasonable to ask the user to sketch out their model—its zones, general level of detail, schedules of use and relationship to site conditions before they sit down at a keyboard. If they cannot do this then it may not yet be time for this person to attend a vendor's course (and get full value). They are certainly not at a point where they can compose a model even if they are comfortable with the interface of a particular tool.

This may seem a harsh analysis, but consider that it may indicate that the potential user does not yet either understand enough of the problem to describe the overall characteristics of the design, or grasped the concepts associated with the basic "building blocks" of assessment tools.

Experience shows that those who are opinionated about what it is they want to accomplish are more adept at using (driving) their tools efficiently. Otherwise, they are easily tool-led and not in a position to work within the constraints of a project.

SUGGESTED CURRICULUM

The authors believe that a course of training for simulation users should consist of three major parts: introduction to simulation, applying simulation in design and retrofit, and tool-based training. Each is described in more detail below.

Introduction to Simulation

In addition to the attributes of the expert user described above, other elements that should be included in an introduction to simulation include:

- Classes of assessment tools and aspects of reality which they are able to deal with.
- General relationships between physical aspects of buildings and systems and their embodiment as "product models" within tools.
- Types of relationships between entities (i.e., between site parameters and climatic data).
- How simulation tool interfaces differ from CAD tools.
- How to recognize the underlying product model in interface elements.

Applying Simulation in Design and Retrofit

Probably the most significant issue facing a new user is when and how best to apply simulation in design. What level of detail is appropriate at what time in design and for which issues? Thus, this part of the training is as important as the tool-specific training. Particular areas that should be covered include:

- How design issues are translated into simulation tasks. Include aspects of the levels of detail needed to address particular issues—peak demands may be resolved from coarse descriptions while the prediction of local comfort or air flow patterns requires another level of detail.
- Use of assessment tools at each stage of the design process—constrained tools for quick explorations and more robust tools for

exploration of detailed assessments (and when to shift from one to another).

- Types of decision which assessment tools pose to users and techniques for getting data to support such choices.
- Limitations of tools and how to recognize and cope with such limitations. Sub-topics might be sources of advice when a project demands predictions of the interaction between two analysis domains, such as fabric flux and airflow, when these are not linked in a particular tool. Conversely, tools may offer facilities that only experts are in a position to use, such as CFD or fuzzy logic controls, and so training should include guidelines on when such facilities should be invoked and what level of expertise is demanded.
- Types of predictive information available in tools, and what analysis and pattern matching skills are required to understand such information and make use of it in a design project.
- Levels of abstraction and the criteria needed to select appropriate detail of description and analysis to answer questions within the limited time scale imposed by clients.

Tool-Based Training

Although not always possible, more than one tool should be used in simulation training courses. This provides users with perspective by demonstrating different approaches and usefulness. Also, an introductory course ought to equip prospective users with some of the skills needed to guide their selection and or appraisal of tools. For instance, tools are often selected on the basis of the mix of facilities they offer and the level of complexity they can deal with. Training should use exemplars of a range of design questions to show:

- Level of model detail needed to answer particular assessments.
- Types of facilities necessary to support particular assessments.
- Whether an assessment is a single issue or might be better approached by using an integrated tool.
- What it means to translate design questions into modeling tasks. For example, if the design question is “does this building require air conditioning?” Then the modeling question might be “what will be the peak summertime temperatures and their frequency of occurrence with a naturally ventilated scheme?” Such a modeling question includes the metrics by which a decision can be taken (i.e., number of overheating hours), as well as providing evidence needed to help select a reasonable type of air conditioning and the viability of an alternative design.

To guard against information overload, such concepts and relationships should be made available as course notes. The tool-specific course should make available a synopsis of these relationships as a reference. Also, tool-specific training should openly discuss the capabilities and the limitations of the tool. A tool that supports one-dimensional conduction will only approximate edge effects and thermal bridges. A tool that does not maintain moisture balances will only approximate the impact of latent loads and dehumidification processes. Thus, users should be made aware of the appropriate applications for each tool.

Much of this discussion has application to both managers of assessment groups and their staff. The authors do not advocate that potential users necessarily are required to understand the physics underlying such topics. Rather it is important to be aware of issues out with the specifics of the tool and the often-narrow focus of a vendor’s training program.

But this general curriculum must become specific if it is to address the needs of introductory and tool-specific training. Just as it is difficult for novices to acquire new skills, it is difficult for an observer to observe the progress of participants to ensure that instructional media and exercises are appropriate, and the problematic aspects of the tools.

JOURNALING TO IMPROVE TOOLS AND TRAINING

Over a number of years, the authors have observed the barriers new users face in acquiring skill in using a variety of assessment tools. In previous papers Hand (1995, 1993) reported the time taken to complete standard tasks and the information sources novices use, or prefer to access. Initially, this was based on questionnaires, but such forms of information gathering can be distracting and may not be uniformly applied.

Applying knowledge-based control to the cooperative use of tools is also predicated on assessing what a user is doing and what facilities are being accessed. In the COMBINE programme (Clarke et al. 1995, 1994), a blackboard metaphor was used to intercept messages from and between tools. This blackboard interaction was used to invoke subsequent tools or bring issues to the attention of the user. It was assumed that a journal of the actions and data exchange would prove useful for those who wish to understand the use of assessment tools.

In training, trying to understand how a user has arrived at an impasse or has corrupted a model are difficult because prior sequences of actions are unknown and novices are not necessarily adept at making use of support facilities. The journaling concepts introduced in COMBINE were recently introduced in esp-r. Esp-r now captures a journal of user progress, decision points, and access to tutorials and contextual help in order to quantify whether there are patterns related to recognizable user types.

Table 1 presents an extract from one such session. This journal indicates that the user:

> first scanned training exemplars,
>
> selected an exemplar,
>
> reviewed the summary and images of its composition and control,
>
> ran a simulation,
>
> looked at the predictions, and
>
> finally invoked a graphing tool before ending the session after 19 minutes.

Evidently, the information provided in the exemplar summary was sufficient for the user because they went on to undertake a test simulation without looking at further details of the model.

In the session shown in Table 2, the student composed a new problem, beginning with site definitions. Then proceeded to create a zone from an extruded floor plan and attributed some of the surfaces in the zone. After the third surface the user found it necessary to leave the zone definition and look at entries in two databases. The user then refocused on the zone and continued attributing surfaces. Finally, the user ended the session. In the next journal (or session) the user goes first to the databases and adds a new wall type and then makes use of this in the zone that had been created previously.

The user has assumed that information about wall compositions is only available by going to the database facility and the instructor could advise the user of a reporting option which makes this traverse unnecessary. The tool developer might look to see how often users jump back and forth between geometry and database editing in order to see if a shortcut is needed.

Table 3 summarizes actions from six student journals during a recent three-day workshop for a mixed group of building physicists and program managers ranging from novice to expert user.

It is clear that Students 1, 4, and 5 were able to perform many more actions during the period. Other students seemed to have stayed longer within particular activities—probably learning as they went. Students 1, 4, and 5 had prior experience with simulation tools and were learning how this particular tool worked more than learning its syntactical structure.

Such journal information, if available from a number of tools would benefit tool vendors in their own training and provide authors of introductory courses with issues which have proved problematic. Journal information from students involved in specific tool instruction and design professionals who attend workshops is already being used to upgrade introductory courses at the University of Strathclyde.

SUMMARY/CONCLUSIONS

There is a need for tool-independent simulation training for new (and many existing) simulation users in the profession. Training needs to teach users how to select the right tool for the problem, abstract the problem to the appropriate level of detail and metric, and correctly utilize the capabilities of the particular simulation tool. Once someone has been trained in the basics of simulation, then they should progress onto the appropriate use of multiple tools throughout a project and finally, tool-based training.

Table 1. Sample Student Journal

```
Journal for: student2
Date: Wed Feb 12 16:15:41 1997
PRJ: scanning exemplars enter @ Wed Feb 12 16:15:48 1997
PRJ: owning exemplar @ Wed Feb 12 16:16:00 1997
/export/home/student2/simple/cfg/bld_simple.cfg @ Wed Feb 12 16:16:00 1997
PRJ: current problem @ Wed Feb 12 16:16:02 1997
bld_simple.cfg @ Wed Feb 12 16:16:02 1997
HELP: zones domain summary @ Wed Feb 12 16:16:09 1997
HELP: control domain summary @ Wed Feb 12 16:17:24 1997
PRJ: enter uncertainty control @ Wed Feb 12 16:19:00 1997
PRJ: enter simulation controller @ Wed Feb 12 16:21:14 1997
PRJ: beginning simulation @ Wed Feb 12 16:21:17 1997
PRJ: enter assessment controller @ Wed Feb 12 16:30:48 1997
PRJ: beginning res @ Wed Feb 12 16:30:58 1997
PRJ: enter assessment controller @ Wed Feb 12 16:33:31 1997
PRJ: beginning graphing tool @ Wed Feb 12 16:34:29 1997
Finish project manager @ Wed Feb 12 16:34:46 1997
```

Table 2. Second Student Journal

Journal for:student1
Date: Wed Feb 12 16:40:33 1997
PRJ: beginning new problem @ Wed Feb 12 16:40:48 1997
HELP: new site @ Wed Feb 12 16:41:44 1997
PRJ: create configuration rideau @ Wed Feb 12 16:41:46 1997
PRJ: new zone @ Wed Feb 12 16:42:05 1997
HELP: extrud @ Wed Feb 12 16:42:13 1997
PRJ: focus on foyer @ Wed Feb 12 16:42:25 1997
PRJ: enter zone vertices @ Wed Feb 12 16:42:52 1997
PRJ: enter zone topology @ Wed Feb 12 16:42:59 1997
PRJ: enter single surface attribution @ Wed Feb 12 16:43:20 1997
PRJ: add def window @ Wed Feb 12 16:46:36 1997
PRJ: enter single surface attribution @ Wed Feb 12 16:49:47 1997
PRJ: enter zone topology @ Wed Feb 12 16:50:09 1997
PRJ: insert surface into another @ Wed Feb 12 16:50:32 1997
HELP: surface details @ Wed Feb 12 16:53:44 1997
PRJ: enter single surface attribution @ Wed Feb 12 16:54:39 1997
PRJ: enter single surface attribution @ Wed Feb 12 17:01:49 1997
PRJ: enter single surface attribution @ Wed Feb 12 17:02:21 1997
PRJ: db management enter @ Wed Feb 12 17:04:04 1997
PRJ: enter primitives @ Wed Feb 12 17:06:22 1997
PRJ: enter composites db @ Wed Feb 12 17:08:36 1997
PRJ: update problem configuration @ Wed Feb 12 17:11:46 1997
PRJ: db management exit @ Wed Feb 12 17:11:46 1997
PRJ: focus on main @ Wed Feb 12 17:15:09 1997
PRJ: enter single surface attribution @ Wed Feb 12 17:15:41 1997
PRJ: enter single surface attribution @ Wed Feb 12 17:15:49 1997
PRJ: add def window @ Wed Feb 12 17:51:14 1997
PRJ: update zone geometry @ Wed Feb 12 17:52:13 1997
Finish project manager @ Wed Feb 12 17:52:52 1997

Table 3. Actions from Six Student Journals

	Student					
Topic	**1**	**2**	**3**	**4**	**5**	**6**
False start	2	2	3	2	7	3
Navigation?	8	5	7	12	8	3
Dialog?	7	3	4	10	10	2
Splash?	16	6	10	18	16	6
New application?	11	4	11	16	9	1
Sessions	13	17	12	19	10	1
Browse exemplars	9	8	4	11	0	0
Browse database	6	7	5	3	2	1
Shift zone focus	34	23	4	32	38	3
Create zone (in-built)	3	4	1	0	0	7
Import from CAD	2	0	5	13	4	0
Surfaces attributed (singly)	52	21	4	63	68	10
Global attribution used	0	0	1	29	11	2
View/edit vertices	8	3	0	1	1	8
Check topology	2	1	3	3	3	1
Run simulations	11	4	9	33	8	0

The authors believe that such a training regime will enhance the subsequent acquisition of interaction skills and the user's ability to support the design process concisely and flexibly.

Who is in a position to deliver such initial training? The authors believe that IBPSA through its regional affiliates such as BEPAC and IBPSA-USA, universities, and continuing professional development groups are often better equipped than tool developers to provide this sort of regime.

The authors also believe that tool developers would be well served by implementing journaling in their tools. This would provide them with better feedback on the areas that novice or expert users have difficulty working with in their tools.

REFERENCES

Bartholomew, D., J. Hand, S. Irving, K Lomas, L McElroy, F. Parand, D. Robinson and P. Strachan. 1997. "An Application Manual for Building Energy and Environmental Modelling", *Proceedings of Building Simulation '97*, Prague, Czech Republic, September 1997. IBPSA.

Clarke, J. A., J. W. Hand, D. F. Mac Randal and P. Strachan. 1995. "The Development of an Intelligent, Integrated Building Design System within the European COMBINE Project", *Proceedings of Building Simulation '95*, Madison, Wisconsin, USA, August 1995, pp. 444-453. IBPSA.

Clarke, J. A., J. W. Hand, D. F. Mac Randal, and P. Strachan. 1994. "COMBINE IIBDS Prototype 3: Progress, Design and Implementation Issues", *ESRU Technical ReportR94/10*, University of Strathclyde, Glasgow, Scotland.

Hand, J. W. and J. L. M. Hensen. 1995. "Recent Experiences and Developments in the Training of Simulationists", *Proceedings of Building Simulation '95*, Madison, Wisconsin, USA, August 1995, pp.346-353. IBPSA.

Hand, J. W. 1993. "How to Train Users of Simulation Based Thermal Performance Analysis Tools", *Proceedings of Building Simulation '93*, Adelaide, Australia, August 1993, pp. 39-45. IBPSA.

Oxford University Press. 1971. *Oxford English Dictionary*, pg. 2829. Oxford, United Kingdom.