Hrothgar IE
A Distributed, Persistent Inquiry Environment For K-12 Learning

T. D. Gottschalk
Center for Advanced Computing Research
California Institute of Technology
Pasadena, CA 91125

tdg@cacr.caltech.edu

October 18, 1999

Abstract

The concept of a scalable, interactive simulation environment is explored as a substantive application of High Performance Computing (HPC) technology within K-12 education. An integrated system, combining a scalable simulation engine, information/discourse database components and WWW-based access is proposed as a prototype framework for an inquiry-based learning environment - a progressive approach consistent with the calls for fundamental changes in science education advocated in both the *National Science Education Standards* and *Project 2061*. Pursuing the curriculum issue further, it is suggested that this simulation Inquiry Environment could be positioned as a key tool for constructing knowledge-building representations ('artifacts') in the sense advocated in modern, dialogue-focused theories of learning and teaching. Middle school is identified as the ideal testing area for this approach, with a broad, possibly multidisciplinary subject matter for the simulation (e.g., ecology, government, and economics). A number of technical issues related to design choices and implementation strategies for the simulation engine are explored. It is argued that the HPC features within the system design are essential in enabling the overall educational goals. The next steps in the implementation process are discussed, emphasizing the need for a collaboration of computational scientists, educators, and cognitive scientists in the identification of simulation scenarios and associated curriculum elements.

Contents

1 Synopsis

Technology and media innovation in American schools have been characterized by cyclic fads and a failure to use the sound tools and processes of science to systematically and progressively improve the quality of instruction. As we enter the 21st century, technology has become far too powerful and valuable a learning tool to allow this pattern to repeat.

- Project 2061 Blueprints for Reform

This document is a concept white paper concerned with two questions:

1. Is there a substantive role for High Performance Computing in elementary and secondary education?
2. Can High Performance Computing play a significant role in the kind of **Inquiry Learning** advocated in the *National Science Education Standards* and in the AAAS's *Project 2061*?

It is suggested here that persistent, highly interactive simulations are ideal educational applications of scalable computing, particularly for middle school and early high school. Going further, it is suggested that such systems provide innovative *Inquiry Environments* supporting knowledge building activities, as advocated in the National Science Education Standards, the Project 2061 Blueprint, and many cognitive science research groups. It is argued that, for K-12 education, these simulations have their greatest potential in providing knowledge building laboratories for the social sciences, with 'knowledge building' here understood in the sense advocated by modern cognitive science.

The technical core of the Inquiry Environment is a scalable interactive event simulation, with behaviors of individual entities controlled by students. An appropriately general simulation framework, adapted from existing large scale simulation models, would support implementations of a wide range of synthetic environments relevant for knowledge building curricula. Basic considerations of necessary support infrastructure (e.g., interactive, persistent dialogue capabilities) lead naturally to the notion of an educational metacomputing system - hence the 'Environment' part of 'Inquiry Environment'.

Sections 2 and 3 set the stage, presenting general ruminations on the plausible educational roles of High Performance Computing and a brief review of work in 1998 on *'Hrothgar'* - a project to use cluster computing for technology insertions within high school science curricula. The pragmatic difficulties with the 1998 approach are noted.

A revised strategy is suggested in Section 4. The essential element is a shift of focus away from 'modified big science' towards large, highly interactive simulations of simpler, more familiar concepts as would be relevant for inquiry-based elementary and secondary education. A consequence of this shift is a bias away from 'pure sciences' and towards integrated, multi-disciplinary environments that fall more naturally within the students' 'comfort zones'.

Educational aspects of Hrothgar IE are examined in Section 5, beginning with a list of desirable features of an education-directed simulation and some initial thoughts on candidate simulation scenarios. Basic themes from contemporary cognitive science and educational research are explored, leading to the suggestion that the proper role of High Performance Computing for K-12 is that of a knowledge building 'artifact' within inquiry-based curricula.

The technical aspects of the proposed model are explored in Section 6. These include discussions of an appropriate simulation framework and extensions of the basic system into a more comprehensive metacomputing framework. The roles of some existing educational knowledge building software and procedures within this framework are explored.

Section 7 attempts to form a more focused picture for a realistic Hrothgar IE project and starts the task of proposing a specific plan of action for system design and implementation.

2 Prologue

Caltech's Center for Advanced Computing Research (CACR) [1] has been a leader in the area of High Performance Computing (HPC) essentially since the inception of the field. HPC continues to expand the realm of 'feasible computation', with significant research and practical applications in a number of areas. The development of effective, cluster-based HPC ('Beowulf' technology [2]) makes high performance computing significantly more affordable and accessible.

The K-12 (kindergarten through twelveth grade) educational outreach activities within CACR are exploring the possible uses of HPC within elementary and secondary education. Specific questions to be examined include

1. How can HPC best be incorporated into the K-12 curriculum, as a **technology demonstration** and/or a **technology exploitation**?
2. Equally significantly, how can this be done so that the 'High Performance' part of 'HPC' is essential? (We do not need yet another superfluous high-tech demo.)

There are two general approaches one could consider for using HPC (specifically Beowulf) in K-12 education:

An Extra-Curricular Activity: Hardware access and some level of training are provided to the high-end students. This gives hands-on opportunities for likely future scientists, engineers, Wall Street high rollers, etc.

A General Classroom Tool: HPC is explored/pursued as the enabling technology for classroom learning opportunities or demonstrations which would otherwise be difficult or impossible.

The first objective is well worthwhile, and is indeed the focus of a number of existing HPC K-12 outreach projects (see, for example, the web site of NPACI-EOT [3], the National Partnership for Advanced Computational Infrastructure - Education, Outreach, and Training). This project will instead concentrate on the second item. An important objective is the identification of an HPC strategy which affects as many students as possible, not simply the brightest and the best.

There are several fairly obvious motivations in attempting either approach to K-12 HPC:

1. Expose students to leading edge technology and university research.
2. Enhance appreciation/understanding of mathematics and science concepts.
3. Motivate students to explore/pursue technical careers.
4. etc., etc., ...

There are, however, three very significant flies in the ointment for educational technology insertions of this magnitude:

- **Relevance:** The content of an HPC application must match and extend elements of an existing K-12 curriculum (often drawn from unenlightened state curriculum guidelines).
- **Scale:** Applications should truly require HPC in the sense that some significant benefit or lesson would be lost in (simpler) conventional, single-computer applications.
- **Teachers:** Training requirements and general 'buy in' considerations for the classroom teachers are non-trivial obstacles.

This document outlines a proposal (actually, 'emerging proposal' might be a better term) for incorporation of HPC computing into mainstream K-12 education. This is, in fact, a very broad problem and the correct solution could require some change in the character of elementary and secondary education - changes which are, however, within the direction and mandate of the recent *National Science Education Standards* [4] and consistent with an overall emphasis on collaborative learning.

3 The Initial Hrothgar Experiment (1998)

The concept of a Beowulf machine dedicated to K-12 applications had been discussed within CACR since early 1997, and the machine itself (named 'Hrothgar', after the old king in the Beowulf legend) was constructed during summer, 1998. The project is documented in more detail in the Hrothgar WWW site [5].

The early Hrothgar activities were based on three nominal assumptions:

1. **Large-Scale Simulations Have Educational Utility:** It is worthwhile to expose students and teachers to advanced technology. This should enable new categories of modeling and analysis as aids to learning.

2. **Underlying Simulations Should Come From The Research World:** Involvement of computational scientists is required to exploit the potential of High Performance Computing.

3. **K-12 Educators Must Direct Curriculum Design:** Computational scientists are generally incompetent for the task of 're-focusing' underlying science concepts to the level of the target audience.

These tenets resulted in an initial strategy of technology insertion through a number of short units focused on specific aspects of existing curricula. To enable widespread access, the units would be WWW-based, with interactive use of the Beowulf cluster as a computational engine.

An initial unit, based on the Ising Model picture of magnetization, was completed during the summer of 1998. In addition to the physics of phase transitions, the unit illustrates a number of more fundamental concepts:

1. The role of computers and computer simulations in science.

2. Basic concepts of applied statistics.

3. Modeling and 'Thinking Like A Physicist'.

4. The role of Quantum Mechanics in macroscopic phenomena.

The unit has two interactive computational components, a *Sampler Page* which generates images of the system at user-controlled parameter values (e.g., temperature), and a *Physics Calculator* which lets the students 'measure' various system parameters (e.g., magnetization).

Fig.(1) illustrates output from the configuration sampler. The two images correspond to temperatures just above and just below the critical temperature, with the different shadings indicating 'up' and 'down' spin states. The numbers in the 'Correlations' column are quantitative measures of clumping, as discussed in supporting WWW pages. The underlying calculation is significantly larger than those associated with Ising Model research calculations done and published in *Physical Review* during the early 1980's. Instead of the hours or days of the initial studies, the entire calculation for Fig.(1) was done in about 50 seconds.

Additional units identified as candidates for the Hrothgar project included *Airshed* [6][7], an interactive air quality model for the Los Angeles basin, and a (still) vaguely-defined simulation of stock markets.

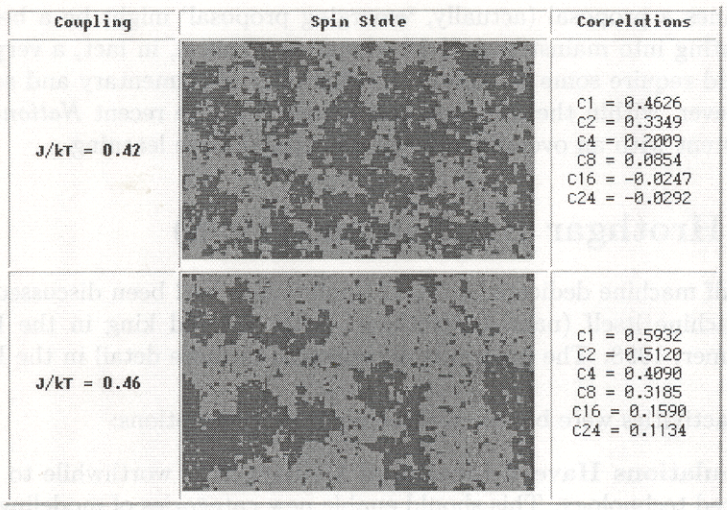

Figure 1: Hrothgar Ising Model Output

3.1 Assessment of the Initial Effort

There is considerable promise in this approach, particularly in terms of reaching the high-end students and introducing these students to the key role of computational science in real world research and applications. (This claim is supported by, among other things, the definite contributions of a number of such students in the actual implementations of the Hrothgar units.)

However, the initial efforts have brought to light two generic difficulties with the original Hrothgar strategy:

1. When the underlying science is simplified to the point of being accessible to high school students, the need for HPC becomes less obvious. (Science education at this level, even for high school, is much more hands-on than computational).

2. The current focus on college admissions and Advanced Placement Exams leaves little room for 'innovative excursions' for the high-end students.

In short, the initial objective of bringing research quality computational science down to the level of K-12 will be difficult. Though still clearly worthwhile, this 'obvious' first implementation model has significant problems. The differences between research science at a level which necessitates HPC and the kind of 'experiential' science needed in elementary and secondary education are nontrivial!

The Airshed environmental simulation provides an instructive illustration of the issues encountered when adapting research tools to the K-12 environment. Airshed provides high fidelity physics and chemistry models of the processes leading to air pollution in the Los Angeles basin. In order to perform a full calculation in under one hour, a Beowulf machine with more than 20 processors is required. However, most of the computational complexity within Airshed is due to careful modeling of aerosols. If the code is run in a simpler mode with aerosol physics ignored, a single processor is adequate.

For the types of investigations useful in K-12 applications, exact modeling of aerosol physics and chemistry is not particularly important, and the case for using a Beowulf becomes far less compelling.

4 Revising The K-12 Beowulf Strategy

If one insists on the initial objectives of seeking a wide-reaching niche for HPC in K-12, the important lessons from 'Hrothgar, The First Year' might be summarized as

1. The relevant aspect of HPC for K-12 is *not* simplified simulations of research science but rather big simulations of simpler K-12 concepts.

2. The driver in the utilization of HPC must be the K-12 educator, with the computational scientist taking more of the role of a facilitator and evangelist.

3. High school juniors and seniors preparing for college are not likely to be the best target audience, as these students (and their teachers) are too focused on college admissions and test taking. If one wants to explore something big and new, the relatively greater curriculum freedom at the middle school level provides a much better laboratory.

Much of the important academic work on big HPC machines deals with difficult science. Reducing the science component of such applications to the K-12 level without eliminating the need for HPC is difficult, especially if the objectives here are applications aimed at the general student population.

Visualization is an attractive alternative application of HPC. Along these lines, the National Center for Supercomputer Application's (NCSA) *RiverWeb* project [8] intends to explore (among other things) the use of a CAVE for visualizing/exploring the Mississippi River watershed. However, high end graphics also demand high bandwidth to the end user (i.e., the students), and this cannot be assumed for the notional target classrooms of this project.

Simulations remain the most attractive option for a broadly applicable technology insertion of HPC into the K-12 environment. However, unlike the Hrothgar '98 effort, the design and scope of such systems should be driven from the K-12 end, not the HPC end.

It is worth restating the shift in emphasis of this paper relative to the efforts of 1998: The focus is now on 'Big K-12 Computing' - which is very different from 'Modified Big Science'. Bringing true HPC-style big science down to the K-12 world will likely have far less lasting impact than the approach of adapting Big Computing as a new tool within the constraints of K-12 education.

Properly defining the (still vague) term 'Big K-12 Computing' will be an interesting, ongoing task. Indeed, it will likely be the central issue of 'Hrothgar 1999, Phase II'. Restating some of the lessons of the 1998 work:

- Educators tend to think far too small when asked to imagine what 'big computing' means.

- Computational scientists have vague, inadequate pictures of how children and adolescents learn.

This project retains the belief that HPC can have a significant role in education - particularly inquiry-based learning. However, it is now evident that neither computational scientists nor educators, individually, can lead the effort. The goals of 'Hrothgar 1999' cannot be achieved without fundamental partnerships - partnerships on the actual assembly line, not in the board room.

Along these lines, a series of informal meetings at CACR was begun in June, 1999. To date, these sessions have been deliberately free form, with the educators describing good learning processes and the CACR personnel constantly chiding the educators to think bigger.

These meeting have lead to the notion of some sort of large-scale, (highly) interactive simulation as an application with considerable potential. In particular:

- Students would directly control a number of 'entities'.

- These entities would interact with each other and with an appropriately large set of background entities simulated entirely on an HPC machine.

- Entity interactions would be developed specifically to support inquiry-based learning within some broader curriculum element.
- The usual HPC simulation toolbox would be exploited, including, for example:
  - Faster than real time 'what if' investigations.
  - Event logging and checkpoint/restart.
  - Simulations linking geographically distant sites.
  - Dynamic, distributed databases.

The next two sections address, respectively, the educational considerations which will direct the selection of candidate simulation scenarios and the set of HPC tools which would enable construction of the actual software system. These discussions are fairly lengthy, in keeping with the intent of a true 'concept exploration'.

5 Educational Considerations

The National Science Education Standards [4] can easily be interpreted as a call for altering the entire education process, not merely the science component. The call for *Minds-On*, not simply *Hands-On* and the slightly reworded characterization that *'Learning is something that students do, not something that is done to them'* should ideally apply across the full education landscape, not solely in 'Scienceville'.

Science learning in K-12 needs to be Hands-On/Minds-On, done with bar magnets, bits of wire, and disgustingly real dead frogs. 'Interactive' frog dissection software is, at best, a reference tool, similar to a set of good color charts in a text book.

There are, however, areas in which computers and software can become the legitimate basic building blocks for inquiry-based learning. Software like *Widget Workshop* [9] and Mitchell Resnik's *StarLogo* [10] can be characterized as computer-simulated buckets of bolts and nuts which the student then uses to explore and define new learning worlds. Indeed, such simulations are ideal laboratories for discovering large-scale consequences of simple, local rules.

The Hrothgar brainstorming sessions of summer 1999 identified the general area of interactive, student-directed simulations as the domain in which HPC could likely have its biggest effect on K-12 education. In order to justify big computing, it was realized that broadening the scope beyond conventional sciences would be necessary, leading to a variety of (multi-disciplinary) simulation scenarios, as examined briefly in Section 5.2 below. The more important issue of aligning the revised Hrothgar effort with contemporary research in education and cognitive science is addressed in Section 5.3.

Identification/selection of prototype simulation scenarios will be a critical contribution of educators and cognitive scientists to the next phase of the Hrothgar project. Work to date has attempted to keep the box of possibilities as wide open as possible, leading to a number of general considerations as described in the next section. At the present time, '*Why?*' is regarded as a far more important question than '*What?*'.

5.1 Simulation Characteristics: A Wish List

This section simply lists a number of key features for a K-12 simulation engine, as developed through a number of small meetings during summer 1999. The ordering of the items is largely arbitrary.

An Environment For Development Of Critical Thinking Skills: In the spirit of inquiry-based education, the simulation must provide an environment which the student *directs*, not something she simply views. Ideally, this would be a rich, loosely structured world with ample opportunities to analyze, synthesize, summarize and hypothesize. More importantly, the student then has the possibility/necessity of deciding on some course of action which actually <u>alters</u> the simulation, with the consequences of these decisions unfolding as the simulation progresses.

Within the general simulation framework under consideration, it is essential that students have considerable freedom in defining and altering the character of the entities that they control.

Relevance To Existing Curricula: On the one hand, this is simply a statement of motherhood and apple pie. On the other hand, it is an essential step in securing teacher acceptance of the underlying concept. The experience with Hrothgar 1998 has been that convincing teachers to experiment with a new paradigm is a non-trivial task. In the present case, the teacher technology discomfort factor is worsened by the realization that many students are already extremely comfortable with interactive simulations through WWW-based games. As Tapscott discusses in *Growing Up Digital* [13], this is an area that many students already know better than some of their teachers ever will. This is a *huge* problem.

Teacher training will, of necessity, be a major component of this project. This should not, however, be a surprise as teacher training is also a large component of any educational innovation, such as the inquiry-based science curricula developed by CAPSI, the Caltech Pre-College Science Initiative.

Support Multidisciplinary Extensions: This is a straightforward concept: a well-designed simulation could be used within a number of classes. For example, an Environmental Simulation could support teaching objectives for both biology and social science classes.

Promote Distance Learning and Collaboration: Once suitable interactive simulations are established for individual schools or classrooms, the possibility exists for interactions among simulations. For example, two separate simulations of feudal kingdoms could choose to interact through either commerce or warfare.

Develop Problem Solving Skills: While an open, student-directed simulation framework remains the central objective, it should also be the case that a successful tour of the simulated world involve a number of challenges or puzzles which test the student's problem solving skills in various areas. (It is worth noting that many popular interactive WWW-based games include significant challenge components.)

Provide Replay and Fast Forward Capabilities: Students will make 'wrong' decisions at various stages (assuming the underlying simulation scenario is appropriately rich). Once the bad choice is identified and understood, it would be good to have the opportunity to go back and try again. Similarly, when faced with some decision, it would be useful to quickly play through a few 'what if' sample simulations.

Encourage/Require Teamwork: The problem-solving challenges presented through the simulation should be rich enough that best solutions involve cooperation among the entities controlled by individual students.

Provide A Persistent Educational Element: The environment portrayed by the simulation should continue to 'live' between classroom sessions. This clearly lets students develop understanding and strategies incrementally over an extended period. Moreover, if the underlying background simulation is rich enough, it allows the consequences of student actions (e.g., dumping used oil into a stream) to unfold at a more realistic rate.

This notion of persistence is an important part of new interactive environments from both the educational (Whyville [11]) and commercial/entertainment (Terra [12], for example) realms.

Student Directed and Interactive: The basic elements outlined above are inherently interactive with the students observing, analyzing and altering the underlying simulation. Beyond this, however, it is desirable that the *depth* of interactions supported by the system could vary according to individual student abilities. Particularly in early adolescents, high level qualities such as 'global thinking' emerge along highly personalized time lines. A good simulation must engage students at all stages.

5.2 Plausible Simulation Scenarios

The feature list of the preceding section is (ideally) independent of the actual underlying simulation. As previously noted, the content of the simulation environment is presently - and intentionally - undecided. The general categories considered to date include:

- **Cities Or Other Societal Units:** Basic entities would include citizens and government agencies. Lessons might include the difficulty of defining 'the common good' in a diverse society.
- **Ecosystems:** The underlying simulation might model large numbers of individual, 'locally interacting' organisms, and the students could explore both the characteristics of equilibrium and the consequences of disrupting this balance.
- **Economies:** The underlying simulation could model and monitor some framework of normal commerce activities as well as sporadic unusual events, such as natural disasters.

The list of interesting candidates is surely much larger, and the next phase of this project will first expand the set of options, and then select candidates for an initial, proof of concept system.

As an illustration, the City example can be expanded a bit. The following is a (nearly) verbatim reproduction of a Concept Paper prepared by Dr. Janet Hoult, Education Department, California State University of Los Angeles.

Elements Of A City-Sim

Rationale: The use of high performance computing can add an exciting dimension to the secondary school classroom. The technology is available but the curriculum in which to use it needs to be designed. The difficulty is that any curriculum must be such that it demands the use of high performance computing. Such curriculum must avoid the pitfalls of establishing scenarios which require limited data bases in order to reach a solution. An ideal curriculum would require the students to use the extensive data bases in their research efforts in order to come to a consensus on how to handle a problem. An ideal curriculum would create an artificial environment that students could interact with. An ideal curriculum would include a 24 hour web-site for students to access at any time.

The format of such an environment might include:

City: One or more classrooms involved. Subject areas: Government, Biology, Economics, etc. Setting: Environmental aspects - geographical, industrial, etc.

Timeline: Current to future

Population: Citizens: variety of occupations/vocations (City Council, Businesses, Criminals, ...) Occupations: descriptions of job and place in the hierarchy of the city.

Activity: Students role play decision making roles in the city based upon a series of scenarios. Students conduct research using the HPC based upon the problems incorporated in the scenarios

Town Hall: Representatives from the various classes meet to compare notes based upon their research using the HPC and use the information in making informed decisions

Possible Scenarios: Flood - no food and water, Air Pollution, Water shortage, Epidemic, Earthquake, Fire, Embezzled funds, Drug dealers, Driving/owning your own car, Train - toxic spill.

The HPC component lies in the *scope* of the underlying simulation, with the computers providing a large, heterogeneous environment of interacting entities. Most importantly, the simulated environment is simply

a playing field. The only real outcomes and results in the simulation are consequences of student-initiated actions.

The examples/candidates listed in this section are all drawn from the social sciences. This is not an accident. As has been noted above, the K-12 curricula for the hard sciences have very little to do with the areas in which HPC is used in real research. Borrowing a descriptive term from Vygotsky's social-constructivist theories of knowledge and learning [14], there is very little overlap between the world of big science and a middle school student's Zone of Proximal Development (ZPD) - the zone between what a student can already do alone and the upper limit feasible with appropriate assistance.

The situation is quite different in the social sciences. The underlying 'nuts and bolts' of potential curricula can be drawn in great part from existing student experiences. The basic concept of a user-directed simulation has a great deal in common with computer and WWW-based games that are well known to a large fraction of the target student population (if not their teachers).

Finally, it is worth considering a claim from the introductory page of *Growing Artificial Societies: Social Science from the Bottom Up* by Epstein and Axtell [15], namely that social sciences are, in fact, the true "hard" sciences. Social processes are large-scale, complex, and are not easily separated into clean subprocesses. Controlled experimentation in the real world is difficult, as it is not possible to isolate and vary specific factors while keeping others rigorously fixed. The social forces which produce both eagle scouts and bigots are far harder to comprehend than those which hold atoms together. An emphasis on sociology and government rather than physics and chemistry is arguably an introduction to the harder unsolved problems in both the research and real worlds.

5.3 Inquiry and Dialogue

> *Something is going on in elementary schools across North America that might strike the detached observer as insane. Millions of dollars are are being poured into high-tech equipment that is used mainly to produce the kinds of 'projects' that in an earlier day were produced using scissors, old magazines, and library paste. At the same time, and in the same schools, a back-to-basics movement has teachers obsessively concerned with covering traditional content and preparing students for tests.*
>
> - Carl Bereiter, *Education and Mind in the Knowledge Age*, in preparation.

> *The research paper would read more like Dostoevsky than like Tolstoy.*
>
> - Alexander Sidorkin, *Beyond Discourse*, 1999.

The various meetings and discussions which led to this paper invariably included some musings as to why the entire project was even worth pursuing. Suggested reasons ranged from the mundane (need to make CACR 'Outreach Compliant') to more grandiose thoughts on designing *the* educational framework for the twenty first century.

It is, in fact, entirely appropriate to be thinking 'really big picture' at this point in the Hrothgar IE project definition activities. Both the *National Science Education Standards* [4] and the AAAS's *Project 2061* [16] present strong arguments towards significant changes in the goals and practices of K-12 education. In this vein, it is important to explore and assess the high-end, enabling technology aspects of HPC. It would be a mistake - or at least a significant missed opportunity - to settle for a mere technology insertion.

High Performance Computing - in particular, the Beowulf model - makes an order of magnitude increase in classroom computing capabilities a technical non-issue. However, as Bereiter notes in his book [17], most technology applications within current K-12 practice fall far short of the potential inherent in a single computer, much less a 'supercomputer'. The next subsections present cursory overviews of two education research programs which indicate directions in which Hrothgar IE might become a true enabling technology.

These comments should be viewed as initial steps on a much longer road of matching Beowulf capabilities to ongoing research within the education world.

5.3.1 Center for LifeLong Learning and Design (L^3D)

Two cartoons from a 1995 NSF presentation [18] by Gerhard Fischer, Center for LifeLong Learning and Design (L^3D) [19], University of Colorado at Boulder, provide simple schematics of both the current state and needed future of education within an increasingly technical society.

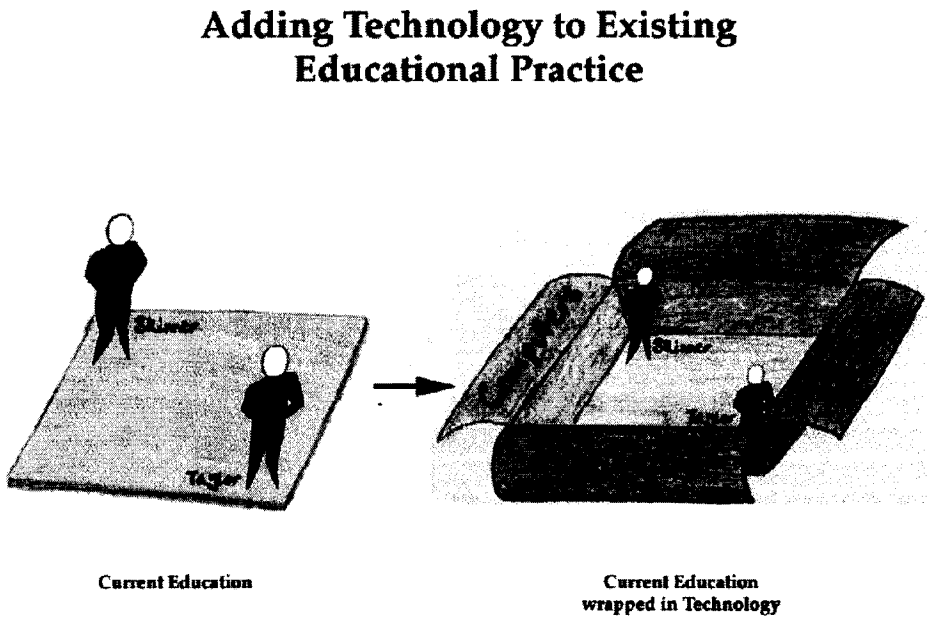

Figure 2: The Standard Model

Fischer first notes that most technology 'utilization' within K-12 is simply 'Gift Wrapping', as illustrated in Fig.(2). The Internet, WWW and desktop publishing may well be faster and tidier than old fashioned scissors and paste, but they do little to advance real educational practices beyond the standards of the past decades. This is also the essence of Bereiter's complaint at the beginning of this section. Even the more nominally enlightened uses of educational technology, such as intelligent tutoring systems and computer-based training modules, do little to advance an educational system that is still entrenched in curricula memorization, decontextualized rote learning, and exam performance standards.

The first step in proper technology utilization is likely a big step back to reassess the entire education process itself. A significant, recurring theme in contemporary cognitive science is the recognition of knowledge as a process, and not as a commodity which is transferred from one receptacle (e.g. a textbook) to another (a student's mind). Knowledge is developed - indeed defined - through the discourse of people doing things together. The structured old way of Fig.(2) is replaced by a dynamic, reengineered, and partially ill-defined educational dialogue, as indicated schematically in Fig.(3). In essentially all implementations of the redesign of Fig.(3) the role of teachers changes from truth-tellers to coaches, mentors, and even co-learners. (This is often described as replacing the 'Sage on Stage' by the 'Guide on the Side'.)

The real meaning of Fig.(3) is, of course, vague and the subject of considerable ongoing research. Knowledge discovery through discourse is seen to an essential component, although this is not possible without some underlying, structured framework. (An interesting, albeit somewhat simplified *'Three Courses'* picture of authoritative, dialogical, and diffused discourse is presented by Sidorkin in Ref.[21].)

Rethinking, Reinventing and Reengineering Educational Theory and Educational Practice

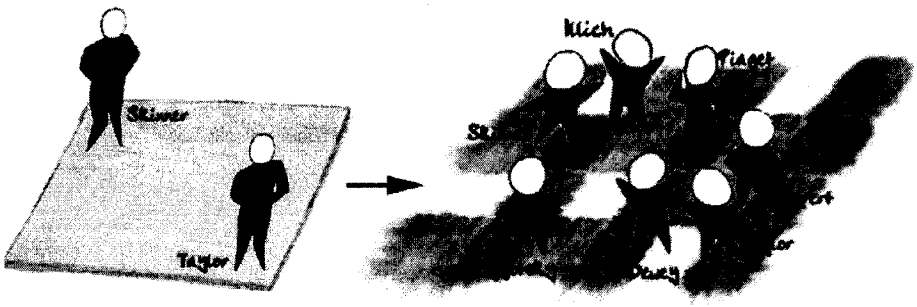

Figure 3: Beyond Standards

In Ref.[20], Fischer considers a number of issues related to learning not only in the traditional school context but also from the broader picture of a lifelong process. In order to train students for the continuous (re)learning demands of contemporary careers, the education process should reflect the nature of learning demands within the workplace, including:

- Learning should take place in the context of authentic, complex problems.
- Learning should be embedded in the pursuit of intrinsically rewarding activities.
- Collaborative learning must be supported.
- Skills for 'learning-on-demand' must be developed.
- Learning happens in doing. *Real* learning most often happens in trying, getting stuck, and finding some resolution.

All of the items in this 'Life Long Learning Litany' are relevant, but it is perhaps the last one which should be noted most from the perspective of Hrothgar IE design. One need only count the hours adolescents will willingly spend trying to outsmart a recalcitrant troll in some interactive fantasy game in order to appreciate the learning opportunities available through (appropriate) failures. Hrothgar IE should certainly create and pursue instances of 'controlled frustration' as genuine learning opportunities.

Finally, it is useful to extend the right-hand caption of Fig.(3) into a more hierarchical view of computers in the 'Education of the Future', as shown in Fig.(4). Basic desktop publishing applications are in the support category, as are the data collection and plotting packages used extensively within Caltech's Project S.E.E.D. Other programs more directly related to knowledge creation activities might be classified as mediating new knowledge (a properly used word processor is a key example here). At the extreme end are activities which are simply not feasible without a significant computational framework. This last category is becoming increasingly common in research and industry, but is still largely unknown in the K-12 world. In this regard, Hrothgar IE can be viewed as a specific campaign to demonstrate the utility of the last bubble.

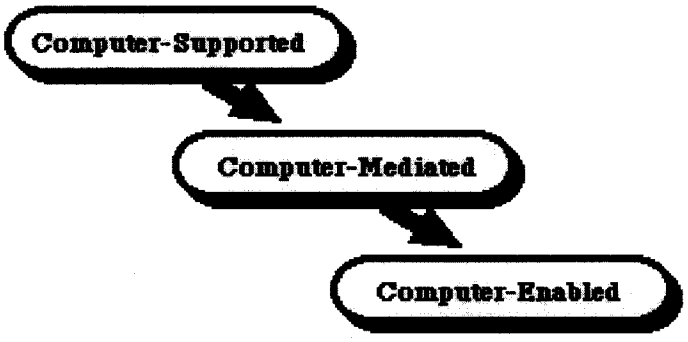

Figure 4: Levels of Educational Computing

5.3.2 Ontario Institute for Studies in Education (OISE)

Continuing the cartoon analogy of the preceding section, the book *Dialogic Inquiry* [22] by Gordon Wells might be characterized as plucking Vygotsky from the discussion carnival of Fig.(3), allying him with the linguist Halliday in the formation of a more unified picture of knowledge and knowing, and then unleashing this tandem onto the problem of 'modern curricula'. This social constructionist model, with an emphasis on the importance of semiotic artifacts in the 'co-construction' of knowledge, is presented as an alternative to both structured, teacher-directed curricula and unstructured discovery learning. A number of specific educational activities and classroom practices along these lines have been constructed by Wells and his colleagues at the Ontario Institute for Studies in Education (OISE) [23] at the University of Toronto.

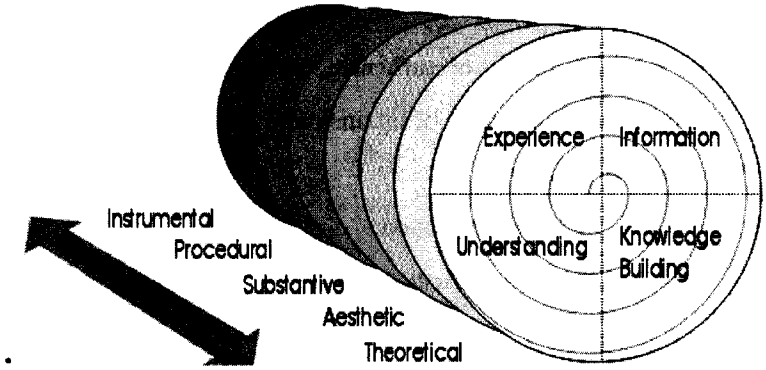

Figure 5: Wells' "Spiral of Knowing"

As with many modern investigations of 'learning', Ref.[22] begins with discussions on the meaning of knowledge itself, leading to the spiral metaphor shown in Fig.(5). The projection into the figure characterizes varying modes of knowing, ranging from the 'Instrumental' knowledge of basic tools on the part of primitive humans to the development of Theoretical (scientific) knowledge within the last three millennia. While the identification of socio-historical phases in the development of knowledge is clearly interesting, it is the common, spiral model of constructing knowing within each mode that is more important for present purposes.

According to the spiral metaphor, knowing is achieved (or, more properly, incremented) in a four step process:

Experience: An individual's social history defines the context within which new stimuli are to be encoun-

tered and processed.

Information: This is, in general, an 'interpretation of others' - an expression of meaning as construed and presented by some external, often authoritative, agent. It can come in a number of genres, including speech, written text, physical artifacts, and works of art.

Knowledge Building: In order to assimilate externally provided information, the learner must construct, use, and progressively improve various representational artifacts. Ideally, this produces a consistent, coherent 'internalization' which is, however, individual and personalized.

Understanding: With time (and recurring use), the internal representations constructed during knowledge building become 'second nature', and part of the learner's enhanced experience base. This transformation of 'knowledge' into 'understanding' is almost holistic.

The cycle then repeats.

Beginning from a personal experience base, knowledge building transforms new information into understanding. Understanding, in this sense, is taken to be the real goal of any educational activity.

There are several key aspects to the educational model associated with the knowledge picture of Fig.(3), most importantly

1. Learning occurs through engaging in purposeful activity.
2. Teaching becomes a rather adaptive activity, requiring
   - Organizing the curriculum in broad themes for inquiry that encourage a willingness to wonder.
   - Providing/presenting a 'factual base' (the Information component) for individual curriculum elements.
   - Fostering personalized knowledge building activities through which students invent and use a range of knowledge artifacts.
3. Creation of a 'collaborative community of practice', using appropriate assistance to enable students to (eventually) construct understanding.

This is, indeed, neither the 'Standard Model' of Fig.(2) nor an unconstrained exercise in Discovery Learning. This is, instead, a carefully controlled *apprenticeship* model, in which the teacher demonstrates, observes, assists, and then withdraws while the student creates successively more elaborate - and hopefully new and creative - knowledge artifacts.

There would appear to be ample opportunities for incorporation of the Hrothgar IE simulation model within this picture. In particular, an appropriately flexible simulation would fit naturally into the knowledge building quadrant of the spiral. This could enable construction and exploration of truly innovative knowledge models for disciplines in which experimentation would otherwise be difficult or constrained (e.g., studies of AIDS transmission mechanisms).

5.3.3 Apologies

The preceding overviews of approaches and activities at both L^3D and OISE are, of course, extremely sketchy. Both groups have created specific programs and software procedures implementing their overall approaches, and some of these (in particular, *AgentSheets* from L^3D and *CSILE* from OISE) would appear to be good potential components for an overall 'metacomputing' Hrothgar IE metacomputing system, as discussed in Section 6.

Ref.[20] cites a number of research institutes working on self-directed learning models and practices. The notion of 'learning as a fundamentally social activity' is common to many of these sites (e.g., The Institute for Research on Learning [24] and the EduTech Institute [25]). There are many examples of

successful practices within these sites, and potentially useful models for student-computer interfaces. However, the emphasis on multimedia technology as the medium/mediator for social activity indicates a rather different focus than that pursued within this work. In a rather different vein, the DICEP group [26][27] (Developing Inquiring Communities in Education Project) at OISE/UT is focused on the almost 'human factors' problems of forming a true community of inquirers - problems which are likely very relevant when 'oddball' computational scientists are thrown into the mix of students, teachers and cognitive scientists.

Adoption of a specific educational philosophy is well beyond the scope of this paper (not to mention the understanding of the author). Nonetheless, from even the limited discussions in this section, it seems reasonable to conclude that the overall Hrothgar IE model could become a natural tool - a true knowledge building artifact - within the context of the 'Education of the Future' panel of Fig.(3). It is important to keep this big picture in mind as work on Hrothgar IE shifts into the realm of the disgusting details.

5.4 The Matter of Assessment

Theoretical physicists (and, presumably, chemists, biologists, ...) are fond of lamenting the fact the 'Beautiful theories are often killed by ugly experimental facts'. It is rather harder to find uncontroversial 'facts' in the more complicated realm of education and cognitive science. Instead, the role of theory killer is taken over by an even more insidious agent - state education guidelines.

Both the National Research Council's *National Science Education Standards* [4] and the American Association for the Advancement of Science's *Project 2061 Blueprints for Reform* [16] devote substantial sections to the issue of assessing the success of curricula and curricula reforms, with conclusions that are hardly encouraging. The National Science Education Standards lists a number of criteria which should be covered by an 'enlightened' science education assessment procedure:

1. The ability to inquire.
2. Knowing and understanding scientific facts, concepts, principles, laws, and theories.
3. The ability to reason scientifically.
4. The ability to use science to make personal decisions and to take positions on societal issues.
5. The ability to communicate effectively about science.

However, most state assessment guidelines currently focus on only the second item of the list - or more precisely its emasculated version of standardized factual testing. As is noted in the AAAS Blueprints for Reform:

> 'However, many statewide and most district-wide tests are inconsistent with goals of science reform. They are "off-the-shelf", standardized, multiple choice tests that are not well-aligned with standards or benchmarks and do not allow students to develop their own solutions to problems or to analyze, synthesize, and present information on their own.
>
> ... ...
>
> These tests force students and teachers to emphasize test-taking skills over and above other educational concerns, and they exclude many kinds of knowledge'

These considerable difficulties are exacerbated in multi-cultural, multi-language societies (such as most of California) in which 'solutions' such as English-only testing become propositions in state-wide elections. A front page article [28] in the September 2, 1999 *Los Angeles Times* chronicles the poor test scores in a well-funded school that is strongly committed to discovery learning - with a sub-headline suggesting that 'discovery' itself might be the culprit. The remarkably diverse student population at the school (racial, cultural/language, and economic) is not noted until well into the second page of the article.

In one regard, assessment is important to Hrothgar IE in much the same same way that the sheriff of Nottingham was important to Robin Hood - as something to be avoided! This remark is only partially facetious. Matching HPC capabilities to K-12 needs is still in the exploratory phase, and, as noted earlier, the decision to shift Hrothgar activities from high school to middle school was done largely to get away from curricula which are strongly focused on test-taking preparations. In this vein, the possibility of structuring the initial Hrothgar IE 'product' as a persistent after school activity is also being explored. Some amount of unconstrained, almost free-form experimentation will be required initially (after all, as von Neumann noted, "Research is what I am doing when I have no idea what I am doing.").

However, assessment strategies for the knowledge building efficacy of student-directed simulations will be needed soon enough. The dual questions

- What can/should a student learn from a given simulation component?
- How is it determined that this 'learning', in fact, happened?

will clearly need to be asked throughout the exercise of defining and refining simulation scenarios. Guidance and cooperation from educational professionals is essential in this regard.

6 HPC Technology Considerations

Having abandoned, for the moment at least, the old model of bringing existing High Performance Computing research codes down to an appropriate K-12 level, the actual role for HPC in 'Hrothgar 1999' would seem to be less clear. Indeed, the almost knee-jerk CACR response to various early content suggestions from the educators was 'Yes, but why does this require a big computer?'.

As discussed above, the elements of HPC most relevant to K-12 would seem to come from the realm of distributed, interactive, non-homogeneous simulations. Before worrying about details of an initial, proof of concept system, however, it may be worthwhile to step back a bit and consider high level connections between this project and two ongoing major research efforts within the true HPC community: **Problem Solving Environments** [29][30] and the **Globus Project** [31]. Adopting a somewhat simplistic perspective, both PSE and Globus can be viewed as efforts to coordinate high end computational assets (including data management, communications, and asset allocation issues) in order to facilitate the solutions of complex problems. PSE and Globus are ultimately problem solving systems, with HPC just one of many components.

In this vein, it is tempting to regard this proposal as the first steps towards an analogous system for K-12: an **Inquiry Environment**. In place of the integrated Problem Solving Environment of the research world, an Inquiry Environment for K-12 education should be designed with the primary intent of enabling '*What If?*' explorations wholly (or at least largely) controlled by the students. The HPC simulation engine would ultimately be but one component of a more integrated system.

The distributed environment considerations of the previous paragraph define a sort of grand objective, and will be considered again in Section 6.3. First, however, the more basic issues related to the simulation engine itself and minimal required/desirable support features are examined in Sections 6.1 and 6.2.

6.1 Models for the Simulation Engine

A solid, interactive simulation framework provides the technical core for this project. Required features for such a system include:

1. A systematic procedure for defining *entities* and their interactions (*behaviors*).
2. A simple interface for direct, interactive student control of designated entities.
3. Automated evolution of a large population of background entities.

4. Simulation time management of various kinds (normal evolution, rollbacks, fast-forward, ...)

5. An appropriate model of 'the environment' and interactions of entities with the environment.

There is considerable in-house expertise at CACR and Caltech in regards to large, distributed simulations, including the *SF-Express* [32][33] redesign of ModSAF, a large DoD entity-level simulation [34]. *SF-Express* was ultimately run on a distributed, 600+ GFlops metacomputer that spanned seven time zones. This experience will be used in crafting an appropriate simulation framework for Hrothgar IE.

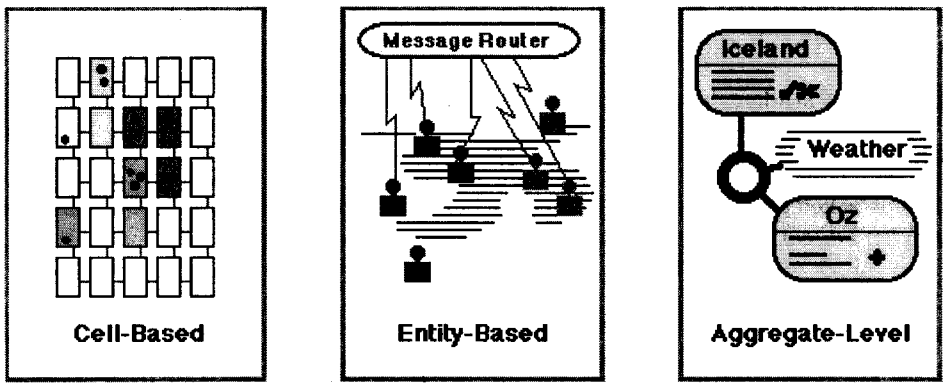

Figure 6: Event Simulation Models

Three possible architectures for the simulation engine are sketched in Fig.(6). The distinguishing features of these models are as follows.

Cell-Based: The 'universe' is represented by an array of cells, with each cell describing a (local) piece of the environment. Simulated entities (the dots in the picture) interact directly with cells and with each other.

Entity-Based: Each entity in the (conceptual) simulation is represented by an individual 'object' in the software, and these objects interact directly with each other through the exchange of data messages. The message passing can be one-to-one or one-to-many, and is managed by an explicit message management component of the software. Entity interactions with the environment can also be message-based.

Aggregate-Level: 'Entities' in this approach are large, complex items (army brigades, city sanitation departments, entire countries, ...), and entity interactions are typically implemented using non-trivial probabilistic models.

The **Cell-Based** formalism is used in *StarLogo* [35] and in *'Sugarscape'* [15] to model and explore macroscopic structures and collective behaviors in a simulation populated by large numbers of simple, similar entities (called 'agents' in [15] and 'turtles' in [35]). These simulations support three classes of interactions:

- **Entity-Entity:** Direct interactions between (nearby) entities. These could involve commerce, combat, or almost anything.
- **Entity-Cell:** Modification of an entity's state according to properties of the local environment (e.g., an agent finding itself in a food-rich cell can eat, increasing its energy supply).
- **Cell-Cell:** Environmental characteristics within one cell can affect those of neighboring cells, such as an oil spill diffusing outward and slowly polluting nearby areas.

This class of models has been used to explore the extent to which behavior in the aggregate differs from simple summations of entity-level behavior, extending the classic studies by Axelrod [36] and Schelling [37][38]. For example, large disparities in wealth are readily observed in simulations populated by nominally identical entities, thus suggesting that "The rich get richer" may simply be a law of nature.

In the sense intended in Fig.(6), the canonical **Entity-Based** simulations are those structured around the Distributed Interactive Simulation (DIS) protocol [39]. The most widely used example is ModSAF [34], an entity-level combat simulator. Key components of these simulations include:

1. High-fidelity entity level simulations of various kinds of objects (trucks, tanks, aircraft, drunk football fans, etc.).
2. Interactions among entities hosted on separate computers through a specific message-passing protocol.
3. Extensive support for human interactions, including real-time capabilities to alter behaviors and strategies of individual entities.

ModSAF was originally developed to provide a few dozen computer simulated 'opponents' for training exercises involving manned simulators. As simulation sizes increased, the nominal model of network message exchange fails. The HPC communications solution described in Ref.[33], with an explicit message router replacing network-based communications, enables ModSAF-style simulations with many tens of thousands of entities. Entity and environment models within ModSAF can be quite complex. The proceedings of the numerous DIS conferences, especially in the early 1990's, provide many demonstrations of the flexibility and utility of the ModSAF/DIS approach.

As one moves from left to right in Fig.(6), the complexity of the simulation entities increases. In the **Aggregate-Level** approach, the underlying objects (and their associated computer implementations) become extremely complicated. Typical 'entities' in this approach might include city governments, army brigades, or even entire nations. As the complexity of the basic objects increases, so does the nature of object-object interactions. Many Aggregate-Level simulations use elaborate stochastic algorithms to determine the changes in object state as the result of an interaction. There is, again, a significant example of this class within the DoD world: The Corps Battle Simulation (CBS) [40], which is used as a learning tool for military decision makers. For purposes of this work, the more interesting example is the World Game [41], an interactive implementation of R. Buckminster Fuller's World Peace Game. The entities are entire countries, and the players in the game must find cooperative solutions to a number of global issues.

6.1.1 Comments: Choosing The Simulation Scheme

None of the models indicated in Fig.(6) provides an immediate off-the-shelf solution to the problem of implementing a Hrothgar IE simulation engine. However, any one of them could serve as a useful/usable starting point.

The Entity-Based approach would seem to be the best overall fit to the simulation wish list of Section 5.1 with, in particular, individually-directed entities naturally interacting with a computer generated background population. Unfortunately, the ModSAF software itself cannot be simply adopted/adapted. The software package is huge (well over one million lines of code) and violates most modern software design principles. There is a more maintainable equivalent to ModSAF done using SPEEDES (Synchronous Parallel Environment for Emulation and Discrete-Event Simulation) [42], which will be explored as part of this effort.

Against the complexities of full-scale DIS simulators, the Cell-Based formalism is refreshingly simple (for example, the 'Sugarscape' software has about 20 thousand lines of code). However, these programs may not be immediately suited to the very heterogeneous operations model described above, in which some entities in the simulation are individually controlled by students.

The Aggregate-Level model should not be immediately dismissed as being too 'coarse-grained'. There are, in fact, significant overlaps between the elements of the notional City-Sim of Section 5.2 and the World Game simulation.

Fortunately, the problems associated with the Hrothgar IE simulation design are largely problems of excess - the candy store is almost too full! Given local CACR expertise, a useful simulation framework could be constructed using any of the paradigms shown in Fig.(6). The final design decisions will depend as much on the ancillary considerations of the next subsection as on any fundamental issues of parallel simulation paradigms.

6.2 Necessary Extensions

Beyond the basic simulation engine itself, there are a number of support components which must be incorporated into even the simplest system. These include:

WWW-Accessibility: The only technology which can be be presumed to exist at all user sites is a WWW browser. Student interfaces to the simulation (both active and passive) must be WWW-based.

Appearance Considerations: With a target audience of teenagers, the user interface needs to be engaging - especially in view of expectations students develop during leisure time WWW activities.

Database Components: Several categories of data management capabilities are required or desirable:

1. Simulation state storage, to support the persistence and rollback/replay features noted in Section 5.1.
2. On-line information on lesson objectives, background information, etc. These data could be dynamic as well as static.
3. A searchable storage area for student comments, questions, and dialogue, as generated during use of the simulation.

Entity Manufacturing: The system must provide a simple way for students to alter properties and (eventually) capabilities of individual simulated entities.

The first point emphasizes an implicit design assumption: the Hrothgar IE system will be constructed as a centralized (albeit distributed and scalable) resource which students access through the Internet. This is the model used in both the original Hrothgar project and in the new 'Whyville' project [11] by Caltech's CAPSI group. Whyville is rapidly coming into shape as an interactive virtual community with significant 'persistent persona' and individualized user interactions (e.g., students build their own houses within the sister city 'Myville'). During the creation of Whyville, the CAPSI team has developed considerable insights into the 'Engaging Interface' problem noted in the second item above. This local expertise will be exploited shamelessly during the design of Hrothgar IE screens and user interfaces.

The first two items listed under 'Databases' are fairly straightforward and, again, there is significant CACR familiarity with the design and implementation of these tools. Both the Digital Sky [43] and Globally Interconnected Object Databases (GIOD) [44] projects provide capabilities and procedures far beyond the needs of this project.

Indeed there are no fundamental technical hurdles associated with any of the support items listed above. Prototypes (even production models) can be found in most instances, and the associated Hrothgar IE construction process would seem to reduce to picking and choosing pieces in some plug-compatible fashion. Again, however, this easy technical approach would result in lost educational opportunities. Consider simply the question of interface appearance. Adopting, for example, Java as a likely implementation model, the technical questions are largely settled. But the more important task of presenting information in a manner appropriate for the intended audience has just begun.

In this vein, there are two items in the support list which deserve more detailed discussions:

1. An organized, interactive comment and question component.
2. Mechanisms to support student modifications and/or designs of the underlying simulated objects.

These issues are explored briefly in the following subsections.

6.2.1 Interactive Commentaries and CSILE

Chat rooms and 'ask the expert' areas can be found in a number of education-oriented WWW sites. These are good initial steps, but the potential lack of long-term organization and focus diminishes their utility with respect to the knowledge creation approach of Section 5.3.2. In discussing the World-Wide Web as a plausible medium, Fischer [20] considers an 'Evolutionary and Collaborative Design', in which the WWW is used to foster collaboration, research, and, in particular, *co-learning with peers.* Several key features of good sites are noted, including abilities to

1. Add information directly (without an intermediary).
2. Modify the structure of the information space.
3. Modify at least some of the existing information.
4. Integrate user contributions within an overall knowledge building framework.

The *Dynasites* package [45] by Jonathan Ostwald is an example of a viewer-extendable, dynamic web-based information space. The *Web-SMILE* package [46] from the EduTech Institute [25] is another well-developed example. Of more immediate interest for this project is CSILE (Computer Supported Intentional Learning Environments) developed by M. Scardamalia and C. Bereiter [47][48] at OISE.

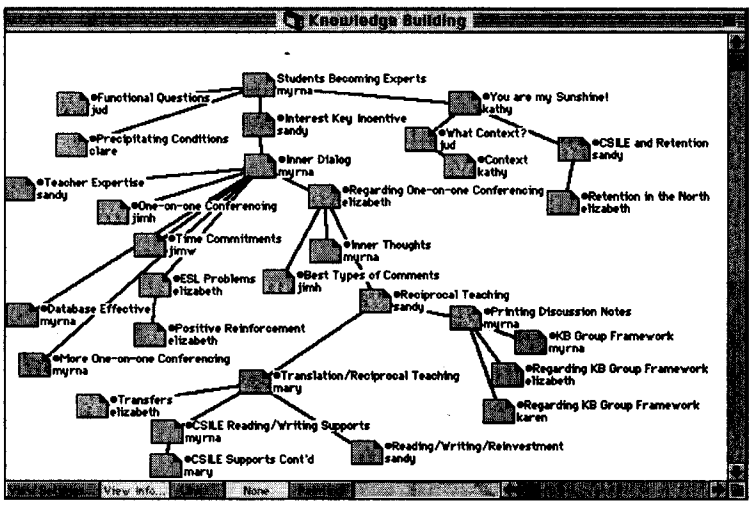

Figure 7: A CSILE Knowledge Map

Technically, CSILE is a centralized discussion database which is accessed through client processes on student computers. There are no separate places within the database for work by individual students or private student-teacher communications. All information is public. The intent is to provide a single repository of 'communal knowledge' which can be browsed, searched, and modified. Fig.(7) shows a typical 'Knowledge Map', providing a graphical directory of database contents relevant to user-specified topics.

The software is only one aspect, and CSILE should more properly be viewed as an integrated package of software and pedagogy intended to foster the notion of a highly interactive community of learners - a (new) culture of classroom knowledge-building. Within these broad goals, the refinements in CSILE over the past decade have involved not only software enhancements but an ongoing reworking of teaching and classroom practices. Standard classroom practices are, again, the primary hurdle. As is noted by Hewitt and Scardamalia in Ref.[49]:

"These highly-entrenched preconceptions interfere with the goal of establishing a classroom-based Knowledge-Building Community. The challenge, therefore, is not simply to provide *opportunities* for distributed processes, because students may not even recognize them as such. Rather, the challenge is to design situations and tools that invoke a ***deliberate bias toward shared activity***. This means placing students in the same virtual space, focusing them on the same problems of understanding, engaging them in constructive, whole-class discourse ... and so on" (emphasis added)

CSILE is not a controlled chat room. It is a distributed, communal knowledge artifact/tool intended to encourage exploration.

Within this environment, a carefully planned Hrothgar IE simulation-based problem might provide useful 'scaffolding' in keeping student CSILE activities focused. At the same time, CSILE could provide the means for students to negotiate cooperative solutions to challenges within the simulation (as appears to happen in a more restricted environment within World Game [41]). The synergy opportunities here are substantial.

Clearly, a 'Discourse Area' is not simply a nice add-on. Adopting something like CSILE into the overall Hrothgar Inquiry Environment is mandatory.

6.2.2 User-Controlled Entities and Interactions

Within the generic simulation frameworks of Fig.(6), the process of updating an individual simulation object at some instance is as indicated in Fig.(8).

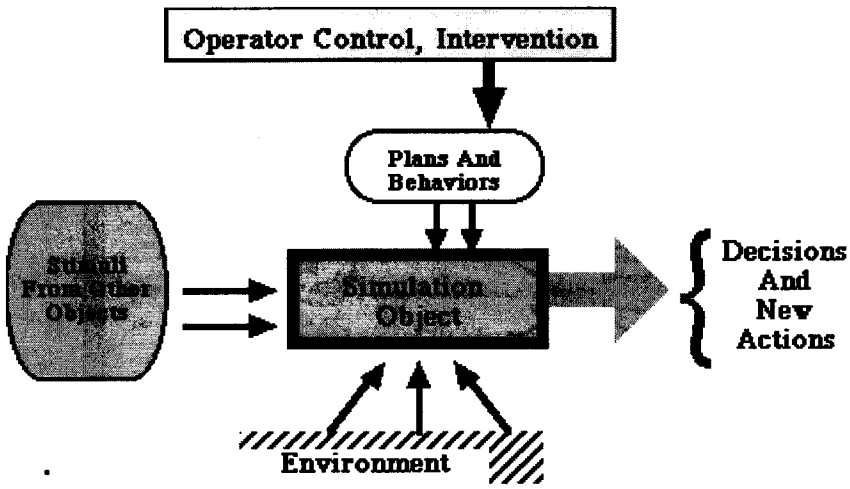

Figure 8: Object Update Paradigm

At the beginning of each update cycle, the object receives new information on its environment and other objects in the simulation. These inputs are processed according to plans and standard behaviors, leading to a new plan of action which the object then follows (e.g., in the simplest examples from *Sugarscape*, an agent simply moves to a nearby unoccupied cell that has a lot of food). In this picture, operator control is best pictured as intervention with the object's behavior model, either in terms of direct commands or in terms of parameter changes for the model itself. In the absence of operator intervention, the object will act/react indefinitely according to its last established standard behaviors.

The operations model of Fig.(8) is naturally implemented using Object Oriented Programming (OOP) practices. It is noted in Ref.[15] that this approach provides an extremely flexible environment for modifying object definitions and behaviors and then observing consequences (i.e., for doing research). Unfortunately,

this level of object control is only available to the simulation designers/implementors. It might be accessible to students at the true high end, but it is irrelevant for the general student population

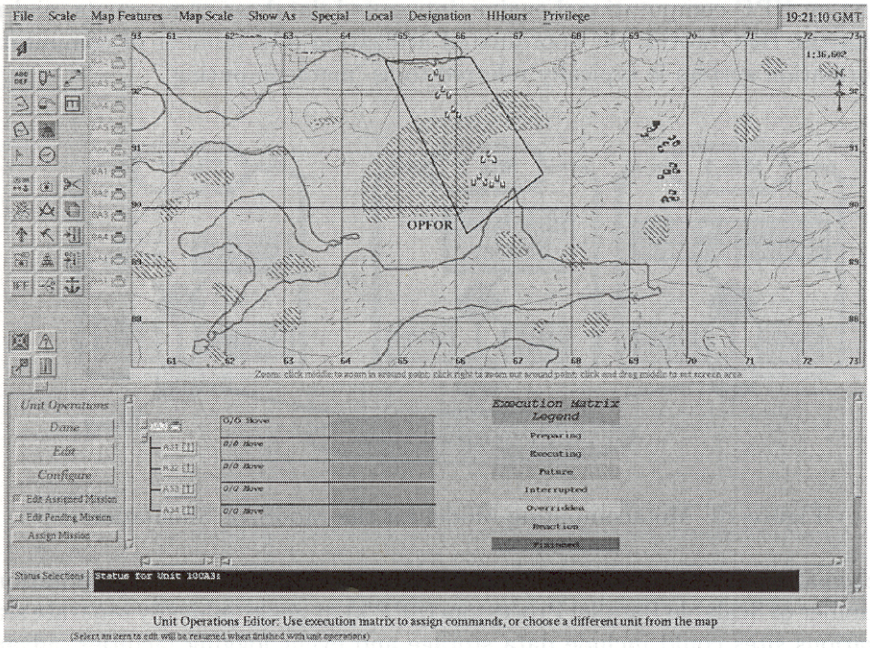

Figure 9: Sample ModSAF Object Control Screen

The standard mechanism for operator/student control of simulation entities is by way of a graphical user interface (GUI), in which information on global simulation status and current object status are displayed. The interface provides additional 'dials and buttons' which are used to alter the object's behaviors. A typical ModSAF GUI is shown in Fig.(9).

In the case of ModSAF, the required functionality of the user interface is extremely complex, with each of the little icons in the upper left portion of Fig.(9) spawning additional user interaction windows. The corresponding interfaces for Hrothgar IE will be far simpler. There is, however, an inherent problem in this *Parametric Panel* approach to simulation control - the problem solving options available to the user are limited according to the imagination/insight of the simulation and GUI designers. For example, the standard solution to gang activities in something like Sim-City would be an increase in the size of the police force. Alternatives such as neighborhood multi-cultural awareness programs lie outside the standard set of options.

The problem of providing true, unfettered user control to the behaviors in Fig.(8) is highly non-trivial. Resnik's *StarLogo* is essentially a separate programming language to allow full control of behaviors. It is quite successful within its realm, but not obviously extendable to the more complex simulations considered in this work. The *AgentSheets* environment [50][51] developed by Alex Repenning is a more elaborate, GUI-based procedure for the definition of processes and interactions. It has been used, for example, in creating *Mr. Roger's Sustainable Neighborhood* [52] - a computational environment for community planning developed at the Center for Lifelong Learning and Design. Again, however, this approach is effectively a new programming language ('Tactile Programming'), carrying with it both additional learning curve requirements for students and limitations inherent in the supported programming primitives.

An 'ultimate' implementation of Hrothgar IE should support student design and implementations of the basic schematic in Fig.(8). The best model for incorporating this flexibility is by no means clear. The initial Hrothgar IE work will almost certainly be GUI-based, but consideration must be given to less constrained alternatives throughout the system design process.

6.3 Educational Metacomputing

The underlying technical implementation emerging from the discussions in Sections 6.1 and 6.2 is already fairly rich, involving an HPC computational engine, integrated database components, and network access to students. Within the HPC world, the coordinated utilization of a number of computational assets to create a single information system is typically referred to as 'metacomputing'.

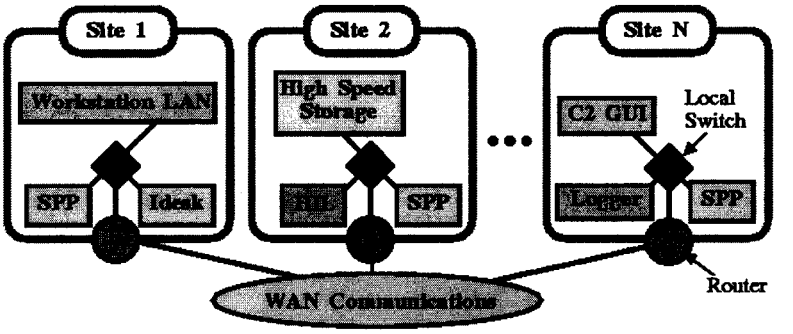

Figure 10: Metacomputing Model for the ModSAF Simulation

Fig.(10) is a schematic model of a metacomputing system for ModSAF. The computational horsepower is provided by a number of Scalable Parallel Processors (SPP's) at geographically separate sites. Additional resources at the various sites could include data storage and event logging facilities, advanced visualization facilities (labelled 'Idesk' in the picture) and various forms of Human-In-Loop (HIL) control. The Globus Project mentioned earlier is developing operations models for this class of very large, distributed computational systems. "The Computer" is no longer a single piece of hardware.

Within the general PSE/Globus picture mentioned above, there are several elements from the domains of simulation and metacomputing which could contribute towards the creation of a true K-12 Inquiry Environment. Topics which overlap existing research efforts at CACR and Caltech include:

- **Advanced Time Management Strategies:** For example, Faster Than Real Time excursions, enabling 'what if' studies within a single class period.
- **Distributed, Fault-Tolerant Databases:** These would be important for both Persistent Simulations (allowing a single simulation to be revisited as appropriate over a long time period, e.g., a semester) and for Logging/Rollback/Replay opportunities. Sophisticated databases would also be useful in analyzing the educational utility of specific simulation scenarios or models.
- **Heterogeneous, Multi-Threaded Beowulf Operations:** In the student-directed Inquiry Environment, the simulation engine is often a 'simulation server' which must respond to user requests covering a wide range of associated computational requirements.
- **Transparent, Meta-computing Environment:** An 'ultimate' system would involve replicated simulation and database servers at various sites, as well as various unique assets (e.g., a particular type of visualization workstation) at various user sites. Seamless operations of distributed metacomputing systems such as this are the primary focus of the Globus project.

These components are not all needed for initial Hrothgar IE work. However, the overall computational framework outlined in this section is such that these features could be incorporated in next-generation versions of the system.

There are several examples in the (WWW) literature of what could be called educational metacomputing systems, such as the *ExploreNet Experiment* [53] from the University of Central Florida, *CALVIN and NICE*

[54] from the Electronic Visualization Laboratory of the University of Illinois at Chicago, and the 'ultimate televirtual (TVR) environment' [55] from NPAC at Syracuse University. These projects are all heavily concerned with virtual reality and visualization and are thus not immediately relevant to the needs of this work. The metacomputing requirements of a mature Hrothgar IE system will be more concerned with issues of scheduling, reliability, and integration of distributed components - precisely the sorts of services being developed within the Globus project [31].

In terms of the 'Computer-Enabled' icon in Fig.(4), a full metacomputing system is truly a high-end solution. However, it should be remembered that the exploration of real, highest-end applications is in fact CACR's primary mission.

7 Outlook

> *Implementing the standards will require major changes in much of this country's science education. The Standards rest on the premise that science is an active process.*
>
> - Overview from the *National Science Education Standards*

> *Simulations, for example, are built on hidden assumptions, many of which are oversimplified if not highly questionable. ... Indeed, after mastering SimCity, a popular game about urban planning, a tenth-grade girl boasted to Turkle that she'd learned the following rule: "Raising taxes always leads to riots".*
>
> - T. Oppenheimer, 'The Computer Delusion'

The search for an appropriate K-12 niche for HPC has lead to the notion of an **Inquiry Environment**, providing knowledge-building capabilities for a range of curriculum elements.

The technical components of Hrothgar IE, as described in Section 6, are fairly straightforward. An initial system would likely contain:

1. An interactive discrete event simulation engine.
2. User access and control through a WWW-based interface.
3. Integrated database and discussion region support.

The High Performance Computing aspects of this system come in terms of the scalability of the simulation software and the integration of the various components into a seamless metacomputing environment. As was discussed in Section 6, there are no fundamental technical hurdles to the construction of the metacomputing system itself. The interesting implementation issues (e.g., user control of individual entities and procedures for advancing time) are dependent on the more fundamental question of educational goals.

Identifying an appropriate 'point of insertion' for Hrothgar IE within the K-12 education process, on the other hand, will require care. The excerpt from the *National Science Education Standards* [4] quoted above is applicable to most K-12 curricula elements, not simply the usual science subjects. Shortly after offering this claim, Ref.[4] suggests 'inquiry' as a significant feature of more enlightened practices:

> 'The *Standards* call for more than "science as process", in which students learn such skills as observing, inferring, and experimenting. Inquiry is central to science learning. When engaging in inquiry, students describe objects and events, ask questions and construct explanations, test those explanations against current scientific knowledge, and communicate their ideas to others.' (emphasis added)

Aspects of contemporary inquiry-based, dialogic models of learning were examined in Section 5.3, leading to the suggestion that a student-directed simulation could provide an effective tool for constructing knowledge artifacts. These discussions also suggested that the social and environmental sciences (as opposed to, e.g., physics) provide a promising general area for big simulations.

The use of simulators to study social sciences is hardly new - it is the essence of SimCity [56], StarLogo [35], Sugarscape [15] and other packages. In an article which is otherwise quite critical of technology uses within contemporary education [57], Todd Oppenheimer acknowledges that 'the best of these simulations may be valuable', but quickly tempers this concession with two complaints:

1. Students typically disengage from the learning potential of simulation software, focusing instead on reflex manipulations of screen images.

2. Many available simulation packages are built on hidden assumptions or limitations which are never fully explained (leading to peculiar conclusions, such as that quoted above).

There is considerable validity in Oppenheimer's analysis <u>if</u> the simulation packages are used and evaluated within the 'Standard Model' of Fig.(1). But, this need not be the case.

Oppenheimer's criticism can largely be avoided if the enabling technology of HPC-based simulation is used as a knowledge-building component ('artifact') within a something like a dialogic inquiry curriculum. More specifically,

1. The model of an interactive simulation <u>within</u> a structured approach as in Sections 5.3.2 and 6.2.1 provides the means for addressing Oppenheimer's first objection.

2. The second objection is solved by a <u>scalable</u>, <u>interactive</u> simulation engine which can accommodate the appropriate complexity, either in terms of numbers of entities or in terms of complexities of the individual entities.

This is an innovative inquiry system, not simply gift wrapped technology. The enabling technology of HPC is the support structure of the learning system, but not its cornerstone. This is an example of the reworked approach to learning advocated in both the *National Science Education Standards* and *Project 2061*.

There are, of course, very many details which remain unaddressed, such as the specifics of the entity interaction model of Fig.(8) or, more importantly, an initial simulation scenario. The next phase of Hrothgar IE development will begin the real 'nit processing'. Before sketching an overall implementation plan, though, it may be helpful to digress with a brief anecdote of experiences working on the Hrothgar project during summer, 1999.

7.1 An Unplanned Exercise in Inquiry

As an exploration into the potential uses of a Beowulf by 'high end' users, two high school students (Tyler and Will) and a new high school graduate (Nick) were given full access to the Hrothgar machine - including the root password - and instructed to "write a simulation". All three already knew C++, had varying degrees of comfort with Linux, and expect to pursue technical careers. The three worked well together, and an interesting 'critical mass' was soon formed (completely taking over the author's office for most of the summer).

After a few expected false starts and digressions (e.g., honing Unix skills by trying to crack password files), the students designed and implemented a simulated universe populated by carnivores, herbivores, and plants. The basic entity-level interactions were straightforward:

- Herbivores look for and eat plants.

- Carnivores look for and eat herbivores.

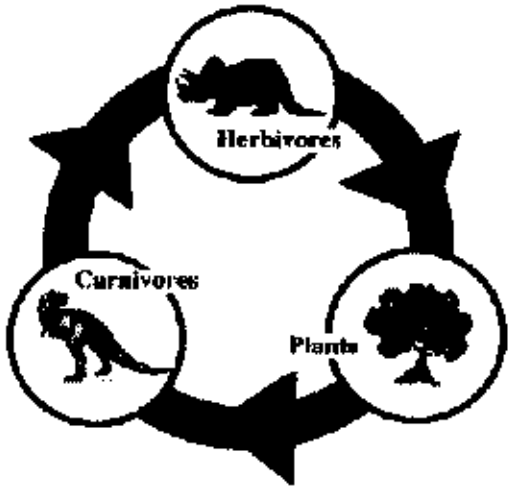

Figure 11: Will, Tyler and Nick's "Circle of Death"

Object behaviors in the sense of Fig.(8) were modified by trial and error changes to the source code. The environment had 'seasons' in the sense that the plant population was increased periodically ('spring happened'). A basic display screen was developed. Typical runs involved thousands of entities.

The students eventually created a usable laboratory for investigating the consequences of behaviors and behavior models within their simulated world. One of the more interesting discoveries is shown in Fig.(11), where a directed circular arc from 'A' to 'B' is to be interpreted as 'A eats B'. The conclusion that plants 'eat' carnivores was initially disquieting, but was soon easily understood. In a world with too few herbivores, the plant population increases without bounds. The foliage eventually becomes so dense that the carnivores can no longer see the herbivores. With no food in sight, the carnivores do nothing, and eventually starve.

The world created by Will, Tyler and Nick was pretty marginal in terms of biological or ecological modeling, but it was a superb environment for studying cause-and-effect relations and the global consequences of local behaviors. The last weeks of the summer saw considerable experimentation with details of the entity models, speculations as to what the code modifications might produce, and post-mortem discussions (occasionally with finger pointing) when the predictions went awry.

This very rudimentary prototype of Hrothgar IE certainly seemed to qualify as a tool for knowledge building. One can only imagine how much better the understanding experience would have been with a more focused learning task and a competent "Guide on the Side" to direct the dialogue.

7.2 Thoughts on the Next Steps

As was suggested throughout Section 6, the building blocks for the Hrothgar IE system can largely be drawn from existing technology. The process of constructing an end-to-end prototype can be divided into four main phases:

Assessment: Existing implementations of the various simulation paradigms of Fig.(6) are reviewed, and initial, provisional limits on the required functionality of the basic interaction picture of Fig.(8) are imposed. Existing 'dialogic tools' such as CSILE are reviewed at the level of understanding implementation models and costs.

Implementation: Proptotype code is written for a test case scenario (hopefully a bit more sophisticated that the Plant-Herbivore-Carnivore world of the preceding section, but this is not absolutely neces-

sary.) The emphasis in this phase is on global software design issues, such as scalability, flexibility, and generic user interaction methods. A sample database module is selected and/or developed.

Integration: The simulation and database components are combined into a 'first-light' metacomputing system, and the development and testing of user interaction models begins.

Interface Refinements: The presumably crude user interfaces of the Integration step are replaced by a sequence of refinements ending, quite likely, in some sort of WWW/Java form.

The emphasis here is on the quick creation of a functioning 'product', and the obvious danger comes in spending too much time in the Assessment phase, trying to make right decisions with respect to Figs.(6,8) without yet knowing how 'right' will ultimately be defined. It is likely that much of the proptyping could be done working closely with members of the CAPSI/Whyville team. CAPSI would contribute assistance and expertise in the interface area and CACR would share lessons from the Integration phase to assist in design and implementation of a scalable Whyville server.

The selection of an initial scenario model will be the harder part of prototype construction. It would be reasonably straightforward to simply select some element from the list of candidates in Section 5.2, implement a 'reasonable' set of initial behaviors for the objects, and then simply see what happens. This would be a next generation version of the Plant-Herbivore-Carnivore world of Section 7.1, with increased complexity and a whole new layer of bells and whistles. This would, in fact, be fairly easy. It would also be wrong. Such a product would likely be yet another example of a high-tech toy, of the type decried by Oppenheimer [57]. It would be, at best, a marginal step in the direction sought by the *National Science Education Standards* or *Project 2061*.

As was stressed throughout Section 5, specification of the scenario content makes sense only when taken together with a genuine curriculum plan. In order to address the challenges of the *National Science Education Standards* this curriculum plan cannot be based on the usual practices of K-12 education in this country. Instead, adopting Ref.[22] as a reasonable general model, the overall classroom unit should involve three key features:

1. Dialogic co-construction of knowledge.
2. A significant underlying activity in which knowing is embedded.
3. The construction and use of 'artifacts' that mediate knowing.

The construction of this curriculum unit must be done by educators and cognitive scientists. With the notable exception of CAPSI, such people are a bit scarce at Caltech.

The key next step, then, is the formation and extension of contacts within the education and cognitive science communities. The summer brainstorming sessions and various e-mail exchanges during the writing of this paper were important first steps. The overall success of the Hrothgar IE concept will depend in good part on the extent to which these contacts are strengthened. Borrowing another apt phrase from the OISE group, it is important to embed this project into a 'Community of Inquirers' focused on finding an appropriate place for High Performance Computing within 'Enlightened K12'.

Acknowledgements

Writing this document was, indeed, an exercise in knowledge building, and it would have been impossible without a extensive interactions and discussions from others. From the research/administrative world: Janet Fisher-Hoult, Linda Polin, Billie Jean Knight, Nora Sabelli, John Cherniavsky, Lorna Miller, Hall Daily, and Jim Bower. From the San Marino Unified School District: Andrea Mitchel, Jennifer Clark, Jerry and Kristen Koiles. From the Pico Rivera Unified School District: Felicity Swerdlow, Melissa Manning, Steve Francis, and Yamileth Flores Orihuela. From the broader community: Fin Cotton, Nancy Hunt, Wynn Wolfe, Chuck Champlin, Molly Feathers, and Mike Boughton. The summer students on the Hrothgar Project were Will Cathcart, Tyler Hunt, and Nick Sun. Finally, e-mail exchanges with Gerhard Fischer, Paul Horowitz, Roscoe Giles and Gordon Wells were extremely helpful and hopefully mark the beginning of longer term collaborations at some level.

References

[1] The CACR WWW site: www.cacr.caltech.edu .

[2] The Beowulf WWW site: www.beowulf.org .

[3] The NPACI-EOT WWW site: www.npaci.edu/Outreach .

[4] National Research Council, *National Science Education Standards*, National Academy Press (1996). WWW site: books.nap.edu/html/nses/html .

[5] The Hrothgar WWW site: oscar.cacr.caltech.edu/Hrothgar .

[6] The Airshed WWW overview: www.eng.uci.edu/~maeadmin/Faculty/dabdub/dd1.html .

[7] D. Dabdub and J. Seinfeld, 'Air Quality Modeling on Massively Parallel Computers', *Atmospheric Environment* **28**, 1994.

[8] The RiverWeb WWW site: www.ncsa.uiuc.edu/Cyberia/RiverWeb .

[9] *Widget Workshop: The Mad Scientist's Laboratory* by Maxis. There is an on-line review and brief description available at the www.superkids.com site.

[10] The StarLogo WWW site: lcs.www.media.mit.edu/groups/el/Projects/starlogo .

[11] The Whyville WWW site: www.whyville.net .

[12] The Terra/Kaon WWW site: www.terra.kaon.com .

[13] D. Tapscott, *Growing Up Digital: The Rise of the Net Generation*, McGraw-Hill (1998).

[14] See, for example, M. Hedegaard, 'The Zone of Proximal Development as Basis for Instruction' in *An Introduction to Vygotsky*, ed. H. Daniels, Routledge, 1996.

[15] J. M. Epstein and R. Axtell, *Growing Artificial Societies: Social Science from the Bottom UP*, Brookings Institution Press, 1996.

[16] The American Association for the Advancement of Science, *Blueprints for Reform* (Project 2061), Oxford University Press, 1998.

[17] C. Bereiter, *Education and Mind in the Knowledge Age*, in press. A draft is available online at csile.oise.utoronto.ca/edmind/edmind.html.

[18] G. Fischer, 'Lifelong Learning - Making Learning Part of Life', presentation slides available at www.cs.colorado.edu/~l3d/presentations/gf-nsf-9.95 .

[19] The L^3D WWW site: www.cs.colorado.edu/~l3d .

[20] G. Fischer, 'Conceptual Frameworks and Innovative Computational Environments in Support of Self-Directed and Life-Long Learning', L^3D report. (Document available in pdf form at www.cs.colorado.edu/~gerhard/reports.html .)

[21] A. Sidorkin, *Beyond Discourse: Education, the Self, and Dialogue*, State University of New York Press, 1999.

[22] G. Wells, *Dialogic Inquiry: Toward a Sociocultural Practice and Theory of Education*, Cambridge University Press, 1999.

[23] The OISE/UT WWW site: www.oise.utoronto.ca .

[24] The IRL WWW site: www.irl.org (Institute for Research on Learning, Palo Alto).

[25] The EduTech WWW site: www.cc.gatech.edu/edutech (Georgia Institute of Technology).

[26] The DICEP WWW site: www.oise.utoronto.ca/ ctd/DICEP .

[27] G. Chang-Wells and G. Wells, 'Modes of Discourse for Living, Learning and Teaching', in *International Action Research and Educational Reform*, ed. S. Hollingsworth, Falmer Press (1997).

[28] Los Angeles Times, *A Unique School or Out of Step?*, Column One article, September 2, 1999.

[29] The PSEware WWW site: www.extreme.indiana.edu/pseware .

[30] E.N. Houstis, et al., *Problem Solving Environments for Computational Science*, IEEE Press (in press).

[31] The Globus WWW site: www.globus.org .

[32] P. Messina et. al., 'Distributed Interactive Simulation for Synthetic Forces', *Proceedings of the International Parallel Processing Symposium*, Geneva, 1997.

[33] S. Brunett and T. Gottschalk, 'A Large-Scale Metacomputing Framework for the ModSAF Real-Time Simulation', *Parallel Computing* **24**. (1998).

[34] The ModSAF/OneSAF WWW site: www.modsaf.org .

[35] M. Resnick, *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds*, MIT Press, 1994.

[36] R. Axelrod, *The Evolution of Cooperation*, Basic Books, 1984.

[37] T. Schelling, *Micromotives and Macrobehavior*, W. W. Norton and Company, 1978.

[38] T. Schelling, *The Strategy of Conflict*, Harvard University Press, 1960.

[39] J. Dahmann and D. Wood, 'Scanning the Special Issue on Distributed Interactive Simulations', *Proc.IEEE* **83**, 1995, and references therein.

[40] The CBS WWW site: stricom.army.mil/PRODUCTS/CBS .

[41] The World Game WWW site: www.worldgame.org/networldgame .

[42] The SPEEDES WWW site: www.ca.metsci.com/speedes .

[43] The Digital Sky WWW site: www.cacr.caltech.edu/SDA/digital_sky.html .

[44] The GIOD WWW site: www.cacr.caltech.edu/SDA/giod.html .

[45] The Dynasites WWW site: www.cs.colorado.edu/~ostwald/Dynasites.html .

[46] The Web-SMILE WWW site: www.cc.gatech.edu/edutech/projects/lbdsoft_ws.html .

[47] M. Scardamalia and C. Bereiter, 'Computer Support for Knowledge-Building Communities', *Journal of the Learning Sciences* **3**, 1994.

[48] M. Scardamalia and C. Bereiter, 'Schools as Knowledge Building Organizations', in *Today's Children, Tomorrow's Society*, eds. D. Keating and C. Hertzman, Guilford (1999, in preparation).

[49] J. Hewitt and M. Scardamalia, 'Design Principles for the Support of Distributed Processes', *Educational Psychology Review* **10**, 1998.

[50] A. Repenning, 'Programming Substrates to Create Interactive Learning Environments', *Journal of Interactive Learning Environments* **4** (1994).

[51] The AgentSheets WWW site: www.agentsheets.com/home.html .

[52] The Sustainable Neighborhoods WWW site: www.cs.colorado.edu/~l3d/systems/mrrogers .

[53] C. Hughes and J. Moshell, 'Shared Virtual Worlds for Education: the ExploreNet Experiment', *ACM Multimedia Systems* (1995), on-line at longwood.cs.ucf.edu/ ExploreNet/papers/VA.Experiment1195.html .

[54] The EVL/Tele-Immersion WWW site: www.evl.uic.edu/cavern .

[55] G. C. Fox *et al.*, 'Integrating Web, Desktop, Enterprise and Military Simulation Technologies To Enable World-Wide Scalable Televirtual (TVR) Environments', available on-line at www.npac.syr.edu/users/timucin/papers/WETICE98 .

[56] *SimCity*, by Maxis. WWW: www.simcity.com/home.shtml .

[57] T. Oppenheimer, 'The Computer Delusion', article in *The Atlantic Monthly*, July 1997. The article is available on-line at www.theatlantic.com/issues/97jul/computer.htm .