Grand Challenges Conference Application

Next-Generation Educational Software and Next-Generation User Interfaces

Andries van Dam

Computer Science Department, Brown University

• Clip Models: A Genre of Next-Generation Educational Software[1]

Over forty years ago, Jerome Bruner proposed a radically new theory of education, that "...any subject can be taught effectively in some intellectually honest form to any child at any stage of development." [1]. While many have disputed the more extreme claims attached to that hypothesis, it is anadmirable goal, though one difficult to achieve. One way to implement this theory is through the “spiral approach to learning” common to formal education where a learner encounters a topic multiple times throughout her education, each time at an increasing level of sophistication.. Furthermore, at any stage in one’s learning, one should be able to mix and match educational modules at different levels of sophistication within the same general topic area. Simpler modules can serve as overviews to a subject for review or to provide context when the intent is to go more deeply into related topics.

Materials on the Web today are not designed to accomplish Bruner’s stated goals. Instead, they are generally designed for targeted groups of students, such as those taking an introductory calculus course. Even more limiting, educational Web content is too-often based on repurposed traditional classroom materials (e.g., putting one’s course notes online) and their associated traditional pedagogical approaches (e.g., lectures). One of the key reasons for the lack of innovative contents is the absence of teacher-usable (as opposed to professional-usable) tools for designing beyond text. In addition, educational content is also designed to run on low-end computer platforms, instead of looking forward to the inevitable arrival of exponentially more powerful platforms.

Within the varied domains and genres that must be part of any future educational system, we propose a particular genre of educational content that is focused on the STEM (Science, Technology, Engineering and Mathematics) domain, oriented toward the platforms with which we expect to be working, and is, in keeping with Bruner's philosophy, appropriate for diverse audiences. We have coined the term "clip model" to help define a genre of interactive, web-based academic content. Clips models are analogous to clip art, in that they are ready-made and are meant to be embedded in a presentation or other framework. Unlike clip art, clip models incorporate both geometry and simulation- or rule-based behavior. They are usable as stand-alone learning objects and, more importantly, unlike clip art, are designed *ab initio* to be combined into more complex models. Each clip model (or combination of clip models) is an interactive microworld meant for constructivist exploration and experimentation in the Kay and Papert sense.

The varied needs of audiences at many different levels of sophistication preclude a one-to-one mapping between a given concept (such as circulation of blood through the cardiovascular system) and a single clip model. Thus, instructional designers must think not in terms of creating a single clip model for a given topic in a field, but of creating one or more elements of a family of interrelated clip models that cover a broad range of explanations and their representations. These models must correctly reflect the ontology and semantics of the subject matter at each point along the multiple axes of age, knowledge level, task, and individual learning style (in Howard Gardner’s sense of "multiple intelligences" [2]). Adding yet more complexity, we must accommodate the variety of learning environments in which such clip models will be presented. These innovative and, by their nature, emergent, learning environments must be made available online and onsite, synchronous and asynchronous, virtual and real classrooms, servicing both single on-demand learners and collaborative learners, either in impromptu virtual study groups, or in formats yet to be defined. Another dimension we need to explore more deeply is team collaboration; we know all too little about effective group learning using digital media and how to extend that learning into further learning in school or the workplace. Clearly, these requirements lead to a huge challenge in instructional design and learning technology.

All these pedagogical needs of clip models are a complicating factor that makes their design immensely harder than that of ordinary components in the standard software engineering sense. A potential approach to thinking about the problem may be to use an extension of the MVC paradigm of Object-Oriented Programming to describe the necessary interrelationships between these different concept representations. Each concept or real-world object must be represented by a multitude of models (e.g., the heart as a pump, the heart as a muscle, the heart as an organ in the chest cavity), at widely different degrees of

[1] I gratefully acknowledge the substantive input of Rosemary Michelle Simpson and Sascha Becker, Brown University and Randy Hinrichs, Microsoft Research.

sophistication. Each model supports multiple views (e.g., simplified 3D models, realistic 3D models, 2D schematics, the sound heard through a stethoscope), and for each view multiple controllers that may present a learner-chosen UI style. Multiple models geometrically complicate the single-model paradigm of classic MVC. To add to the challenge, individual clip models must interact with other clip models at different levels of abstraction, still reflecting the complex interactions between the subsystems of the systems being modeled. For example, how can a heart model at a highly detailed level of simulation interoperate with a vascular system model at a much simpler level that does not expose or even represent the full set of parameters that the heart model needs? Or, worst case, must each model provide a simulation at the deepest level of understanding possible, hide unneeded complexity, but retain the ability to expose parameters that must be interchanged with models of related subsystems? While designing and building a family of clip models, the authors must plan how each model will be combined with other models down the road. This combinatorial explosion presents a huge technical challenge which, as far as I know, has not been targeted, let alone solved by the software engineering community.

Further, consider the ontological issues involved: when you have one, or at most a small team of authors writing a single book targeted at a single audience, the domain specification as seen in the definition and relationships of concepts and terms is an important but manageable task. When you expand the context as described above, the situation becomes orders of magnitude more intractable, closely related to the massive undertaking that the controversial CYC project embarked on twenty years ago by Lenat et al. The Knowledge Web community is now starting to tackle the problem of identifying and encoding domain-specific ontologies for the Web;. in [3], Holsapple and Joshi describe a collaborative approach to designing an ontology that begins with independent ontological proposals from several authors and incorporates input from many contributors. Some sort of collaborative approach to ontological engineering will have to be used in order to build an ontology which is acceptable to many members of a given field.

This intersection of simulation science, software engineering, ontology building, instructional design, and user interface design comprises the technological aspect of this Grand Challenge problem. In addition to the technological challenges, there are intensely interdisciplinary organizational challenges reminiscent of the Manhattan project: building clip models is essentially a new design discipline that requires collaborative teams of experts from cognitive science, social sciences, arts and design, story-telling media, information structure design, teachers, domain experts, and computer scientists.

A very ambitious first step towards realizing this vision is being made with the Digital Human project [4].To realize this kind of high-quality next-generation educational content, across all disciplines, and at all levels, will require a Grand Challenge-level effort, arguably on an even larger scale than previous Grand Challenge projects, such as the Manhattan Project, the Man on the Moon project, and the Human Genome Project.

Indeed, reforming education in all of its dimensions, is even more overwhelmingly ambitious than trying to create clip models as merely one genre of next-generation educational software. Nonetheless, a start <u>must</u> be made, as a national, indeed global, imperative for survival. The start I'm proposing here is a first step in the quest of building next-generation educational content and tools, including clip models. A small steering group that I helped found is proposing the creation of a nonprofit, industry-led, foundation, modeled on the highly successful Sematech Consortium, and, for now, called the Learning Federation [5]. The Federation's purpose will be to provide a critical mass of funding for long-term basic and applied pre-competitive research in learning science and technology, conducted by interdisciplinary teams. This research is meant to lead to the development not only of next-generation authoring tools but also of exemplary curricula in the broadest sense, to be used for both synchronous (e.g., classroom-based) and asynchronous (e.g., distance, on-demand) learning.

---

[1] Jerome Bruner. *The Process of Education*, p.33, Harvard University Press, 1960.

[2] Howard Gardner. *Frames of Mind: The theory of multiple intelligences.* New York: BasicBooks. BasicBooks Paperback, 1985. Tenth Anniversary Edition with new introduction, New York: BasicBooks, 1993.

[3] Clyde W. Holsapple and K.D. Joshi. "A Collaborative Approach to Ontology Design", in Communications of the ACM, February 2002.

[4] Digital Human project: http://fas.org/dh/

[5] Learning Federation: http://www.learningfederation.org/

****

• Next-Generation User Interfaces for Control of a Federation of Digital Devices and Agents

Next-Generation User Interfaces. The user interface is the means by which the user interacts with and controls a computing environment. Shifts in computing environments both enable and mandate paradigm shifts in user interfaces. Earlier watershed shifts included that from batch computing to personal computing and, in personal computing, from textual command lines on character terminals to WIMP GUIs based on bit-mapped raster graphics workstations. Now, massive changes in computing, communications, and infrastructure technologies require an even deeper shift, one whose effective solution constitutes a Grand Challenge.

Future computing environment. In the position papers presented for this conference, there is remarkably little disagreement about our future hardware/software environment: ubiquitous, pervasive, reliable, secure, with an explosion of form factors liberating us from one-size-fits-all desktop or laptop devices [Abowd, Bell, Birnbaum, Borriello, Gehrke, Hellerstein, Jain, Ramachandran]. An individual user, instead of having one or several computers, will have hundreds, potentially thousands, in a federation of devices, many of which need to know about each other. Some will be embedded in one's body, clothing, furniture, and vehicles; others will be hand-carried, such as PDAs; still others will be wall-sized displays anchored to the physical environment. Less agreed on is the availability of immersive virtual reality (IVR) and augmented reality (AR), but I remain optimistic that they, too, will take their rightful place in the spectrum.

Infrastructure. From an infrastructure point of view, also agreed upon [Bell, Borriello, deFanti, Edwards], the computing environment will be widely heterogeneous and distributed, requiring new forms of operating systems, middleware, and applications built on-the-fly from components. Distributed approaches to the design of middleware and applications parallel the physical dispersal of devices, replacing the monolithic client-server model. These fine-grained components will exist in the network as a set of services to be assembled and integrated on demand. Moreover, many of these devices and services will be directed by autonomous agents rather than explicit user control. Devices and agents within an individual user's federation often will need to communicate with each other and to work collaboratively with other users' federations.

The human element. Obviously, a large part of the grand challenge of controlling this anticipated federation of one's own and others' devices and agents lies in resolving the fearsomely complex systems issues. An equally important part, I maintain is the notion of the user as a key component of the total system. Although the role of the user is acknowledged the HCI community [1, 2, 3] and by other participants [Birnbaum, Gehrke, Goldberg, Kay, Scholtz, Soloway & Norris], and the HCI community does seem to be shifting towards a more holistic view of the user's involvement with the overall computing environment,[2] I believe that inadequate emphasis has been placed on research in user experiences, in relationship to user interaction styles, and user interfaces. By user experience, I mean the overall psychophysical-perceptual-cognitive-aesthetic experience of engaging with the computer environment; by user interaction style, I mean the ways in which the user manifests her specific mode of working within a particular user experience (e.g., choosing among and within textual, visual, oral/aural etc., styles of communication); by user interface, I mean the specific means for implementing particular user interactions, i.e. the traditional look-and-feel.

The first aspect of the human element of the grand challenge is how an individual user controls this myriad of devices and agents without being overwhelmed. How do we deal with peripheral and situational awareness when there are thousands of asynchronous activities and dozens of tasks that we are explicitly and synchronously controlling, with distinctly different forms of alerting/notification? Any proposed solution probably will have to go well beyond what even the best human assistant can do to help an individual user.

The second aspect is the evolution from HCI (Human-Computer Interaction) to what I call HHI (Human-Human Interaction) mediated by computers. Doug Engelbart, in his landmark 1968 presentation of NLS, demonstrated live telecollaboration, sharing floor control with a collaborator at a remote site. More than three decades later, we have made little progress. The simplistic notion of a user communicating with one or

[2] as can be seen in the interesting conference on Design for Usability held in London two years ago [4].

a few tasks and one or a few collaborators must shift to the more complex notion of a user communicating with many other users, intermediated by devices, agents, smart applications and the like. How do we interact with multitudes of collaborators, using multiple federations as our intermediaries? It is obviously beyond any current system design to have mathematically complete connectivity of all devices and agents in the 'cloud' of federations, so new organizing principles and systems will have to be designed.

Note that I am not suggesting that traditional HCI is now a mature discipline or that more effort should not go into HCI. In particular, we need much more effort on post-WIMP, multi-modal interaction styles, which haven't progressed much or penetrated deeply, except for primitive gesture recognition in PDAs and voice recognition in telephone-based information services. Despite that, we have to start shifting our focus to researching device- and agent-mediated HHI. It's a larger problem that encompasses HCI, one that must be informed by what we know about how individual humans operate, and beyond that, how teams of humans communicate, experience, and collaborate.

It is a source of considerable concern and frustration to me that while computer graphics is now a recognized specialty, albeit not a core one in computer science departments, HCI is still an elective, at best, in most CS curricula and may not even be available as a regularly-taught specialty. Considering that computing environments and content in the future will be first and foremost about providing rich user experiences, it is distressing that the next generation of researchers and developers typically have essentially no knowledge of the human relationship to the system.

[1] Jakob Nielsen. Designing Web Usability: The Practice of Simplicity. New Riders Publishing, Indianapolis, 2000.

[2] Jenny Preece, Yvonne Rogers, and Helen Sharp. Interaction Design: beyond human-computer interaction. John Wiley & Sons, Inc. 2002.

[3] Ben Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Third Edition, Addison-Wesley, 1998.

[4] Design for Usability conference. http://www.design-agenda.org.uk/DFU/

Grand Challenges Conference Application

Next-Generation Educational Software and Next-Generation User Interfaces

Bio

Andy van Dam is Thomas J. Watson, Jr., University Professor of Technology and Education and Professor of Computer Science at Brown University where he has been since 1965. He co-founded the department of Computer Science in 1978 and was its first chairman. His principal research interests are in computer graphics, computer-based education, and hypertext and electronic books. In 1967 he co-founded ACM SICGRAPH, the predecessor to SIGGRAPH. He has authored or co-authored some 90+ papers and 11 books, including the well-known reference work: "Computer Graphics: Principles and Practice" (with J. Foley, S. Feiner and J. Hughes), Addison-Wesley (1990).

He is a member of the National Academy of Engineering and the National Academy of Arts and Sciences, and has honorary doctorates from Darmstadt Technical University and Swarthmore College. He is a fellow of the ACM and the IEEE. Among his other awards are the SIGGRAPH Steven A. Coons Award, the ACM Karl V. Karlstrom Outstanding Educator Award, the IEEE James H. Mulligan, Jr. Education Medal, the ACM SIGCSE Award for Outstanding Contribution to Computer Science Education, the CRA Distinguished Service award for 2002 and the Brown University Sheridan Teaching award. He is on the technical advisory board of Microsoft Research and multiple startups, and was a founder of Electronic Book Technologies, Inc.