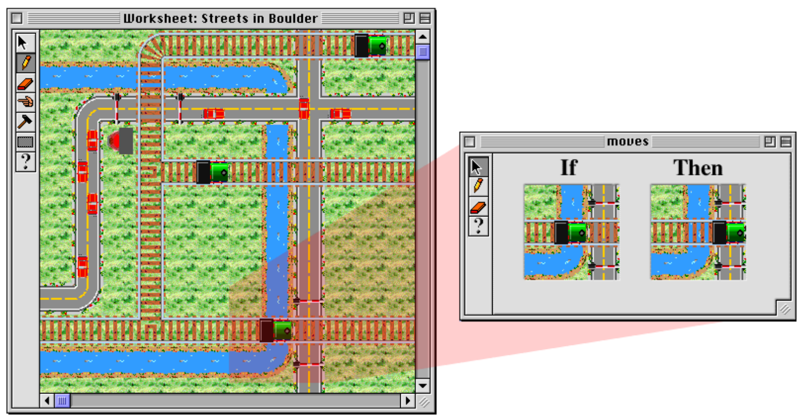
Project Portfolio

Repenning's research and teaching interests revolve around the goal of empowering end-users by combining human skills with computer affordances. How can information technology help people to learn better, gain deeper insights and communicate more effectively? This work brings together end-user programming/development – *as means for people to express ideas computationally* – with multimodal interfaces – *utilizing sound, music, touch interface, speech output, speech recognition* – and networked services – *accessing and interacting wirelessly with large location-aware information spaces* – into a new form of multimedia. A chronological portfolio is provided to display a research and teaching trajectory moving from early educational applications of end-user programming towards an integration of end-user programming with virtual reality, scientific visualization, sound and speech processing, 3D computer graphics, computer animations, and artificial life.

AgentSheets

1991 AgentSheets introduced the use of graphical rewrite rules to program the behavior of agents. As a form of end-user programming, graphical rewrite rules are pairs of before/after pictures edited through demonstration by users. For instance, the behavior of a train agent to follow train tracks is programmed by simply demonstrating an example of how the train moves to the right. AgentSheets evolved and introduced additional end-user programming paradigms including programming by analogous examples and tactile programming. With the support of the National Science Foundation AgentSheets has become a commercial product.

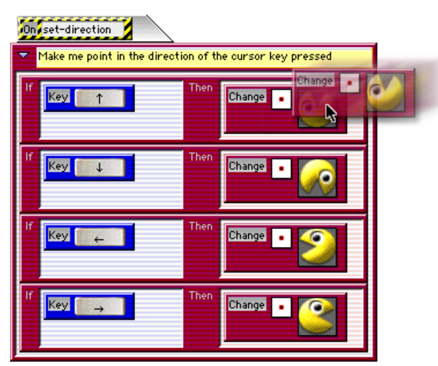


LEGOSheets

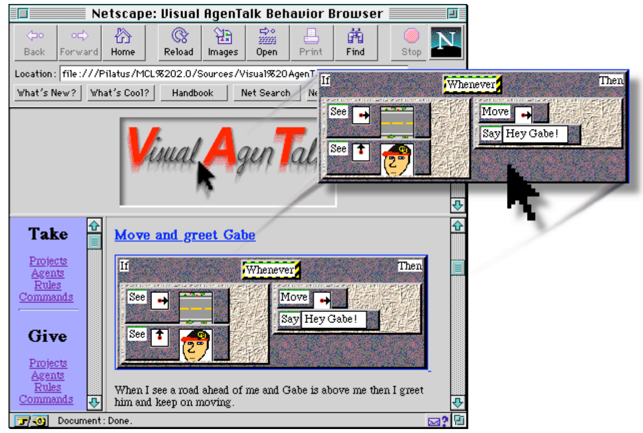
1994 Rule-based end-user programming works well also for real-time and embedded applications. The AgentSheets-based LEGOSheets became the first visual programming environment for the MIT programmable brick. This brick was the precursor to the LEGO Mindstorm. *Examples*: kids building road tracing cars based on optical sensors; artist exploring interactive art in which a vehicle reads the brush strokes of the artist and, at the same time, does it's own collaborative painting with a pen.

Visual AgenTalk

1995 To enable the creation of more sophisticated games facilitating artificial intelligence and of computational science applications graphical rewrite rules proved to be insufficient. Visual AgenTalk is a groundbreaking visual programming language supporting the composition, comprehension and sharing of programs. Composition is supported through drag and drop interfaces preventing syntactic programming errors. Comprehension is promoted through a combination of animated program annotations and speech synthesis allowing users to select program fragments in order to have the system explain the meaning of that fragment. This real-time explanation takes parameter settings defined by the user into account in order to spell out the consequences of these settings. The sharing aspect of Visual AgenTalk empowered users to drag program fragments, or entire agents, directly from web pages into their games.



Behavior Exchange



1996 The Behavior Exchange would today be called a Web 2.0 application. It was a social networking site allowing AgentSheets users to freely share projects and project components including individual actions, conditions, rules, methods, agents, and even entire projects. The submission and download process were directly integrated into AgentSheets. Educational applications included collaborative simulations building. For instance, students would explore eco systems by designing animals and plants that they shared in their class room via the Behavior Exchange. Then they would experiment with eco world combinations by downloading and mixing different animal species. Agents downloaded could be further programmed and shared back.

Ristretto

Ristretto

1997 Ristretto© was one of the first Java byte-code compilers outside of Sun. At the press of a single button Ristretto turns entire simulations including sounds, images, and behaviors instantly into a complete Java applets, Java Beans or Flash. Ristretto is an efficient compiler that can be used for computational science applications with ten thousands of agents. For end-users, Ristretto means the ability to create and publish interactive web-based simulation without the need to learn traditional programming. The ability of turning AgentSheets simulations into Java Bean components is important for collaborative projects as it allowes the combination of AgentSheets simulations with 3rd party components such as spreadsheets into complete educational activities.

```
 0 aload_0
 1 iconst_0
 2 iconst_0
 3 invokevirtual #16 <uAgent
 6 ifeq 18 (+12)
 9 aload_0
10 iconst_4
11 iconst_0
12 invokevirtual #11 <uAgent
15 goto 52 (+37)
18 aload_0
19 iconst_0
20 bipush 15
22 invokevirtual #16 <uAgent
25 ifeq 37 (+12)
```

Educational Software Components of Tomorrow

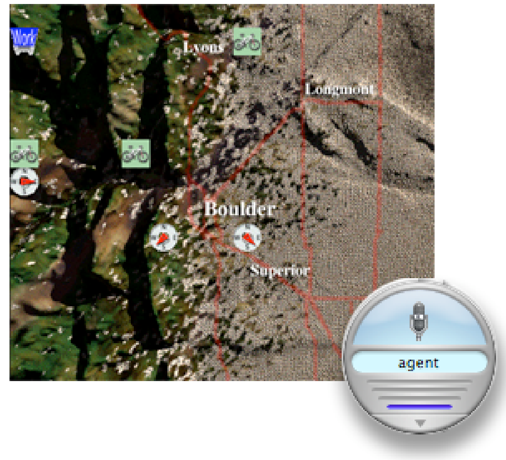
1999 Educational Software Components of Tomorrow (ESCOT) is a testbed for the integration of innovative technology in middle school mathematics. The project investigates replicable practices that produce predictably high quality digital learning resources. Researchers from SRI International, the Mathforum, and the University of Colorado created activities based on educational components. AgentSheets was used as one of the main component generators. The figure shows an AgentSheets simulation component (left part) and three pie chart visualization components (right part) connected into one activity teaching students about ratios.

Game Design for Education

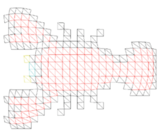
2002 Game design can be a highly educational and engaging activity. In the context of building working games, design is experienced as a rich, creative, and often social exploration process of complex decision spaces. Game design for education is not just about games but about computational literacy. Students acquire strategies to cope with complex, open-ended problems. We have employed game design for education in settings ranging from elementary schools to computer science graduate school. The toughest challenge originated from the Japanese Ministry of Education to teach Japanese middle school students to make video games in a single day. In the morning AgentSheets (Japanese version) was used to introduce basic programming concepts. In the afternoon students could design and build their own video game. Every child produced complete games, which they showcased to the other workshop participants at the end of the workshop.

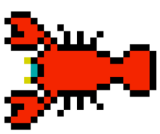
Boulder Mountain Bike Advisor: The Pragmatic Web

2003 The Pragmatic Web is a conceptual framework integrating end-user programming through multi-modal interfaces based on agents with the Web. Users extract information from existing Web-pages and process that information via end-user controlled computation. At the same time users define how to access that information. The Boulder Mountain Bike Advisor is a speech controlled Pragmatic Web application. A user asks “Where should I go mountain biking.” Several agents located on a map of Boulder County react to this voice command. These agents are representing locations that are possible candidates for biking and also feature real time, Web accessible weather information sensors. Rules previously defined by the users capture pragmatic interpretations. For instance, an agent may reply (using speech output): “It’s really nice up here at Betasso but you should bring a jacket because it’s a little windy”.



Inflatable Icons



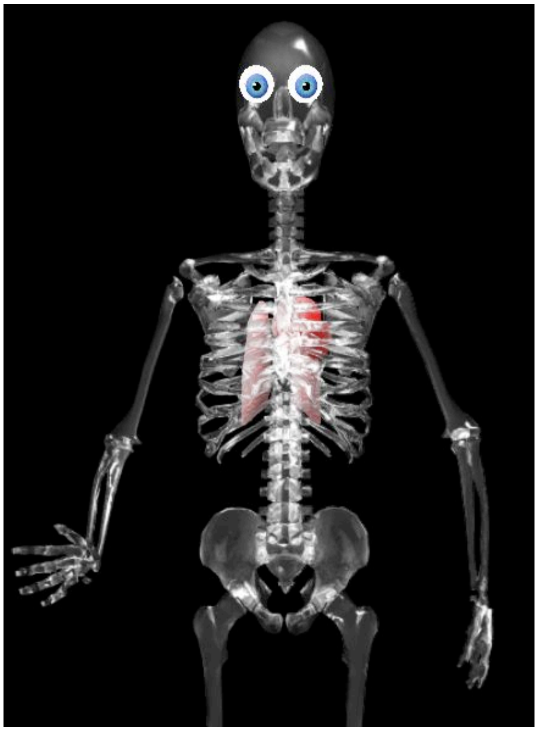


2005 End-user development is not limited to programming. Users of 3D virtual worlds and simulations occasionally need to create their own 3D models. Existing tools are geared towards high end, production quality 3D content. Very few end-users have the necessary time to deal with the steep learning curves of these tools. Inflatable Icons, in contrast, are based on new interactive techniques that help end-users to author intricate 3D objects by extruding them automatically out of simple 2D pixel-based images. User controllable AI extrusion algorithms vectorize images and inflate the flat images into objects usable as 3D agents and avatars.

Mobility Agents

2006 Navigating through a city using public transportation can be a challenge. An effort sponsored by the Coleman Institution helps persons with cognitive disabilities to use public transportation. 30 Global Positioning System equipped busses in Boulder Colorado are tracked wirelessly by agents. Travelers out in the field use mobile devices such as PDAs or cell phone to receive navigation prompts helping them to find the right bus and, once inside the bus, help them to exist the bus again at the right time. A 3D visualization allows care givers to see the current location of busses and GPS equipped bus users. They can watch in real time, play recorded data and assume different camera positions.

Mr. Vetro: A Collective Simulation Framework

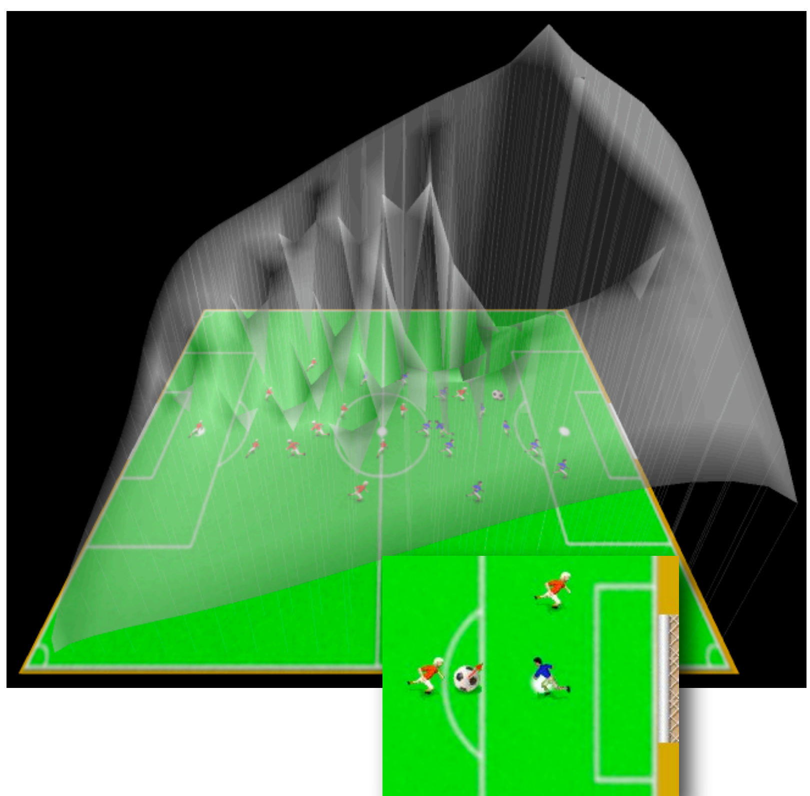


2006 Collective Simulations integrate *social learning* pedagogical models with *distributed simulation* technical frameworks. Mr. Vetro is a Collective Simulation application used to learn about human physiology. Mr. Vetro is based on the C^5 architecture (compact, connected, continuous, customizable, and collective), which has been specifically designed to allow students to immersively experience interdependent complex systems found in biology, chemistry, ecology and economy. To simulate Mr. Vetro students have to collaborate with each other through scientific discourse. Each group of students is handed one organ simulated on a PDA. All the organs are wirelessly connected to the master simulation – Mr. Vetro – where the systems are shown interacting with each other. The master simulation, enhanced by medical instruments showing physiological variables such as O_2 and CO_2, are projected to the entire classroom in order to reflect the global consequences of all the local decisions.

Collective Simulations are examples of Ambient Intelligence in the sense that they are based on a typically large number of small, wirelessly connected computers. These computers are only used to facilitate and augment discourse, not to replace it.

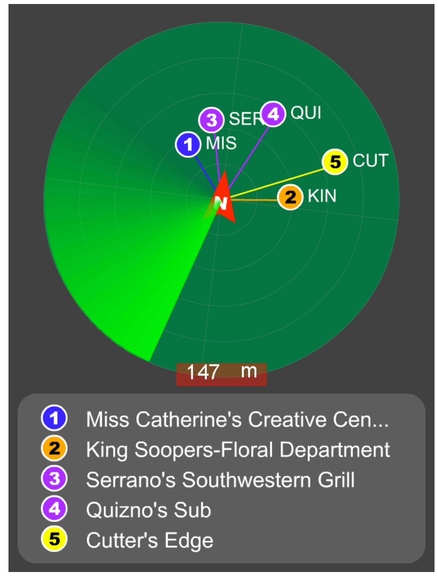
Collaborative Diffusion

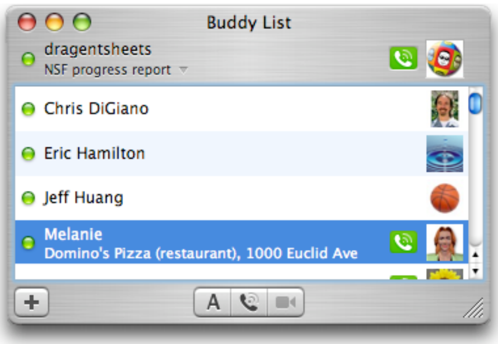
2006 Some problems cannot be solved with rule-based approaches alone. Complex problems may need potentially large numbers of agents that can collaborate and compete with each other in real-time. Collaborative Diffusion is a new multi-agent framework that integrates a symbolic level, based on rules, with a sub-symbolic level, based on continuos values diffusion processes, into a robust intelligence approach. Collaborative Diffusion is well suited for applications in which a large number of agents continuously engage in distributed problem solving. Goals are described as continuous functions diffused over discrete spaces. Competition and collaboration are expressed through diffusion modulation functions. In the soccer application, Collaborative Diffusion has resulted in emerging collaborative behaviors including sophisticated ball passing strategies. We have started to use Collaborative Diffusion in transportation applications to deal with unreliable and noisy sensor-based information.



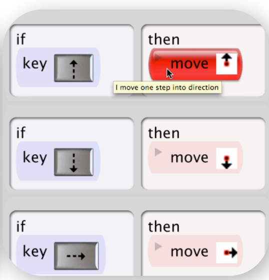
Urban Radar

2006 The Urban Radar is an extension to the Mobility Agent project. Geared at the urban explorer the Urban Radar maintains a display of static and dynamic objects of interest. Showing these objects in accordance to a personal preference, users are able to discover interesting objects just in time. Static objects such as restaurants are found by correlating the explorer's own GPS location with the location of objects found in an extensive Geographic Information System. The Urban Radar may, for instance, find a nearby Italian Restaurant at lunch time. In addition to the location of the restaurant the Urban Radar will offer the lunch menu if the restaurant features a Web site. Moving objects, such as busses or other people, can be visualized in the same space. Alarms and privacy control are based on end-user programming. Using an agent-based integration between Internet Messaging (IM) and Location-Aware services, users can share their location dependent status with others via regular IM clients.



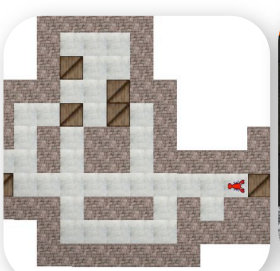


Interactive 3D for Everyone: AgentCubes



2006 AgentCubes makes authoring interactive 3D accessible to everyone by integrating revolutionary end-user tools for 3D modeling, animation and programming. Through its incremental 3D approach, AgentCubes is an ideal tool for educational game design and computational science applications: create sophisticated 3D models with Inflatable Icons, add 3D animations and 3D behaviors when you need them. Mobility Agents and Mr. Vetro are two examples of AgentCubes projects.

animation: Parallel Time-Jump
Add sophisticated parallel animations to hundreds or thousands of mutually dependent objects at a flick of a button.

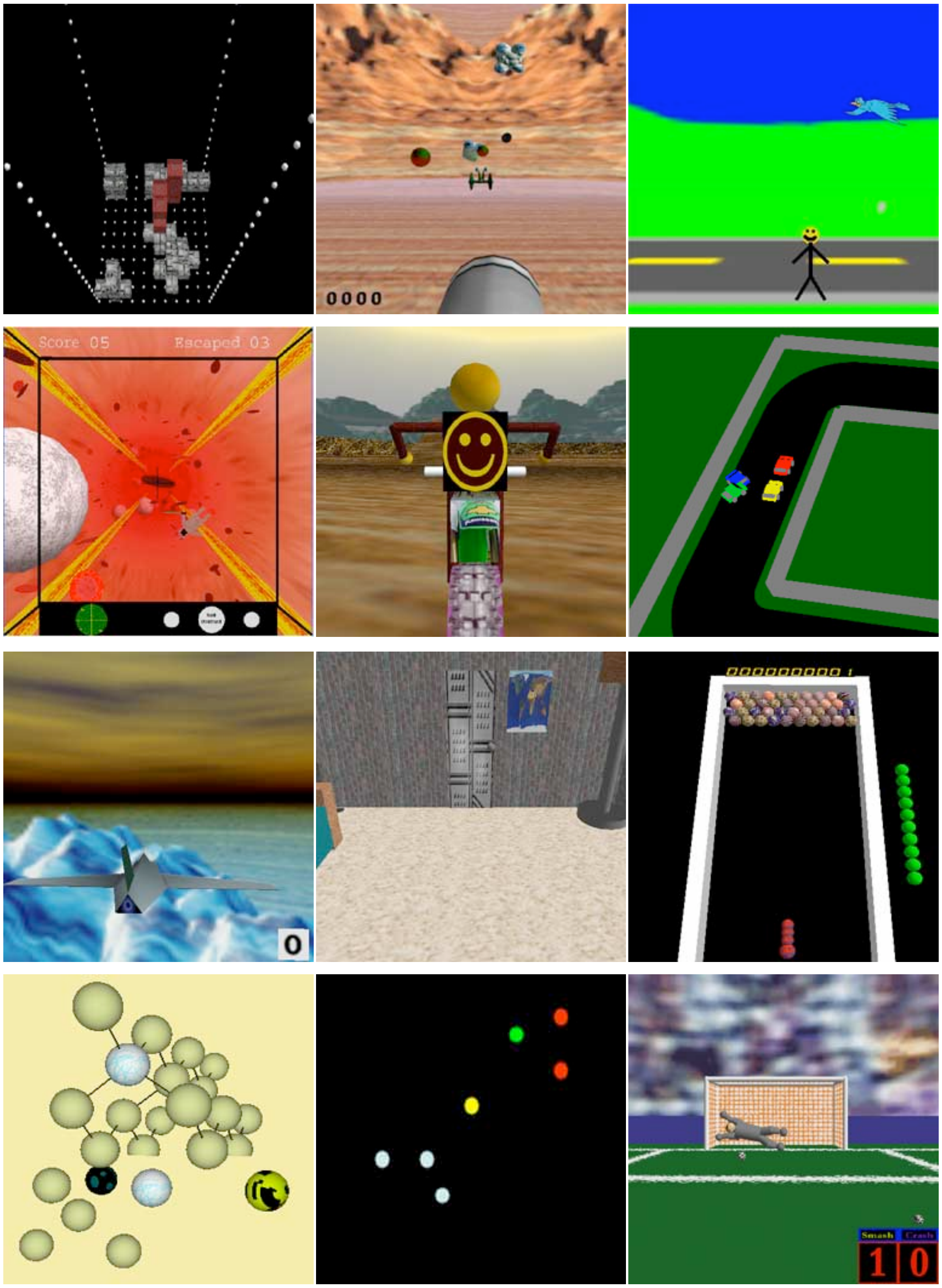


programming: Visual AgentTalk
Gradually move from 2D to 3D programming. Experience global and local coordinate systems by switching back and forth between Bird's Eye and 1st person views.

AGENTCUBES

3D Game Programming and Object-Oriented Design

Teaching and research go hand in hand. An Object-Oriented Programming and Design course is centered around the design and implementation of games. Students learn to work in teams, and work on layered architectures. The very large OpenGL 3D flat library is refactored into an object class hierarchy that includes middleware (game engine) and application (game) layers. The game engine, created by one team, provides functionality such as scene management, event handling, animation control, rendering and simple physics. The other teams produce the applications shown below. The applications use the game engine by subclassing and extending it. Interesting discussions emerge and real design drama unfolds in the classroom resulting from questions regarding where functionality should reside. Should it be implemented specifically in applications or should it (and can it) be generalized and be added to the game engine.



This research has been supported by the National Science Foundation, The National Institutes of Health, the Department of Defense and the European Commission. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the any of these funding organizations.