A Game Developer's Review of
SIGGRAPH 2000: New Orleans
Morgan McGuire <morgan@druids.org>

New Orleans, Louisiana. Home to 'gator swamps and armadillos, Mardi Gras, legal gambling, French Creole cooking, and from July 23-28, the SIGGRAPH 2000 conference. Here, the only thing hotter than the weather and the cuisine is the technology. Scientists with graduate students in tow, game developers, artists, film makers, software developers, and hardware vendors converge on the Morial Convention Center to showcase their latest work, share techniques, and keep up with the latest advances.

Who's hiring? See the list of 3D job openings at the bottom of this article.

SIGGRAPH is home to computer graphics in all its forms. Talks and presentations cover such disparate topics as a Pixar film, the mathematics of special relativity, terrain for video games, and techniques for generating effects in a ray tracer. To cover all of these areas, the conference is divided into venues:

- **Courses, Papers & Panels, and Birds of a Feather Sessions**
  Targeted at researchers, these discuss new techniques and present the latest graphics research.
- **Exhibition**
  A trade show with the latest graphics technologies. The booths are elaborate multi-story constructions where sales staff and engineers are available to discuss their products in detail and give away T-shirts, demo CD's and other geek sales toys.
- **Art & Emerging Technologies Gallery**
  Star Trek meets Saturday Night Live's Sprockets skit in the gallery. The gallery is a giant warehouse room draped in black cloth, haunted by mysterious sounds and lights. Exhibits vary from quivering lumps of latex and data-capturing construction toys to augmented reality laser-tag and new PS2 technologies. A major challenge for visitors was to determine whether each display was intended art or emerging technology (the Wooden Mirror was one of the few that succeeded as *both*; many failed to be either).
- **Animation Festival & Electronic Theater**
  Animation galleries showcasing the industry's best animations. Playstation II FMV's, special effects sequences from The Matrix, A Perfect Storm, and other movies, and animated shorts by Pixar as well as independents. Some of the more eclectic short films made the art gallery look normal, but most were impressive films enabled by technology rather than ruined by it.

  The audience favorite: Robert Rioux's **Block Wars** Lego Star Wars parody. Darth Vader admonishes Luke, *I'm your father... and when I was your age, I single handedly defeated the trade federation and was the only human to ever win a pod race... what do you have to say for yourself? And get a hair cut* Check out Rioux's other work.

What was the hot technology this year at SIGGRAPH? Image Based Modeling and Rendering, Level of Detail, Games, Photon Maps, Geometric Algebra and the Wooden Mirror.

Image Based Modeling and Rendering (IBMR) is a field that is exploding with interest and promise. Image based techniques use images to replace or augment polygon models, achieving levels of complexity in real time graphics that are not possible with polygon-only models. Texture mapping is a well known technique for modeling details without increasing polygon count. Another approach is the 360 panoramic bubble technique used by web plugins like QuickTime VR and LivePicture, where the user is inside of a texture mapped sphere representing the environment around them.

Panoramic Cylinder
by Leonard McMillan
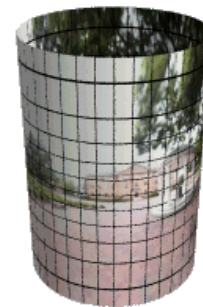
The new image based techniques go far beyond texture mapping or simple panoramas. Two of the most interesting involve light fields and relief textures. Aaron Isaksen, Leonard McMillan, and Steven Gortler presented a paper entitled Dynamically Reparameterized Light Fields describing ways of varying the point of view and focal depth of an existing set of images. They also describe how to make an autostereoscopic light field photograph by placing a hexagonal screen over a specially constructed image (this looks like a very good color hologram with some strange artifacts).

Manuel Oliveira, Gary Bishop, and David McAllister presented a paper on Relief Texture Mapping. The two images shown below are taken from their paper.

Figure 1. Town rendered using conventional texture mapping. The façades and brick walls represented with one texture each.

Figure 2. Same view as in Figure 1 rendered using relief texture mapping. Both scenes contain the same number of polygons. Notice the bricks standing out and the protruding dormers.

Both images are rendered from the same geometry. The left image uses traditional texture mapping, where texture is painted onto a surface. Note that the houses are modeled as cubes. On the angle that the near house is presented, the roof looks terrible because it is not slanted. More subtly, the nearby wall looks flat, as does the facade of the house, since neither the bricks nor the flower boxes stand out when viewed on an angle. The image on the right is also rendered with texture mapping, but the textures are preprocess before the image is rendered. Notice how the roof appears to slant backwards, although the house is still modeled as a cube, and details like the bricks and flower boxes appear to stand out from the surfaces they are attached to. These effects are achieved by applying two 1D transformations to the textures to generate perspective based on known depth information. The first pass simulates vertical perspective and occlusion, the second horizontal perspective and occlusion. After this processing, standard texture mapping primitives may be used for speed.

To learn more about this incredible techinque, read the entire paper and many others at the Relief Texture Mapping website. Also, to learn the basics of image warping techniques, visit Leonard McMillan's introduction to image warping site.

Photon Mapping is a powerful global illumination technique developed by Henrik Jensen. It is easy to implement and runs quickly, but can generate complex illumination effects. These include the soft shadows created by area light sources, color bleeding where bright light reflects off a colored surface, and the focussing of light from reflective or translucent objects known as a caustic.

Prior to the introduction of Photon Maps, expensive techniques like radiosity and monte carlo ray tracing were needed to solve for the realistic lighting of every point in a scene. Radiosity works by bouncing energy emitted from light sources around a scene until equilibrium is reached when each surface emits as much light as it absorbs. This algorithm

can be difficult to implement, typically runs very slowly, and can't produce effects like caustics or those observed when mirrors and glass are present. Traditional ray tracing methods also fail to be realistic when a light source is illuminating a point because photons are reflected through a mirror or focused through a lens. Soft shadows are possible in a ray tracer but they are expensive to compute and tend to appear dithered due to the statistical methods involved.

Photon mapping traces a small number (hundreds of thousands) of photons forward from a light source and models their physical interaction with a scene. Wherever a photon hits a surface, its position is recorded in a 3D "photon map" of the scene. This process continues until all of the photon's energy is lost. This photon map can then be used to produce light maps (2D textures containing illumination data) for real time polygon rendering, or can be used in conjunction with ray tracing methods to improve rendering time and realism.

Photon mapped cognac by Henrik Jensen

Ray tracing experts and enthusiasts at the Ray Tracing Round Table, a Birds of a Feather session, agreed that Jensen's technique is an elegant and practical solution to the global illumination problem for most cases. Niels Christensen and Henrik Jensen gave an in depth talk on implementation details to achieve efficient photon mapping. The examples shown in the talk looked incredible and the photon mapping phase usually occurred in a few milliseconds, suggesting that the technique may be appropriate for real time rendering in games. Jensen's website contains many resources for learning more about Photon mapping and ray tracing.

Games are on everybody's mind at SIGGRAPH. The graphics community has recognized that PCs now have sufficient power to replace aging and expensive SGI machines. They also recognize that at $7.4 billion per year, the games industry is almost as large as the $7.5B film industry and is about to eclipse it.

The trade show floor was filled with the blue and green gleam of PS2 LED's and resounded with audio from PC games like Quake Arena and Everquest. Even many scientific talks referred to Quake II, and research projects frequently used game file formats and engines. Craig Reynolds, from Sony's game research group, hosted a whole day session on games research with Chris Hecker (definition six, inc.), Jonathan Blow (Bolt Action), John Funge (Sony), Robin Green (Bullfrog Productions Ltd.), and Robert Huebner (Nihilistic Software, Inc.).

Sim Theme Park by Electronic Arts

Reynolds was the author of one of the first game-style AI systems. His program, *Boids*, demonstrated that complex group behavior like the flocking of birds could emerge from simple rules governing the behavior of individuals. He opened the game development session with a discussion of his current work at Sony involving advanced Boids-like models. His 1999 Game Developer's Conference paper, Steering Behaviors For Autonomous Characters is available online and discusses this work in detail. Robin Green also spoke on steering behaviors, describing his work on Sim Theme Park and Dungeon Keeper II at Bullfrog.
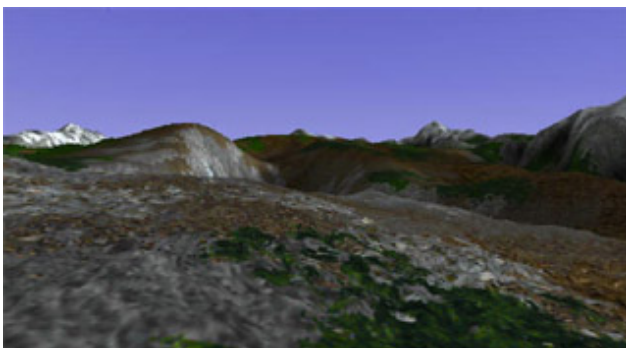
Chris Hecker briefly talked about the intersection between the game development community and the graphics research community. In bare feet, T-shirt and shorts, the youthful Hecker talked a mile a minute, squeezing a lot of content into a short time slot. He described the game development cycle: a few months grabbing the hottest research techniques from conferences like SIGGRAPH followed by an extremely demanding and practical 21 month period of trying to reduce those techniques to practice and failsafe performance. Game developers can take an algorithm out of the theoretical world and add what is needed to make them function in an actual product, but they do not have resources to pursue new directions. Researchers are needed to create entirely new algorithms that will advance the state of the art significantly.

Hecker also discussed the state of physics simulations for games. Physics simulations handle the interaction of objects in a virtual world. Some cases are approachable and can be seen in the industry today. The physics for a well-constrained and well understood model like a race car on a track can be handled. These simulations can only work within a limited domain, however. No race car game today can accurately handle the physics of an arbitrary object like a table fan blowing on a tumbling deck of cards with the engine used to simulate cars. Extremely simple situations, like pool balls, can be handled as well because the complexity is approachable. An ideal physics simulator would handle arbitrary polygon meshes interacting in complex situations like a rock slide or large numbers of other oddly shaped objects. Unfortunately, the numerical issues relating to stability and reproducibility in such a system make it very difficult to approach for game developers.

At the Exhibition, Havok was demonstrating their excellent 3D Studio MAX plugin and licensable game engine, Havok. It performs real time physics on arbitrary meshes, and was easily handling hundreds of oddly shaped objects interacting realistically. The licensing fees are thousands of dollars and so are available to only established professional developers for the engine, but the 3D Studio MAX plugin is available for making cut scenes and canned animations for only $495. I caught up with Chris Hecker and asked him for his impression of Havok, in light of his pessimistic talk on physics for games. He says the Havok development team is great and their product works well. The down side is that using the library may take as much expertise (but not time!) as writing a physics engine from scratch. For a full evaluation, look for his review of Havok in an upcoming Game Developer Magazine, or download the demos from Havok's site yourself.

Level of Detail techniques seek to automatically produce low-polygon count versions of models to speed rendering when a model is very distant or too many models are on screen at once. Jon Cohen gave a good introductory talk on Continuous Level of Detail (CLOD) techniques for models. These techniques are continuous because they remove a single vertex or edge for each step of the algorithm and can thereby produce models of varying complexity in a continuous fashion. The most popular method of CLOD is the Half-Edge Collapse. In this algorithm, an edge is selected for removal and the second vertex of the edge is conceptually to the location of the first vertex, effectively removing the edge because it has zero length. This removes two triangles from the mesh because the triangles along the edge are now degenerate with zero area. This process can be repeated until the model has a desired level of complexity. The Half-Edge Collapse is easy to implement, but tends to produce much poorer representations than other techniques he described and results in visual artifacts when the detail level is changed. Why is it so popular in practice, then?

A panel of game developers gave the answer: it's faster in graphics hardware than other algorithms. Half-Edge Collapses don't change the vertex list for a model, only the face list. This means the vertex buffer stored in hardware (which may include per-vertex data like surface normals and texture data) does not need to be updated when the level of detail changes. With some careful sorting, vertices can be listed opposite the order they are removed so that reducing the level of detail is as simple as shortening the vertex list. Unfortunately, the face list modification is performed at some time cost. This pushes some developers away from CLOD entirely, but many game developers are using Half-Edge Collapses in upcoming titles.

CLOD algorithms for models don't work well for very large data sets like the mesh describing mountain terrain a character is walking on. Jonathan Blow gave a talk described problems he's had working with ROAM and other popular terrain rendering algorithms. He said the algorithms tend to fail in practice in ways they don't in the laboratory. Some of this is due to large data sets and the greater amount of use games receive compared to research projects. Blow claimed that research ignores cases that are important to the game developer and called for more realistic criterion for evaluating algorithms.

CLOD terrain by Bolt Action

One specific problem he experienced was positive feedback in the rendering/visibility process. Terrain rendering algorithms like ROAM depend on a property of realistic animations called frame coherence. Between two successive frames of animation, the scene changes very little. ROAM exploits this by incrementally increasing and decreasing detail levels of terrain, rather than starting from scratch every frame. The farther the viewpoint moves between frames, the more work must be done to produce the next image.

Blow's observation is that this is a positive feedback loop. If the frame rate begins to fall, the viewpoint will move successively farther in each frame because it travels for a longer period of time between renderings. Because the viewpoint is traveling farther, frame coherence diminishes and incremental terrain CLOD algorithms will take longer to complete. This delay drives the frame rate down, causing the viewpoint to move even farther... the process repeats until the frame rate is driven to zero. His development team frequently observed this process and was unable to stabilize the raw ROAM algorithm no matter how much optimization they performed.

To overcome this problem, Blow modified ROAM to reduce the number of geometry recomputations. His approach uses intersections with implicit isosurfaces to trigger the level of detail changes. For technical details, see Bolt Action's papers and presentations site. Look for a beta release of his new game using this technology free on the 'net soon. The early shots looks amazing; in the final version, characters will glide through a world with 2^{31} triangles worth of terrain enabled by the new algorithm.

Geometric Algebra is a vector algebra that is making waves in many scientific communities. It is based on Clifford Algebra, a system of mixed dimensional algebra that was mostly ignored for a century after its discovery in 1878. Mixed dimensional algebra addresses situations where geometries of different rank need to be compared but can't be because they have different dimensions. A simple example of this is a 2D region in a plane and a polygon in 3D are essentially 2D objects, but the polygon is technically 3D.

Renewed interest in quaternions and mixed dimensional algebra from the physics and computer graphics communities brought Clifford Algebra back into the scientific consciousness and led to the extensions that are called Geometric Algebra.

Geometric Algebra is the mathematics from which complex numbers (a + bi) and quaternions are derived. Members of the games and graphics community are becoming familiar with quaternions as four dimensional vector quantities for representing rotations and camera orientations. Many talks referenced Geometric Algebra as the underlying mathematics behind techniques for producing smooth camera motions, thick curves (a circle swept along a 3D curve), texture mapping curved surfaces, and texture mapping of sampled data.

Quaternion Tubing by Andrew J. Hanson

Alyn Rockwood, Chris Doran, Joan Lasenby, Leo Dorst, David Hestenes, Stephen Mann, and Ambjorn Naeve gave a detailed mathematical course on Geometric Algebra and its applications. Beyond the mathematics, they held forth an amazing possibility for Geometric Algebra: it may be the mathematical missing link uniting problems in disparate fields like general relativity, collision detection, particle physics, and object motion. The implication is that problems from many domains may collapse, yielding common solutions and allowing better interaction between scientists in many fields. One of their most startling conclusions was that Geometric Algebra gives solutions to various physical situations on the level of elemental particles where general relativity breaks down or yields inconsistent results. Physicists hope that within a year experimental results will be able to confirm the theoretical findings. If this is indeed the case, Geometric Algebra may bring a new era of scientific discovery in addition to forming an interdisciplinary mathematical language. In short, Geometric Algebra will walk your dog, make you coffee in the morning, and is a desert topping and a floor wax.

Leo Dorst maintains a website of Geometric Algebra information and links. David Hestenes' website contains information on the history of Clifford Algebra and Physics research.

The Wooden Mirror by Daniel Rozin is the hands-down coolest piece in the SIGGRAPH 2000 art gallery. It is a giant octagonal wall mirror constructed entirely of wood. To make it act like a mirror 830 40mm x 40mm wood squares are connected to tiny servo motors with direct overhead lighting. An invisible camera looks out through a hole in the center. A Macintosh computer translates the digital image seen by the camera into angular positions for the wood panels and drives the servos. As the panels tilt, the shading on them changes and an image is formed in shades of brown. The whole piece operates in real time, making it possible to stand in front and interact with the it.
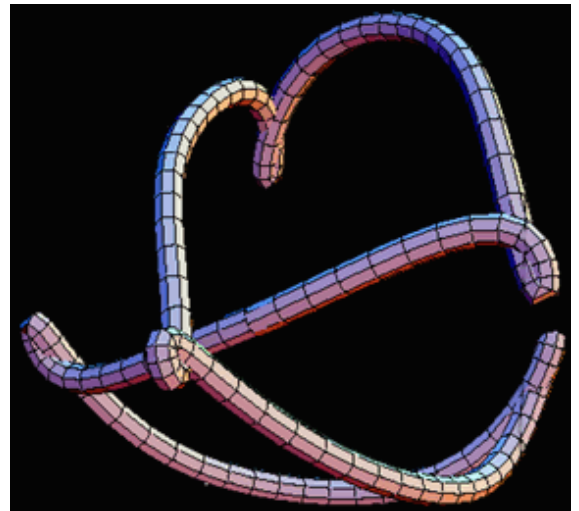
The servos and moving wood make a continual rushing sound not quite organic or mechanical. The extensive use of wood in such a digital and technological context, combined with the rushing sound form a piece of art that is at once clever, comforting, and intellectually challenging. From an engineering perspective, it is impressive that the mirror functioned accurately and robustly and was performing as flawlessly at the end of the week as at the beginning.

See QuickTime movies of the mirror in action on Rozin's site and look at some of his other pieces.

SIGRRAPH 2001 will be in Los Angeles, site of SIGGRAPH 99.
Submissions are currently being accepted with a January 2001 deadline and registration is open. What can we expect to see there?

Well, there are some serious divisions to be resolved in the graphics community. Yes, everyone was open and friendly, and the receptions and exhibits were excellent places to meet other graphics professionals and talk shop. However, the degree of respect attendees accorded one another and generally friendly atmosphere do not mean that there were not deep professional tensions present in the SIGGRAPH crowd.

There are clear professional tensions between communities of graphics developers and researches that work on opposite sides of any given interface.

Photo by Marianne K. Yeung

A Guide to SIGGRAPH Acronyms
Do you speak graphics slang?

ACM	Association for Computing Machinery
AABB	Axis Aligned Bounding Box
BRDF	Bidirectional Reflectance Distribution Function
CLOD	Continuous Level of Detail
CGI	Computer Generated Imagery
DOF	Degrees of Freedom
FMV	Full Motion Video
FOV	Field of View
HOM	Hierarchical Object Model
HRTF	Head Related Transfer Function
HZB	Hierarchical Z-Buffer
IBMR	Image Based Modeling and Rendering
LOD	Level of Detail
OBB	Oriented Bounding Box
PS2	Playstation II
SIGGRAPH	Special Interest Group on Graphics

We've socially bridged these divisions and are rubbing elbows at receptions, but new research and development efforts are needed to connect the various groups.

The divisions are most distinctly played out on two fields: **hardware vs. software** and **games vs. academia**. Hardware developers have achieved incredible fill rates in recent years, enabling PC's to replace SGI machines as the development workstation of choice and the primary platform for most SIGGRAPH attendees. But these incredible fill rates come with long graphics pipelines that make state changes extremely expensive, including texture and geometry changes. New graphics techniques rely on dynamically mutating models and textures. This means that the high fill rates aren't achieved in practice since programs tend to bottleneck on moving data between the CPU and the graphics processor and on stalls/cache dumps created when vertices, textures, and face lists are mutated. Software developers feel that hardware isn't giving them the right feature set to accelerate their techniques, while hardware developers feel that software developers aren't programming to take advantage of the features provided.

In the game developer vs. academics divide, the two groups have grown similar without establishing a working relationship and are now stepping on each other's toes. Game developers are extremely educated about new research and are even holding their own conferences and publishing their own literature. Researchers are working on interactive techniques, incorporating game engines, and attacking real time graphics, physics, and virtual reality issues. On the surface, the two groups seem to have converged. But as Chris Hecker pointed out in his talk, they have totally different resources and mandates.

Collaboration between the games and academic groups needs to improve. Academics and pure researchers should look to game companies to refine theoretical solutions and tackle problems that game developers are facing. Game developers should seek researchers out in order to generalize and publish results achieved during the development process.

I'd like to see more work on leveraging the strengths of our entire graphics community. Hardware vendors need to look farther than the fill rates and simple, easily parallelized effects to provide the computational building blocks for image based rendering, level of detail, and advanced occlusion techniques. This means finding ways to software developers mutate geometry and textures without stalling the graphics pipeline, providing more programmability on graphics hardware, and widening data paths. The flipside is that researchers need to consider hardware when designing algorithms so that hardware developers will be able to implement some algorithms in silicon and software developers will be able to implement other algorithms efficiently using existing hardware.

Maybe there is a Geometric Algebra solution to all graphics problems and we can all let mathematicians and physicists build the hardware, develop algorithms, and write the games... or maybe we'd better take the great advances presented in the past few years at SIGGRAPH and other conferences and figure out how to get them to work together.

So next year, in Los Angeles, I'll be on the lookout for hybrid solutions bridging the divides and gathering the low hanging fruit where disparate techniques come together. I'll also be looking for real time applications performing photon mapping, LOD techniques using IBMR, or more interaction between high level techniques and graphics hardware designs. And I'll be looking for some great game demos at SIGGRAPH 2001, as more game developers participate. And of course, I'll be looking for you.

Seeking a job in graphics research or industry? The following companies advertised at SIGGRAPH that they have open positions.

- Turbine
- Electronic Arts
- nVidia
- Blizzard
- Zoesis Interactive Entertainment
- Will Vinton Studios
- Trenza
- ReelCareers.com (recruiter)
- Metastream
- Edge of Reality (game developers wanted)
- Moon Crescent Animation
- Pulse
- Virtual Technologies, inc.
- Web Scope Inc
- Geometrix

---

return to flipCode