INTEGRATING GRAPHIC SOFTWARE - THE MAKING OF "M.A.T.E"

Sílvia Delgado Olabarriaga
Ana Elisa Schmidt
Carla Maria Dal Sasso Freitas
Manuel Menezes de Oliveira Neto
Marcos Eduardo Nascimento
Rui Bastos
Alexandre Casacurta
Anatólio Laschuk
João Batista de Oliveira
Marcos Kordyas Dossa
Robson Rodrigues Lemos
Soraia Raupp Musse

Universidade Federal do Rio Grande do Sul
Instituto de Informática
Caixa Postal 15064, CEP 91501-900, Porto Alegre, RS, Brazil
e-mail: informatica@inf.ufrgs.br

ABSTRACT: This work describes an experience developed by the Computer Science Graduate Program graphics group at the Federal University of Rio Grande do Sul. The goal was to make a short video showing computer animation completely generated with local software. The video "M.A.T.E" has been produced as a result of this effort. All steps in the production of the video are described: script and storyboard design, objects modeling, motion planning, rendering, sound and final production. Some comments about this experience are presented in the conclusions.

KEYWORDS: modeled animation, computer graphics.

HISTORICAL REASONS

For a long time computer animation has been a very popular subject among computer graphics students and professors of our department. Some simple modeled animation systems have been developed, but people could never see the results on film or video because our laboratories didn't have the appropriate equipment to record computer generated images into these media.

In December'90 the group contacted the computer graphics department of a local TV company (RBS-TV) which has a frame-buffer connected to a U-Matic video recorder. They agreed to lend us the equipment for an experimental computer animation video and then M.A.T.E. project has started.

SCRIPT DESIGN

The script has been created by the whole group (about 12 people) and this was not an easy task. There were many constraints, some internal, other external, that influenced the choices for a story: we could only use locally developed software (from modeling to animation and rendering); the video should show the "state-of-the-art" of the research work under development; we should do something that local animation houses couldn't (because of the technique itself or cost); the video should be very short.

The discussions lasted more than one month, but finally converged into this story about a chess game between gray and red pieces. Initially two moves are shown (red tower and gray king), while the observer flies around the scene. Then an yellow sphere falls over the chessboard, stops in front of the gray king and turns into a red queen, which creates a checkmate situation on him. A trapdoor under the king suddenly opens and he falls from the chessboard. The story ends when the king passes between two letters "F" and "M" forming the word "FIM" ("end" in Portuguese). The whole sequence lasts 13 seconds. Some frames can be seen on fig. 1.

OPENING

The video opening shows a spiral galaxy of 35000 light reflective particles contracting into a sphere, forming the Computer Science Department logo. The animation was generated through SISPA [1], a particle system specially developed for this project.

OBJECT MODELING

The objects were modeled by two different programs, BEZIER4D [7] and SIHMOS [2], that allow the generation of 3D polygon meshes. The chessboard and trapdoor models were plane Bézier patches (BEZIER4D); the sphere (rotational sweeping) and letters (translational sweeping) were created by SIHMOS; the chess pieces were Bézier surfaces created through rotational sweeping (BEZIER4D).

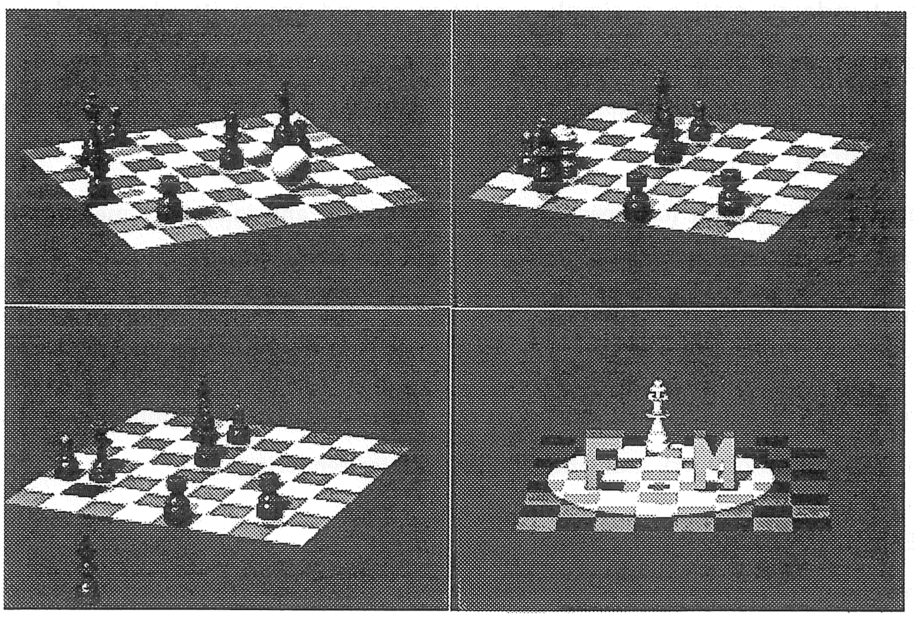

Fig. 1 - M.A.T.E. keyframes

ANIMATION

The camera (observer) and some objects called "actors" (chess pieces and trapdoor) move; the sphere both moves and changes its shape and color.

The camera flies around the chessboard in a spiral movement, slowly going up and always looking at the chessboard center. Key-frame positions for the camera were calculated using a curve-fit program [5] and interpolated by PREVIEW [9].

The pieces move in a straight line from the initial to the final square on the chessboard. The positions were calculated through linear interpolation of key-frames, by PREVIEW.

Sphere's motion simulates a projectile using physically based laws. The intermediate positions were calculated by PROJÉTIL [4]. This actor also changes its color and shape from an yellow ball into a red queen; the intermediate shapes and colors were calculated by AIF [8].

The trapdoor's motion is simply a rotation around one of its edges and was calculated by PREVIEW by means of key-frame interpolation.

The gray king's final motion is a physically based free fall that was also generated by PROJÉTIL.

Each animation part has been calculated separately and joined by means of a script, which described actor's actions and the timing (frame numbers). This action could be informed only for some key-frames (and then interpolated by PREVIEW) or directly specified for every frame (in files). PREVIEW was the program in charge of synchronizing every actor's and camera motion, generating the description of every animation frame for later rendering.

PENCIL TEST AND RENDERING

The complete animation was first tested on the PREVIEW system [9]. Wireframe images of the scene were quickly seen, and the script was changed to accomplish animator's ideas. When the motion was considered ready, the description of each frame in the sequence has been output to files that were used by the renderer. Each set of files described one frame.

The final images included shadows and Phong shading, and were generated by McSHADOW [6].

SOFTWARE TOOLS

Some software tools already existed and had to be adapted, others were under development as prototypes of ideas for graduate research, and a few were specially developed to accomplish this project. Fig. 2 shows all programs involved, which are shortly described below:

SISPA: particle system simulator [1].

SIHMOS: solid modeler for sweep and parametric representations [2]. The output is a polygon mesh.

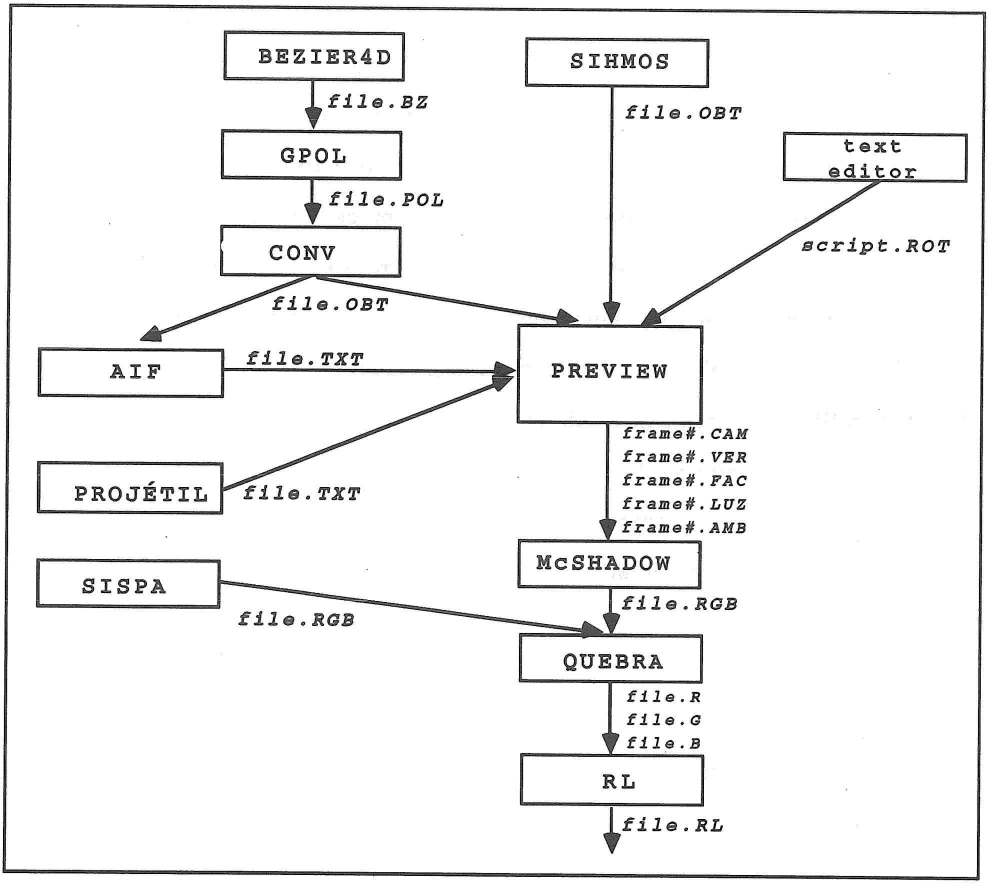

Fig.2 - Software Tools and Files

BEZIER4D: interactive Bézier surface editor and animator [7].

GPOL: converts descriptions of Bézier surfaces into polygons.

CONV: converts polygon files into the local file format.

PROJÉTIL: calculates motion based on inverse kinematics, considering gravity, air resistance and collision with the floor [4]. The motion positions are stored in text files.

AIF: automatically interpolates two given Bézier surfaces [8]. The sequence of shapes are generated into numbered files and it shows a gradual and continuous metamorphosis.

PREVIEW: 3D modeled animation system based on key-frame linear and Bézier interpolation with velocity control through direct kinematic. It reads a script and calculates scene parameters at every frame (camera position and interest, lens, actors transformations, color and shape). This system also displays wireframe images that allow the animator to foresee the result of combined motion. Fig.3 shows a script example showing keyframe interpolation (camera), physically based motion (sphere) and metamorphosis (queen).

McSHADOW: photo-realistic renderer [6]. Simulates colored light sources, Phong shading and shadows. The output is a true-color image file.

QUEBRA: separates red, green and blue image information into three files.

RL: converts local image file into CUBICOMP format (for recording on video).

INICIO NRO_QUADROS= 10	number of frames
CAMERA QI= 1 PI= 0 0 0 QF= 10 PF= 5 5 5 FIMCAMERA	camera motion from (0,0,0) to (5,5,5)
ATOR sphere.obt 1 QI= 1 QF= 10 LISTA_POS= positions.txt FIMATOR	sphere's positions have been calculated by PROJÉTIL
ATOR queen.obt 2 QI= 1 QF= 10 LISTA_FORMA= shapes.txt FIMATOR	queen's shapes have been calculated by AIF
FIM	

Fig. 3 - Example of script

The communication among the programs was done through files, usually text-files. Sometimes it was necessary to develop special "filters" to adapt existing file formats to locally defined ones. The file formats involved in this project are listed below:

.BZ, .POL:	control and polygon mesh of Bézier surfaces.
.OBT:	local "standard" for 3D object description and storage.
.ROT:	text file containing the animation script (see fig.3).
.CAM:	camera specification at a certain frame.
.FAC, .VER:	polygon mesh and vertex information for all objects at a certain frame.
.LUZ, .AMB:	source light and special data for the renderer.
.RGB, .R, .G, .B:	final image (true-color).
.RL:	image in CUBICOMP format, ready for recording on video.

HARDWARE

Three different kinds of computers were used: IBM-PC compatible machines, for object modeling (SIHMOS and BEZIER4D), physically based motion calculation (PROJÉTIL), video opening image generation (SISPA) and image file preparation for recording (QUEBRA and RL); PROCEDA workstation (MC68000 based Brazilian microcomputer), for pencil test (PREVIEW); Sun workstations, for rendering (McSHADOW).

IBM-PC computers could exchange files both with Sun (through NFS) and with Proceda workstations (by serial communication). Image recording was done on a CUBICOMP frame-buffer attached to an IBM-286 compatible. Images were taken from our lab to the frame-buffer in floppy disks.

SOUNDTRACK

The main soundtrack is part of the Carl Orff's "Carmina Burana". The sound effects were specially developed for the required timing on a CASIO CZ-3000 synthesizer. This equipment have been attached to an IBM-PC compatible computer via parallel interface and controlled by program. Both software and hardware were locally developed [3].

VIDEO PRODUCTION

The images (opening + chess game) were recorded on U-Matic video at the TV station. Sound and titling were appended using their regular equipment, with a great help from people from the TV.

RESULTS

Some comments should be done about the video itself: it is too short, and people can hardly see what's happening at the story (the time limit imposed by ourselves was too small); the colors used could have been more carefully chosen (we had no true-color monitor at that time to foresee the results); the camera moves at the same time as the actors, so one sometimes doesn't realize all motion (we couldn't see the animation in real-time before recording).

Comments on software integration must include an analysis about PREVIEW. This tool has been used not only in this project, but also as a quick debugger for all animation work done by the group. Nowadays a new version is under development in Sun workstations, with window interface and a revised script language. Also, the output file format has been changed to accomplish new renderers.

CONCLUSIONS

In spite of these comments, the experience has proven to be very important to our group: people who worked with animation could finally see the results of their work; the final result (the video itself) is something we can use to show other people what we are able to do at this moment; many graphics programs were used together and this was unusual in our group in the past; about 12 people worked together and had to learn how to interact with each other, not only from the software point of view.

Before M.A.T.E has been developed, everyone used to build his own modeler, render and so on. This means that people almost started from scratch and couldn't get very far. Nowadays we have internal file formats, libraries and generic tools which improve the reusability of locally developed software.

THANKS

To Rubem Pechansky, Bayard da Silva Christ, Fábio Rebello, Yara Lemr, Márcio Pinho, Flávio Cauduro and RBS-TV.

BIBLIOGRAPHY

[1] R. Bastos & J. B. Oliveira. *Animação de um sistema de partículas refletoras de luz*. 1991. (Personal Notes).

[2] A. Casacurta. *SIHMOS: Sistema Híbrido de Modelagem de Sólidos*. Porto Alegre, CPGCC-UFRGS, 1991. (MSc).

[3] M. Dossa. *Controle do CZ-3000 por microcomputador*. 1991. Personal notes.

[4] R. Lemos. *Controle de movimentos em animação: teoria e aplicação experimental*. Porto Alegre, CPGCC-UFRGS, 1992. (MSc).

[5] Y. Lemr. *Análise de deformações em superfícies Bézier bicúbicas*. Porto Alegre, CPGCC-UFRGS, 1991 (TI 223).

[6] M. Nascimento. *Um estudo de algoritmos para geração de sombras projetadas na síntese de imagens foto-realísticas*. Porto Alegre, CPGCC-UFRGS, 1992. (MSc).

[7] M. Oliveira Neto & R. Bastos. *BEZIER4D - ambiente para modelagem, visualização e animação tridimensional*. In: SIBGRAPI'91, São Paulo, 1991.

[8] M. Oliveira Neto. *Algoritmo Interpolador de Formas entre objetos modelados por superfícies spline*. Porto Alegre, CPGCC-UFRGS, 1992. (MSc)

[9] A. Schmidt & S. Musse. *PREVIEW - sistema de animação*. Porto Alegre, CPGCC-UFRGS, 1991. (Research Project 184).