



Game Programmming Lab

Interim Report

Pharaoh's Tomb

**Melanie Imhof
Stefan Geiger
Thomas Brunner**



Contents

1	Development State	3
1.1	Content Pipeline	3
1.2	Character	3
1.3	Environment interaction	3
1.4	Animation	3
1.5	Map Creation	4
1.6	Puzzles	4
2	Development Schedule	5
2.1	Tasks	5
2.1.1	Functional Minimum	5
2.1.2	Low Target	5
2.1.3	Desirable Target	6
2.1.4	High Target	6
2.1.5	Extras	6
2.2	Schedule	7

1 Development State

1.1 Content Pipeline

We have implemented an I3D loader which loads all basic attribute from an I3D file. These files can be created with Maya, 3DS Max or any other 3D content creation tool which has a I3D exporter available. The I3D file stores a scenegraph structure with all geometry and material information as well as some user defined attributes.

One of them is the concept of onCreate attribute, which define a function name, which is called as soon as the specified object is created. This allows the model creator this specify special objects which can later be referenced inside the game code.

E.g one can add an onCreate attribute to a light switch, to automatically attach some logic to the switch as soon as the player uses the switch.

Additionally several predefined attributes can be set to each object inside the Maya, such as physics attributes and rendering properties. These values are stored inside the I3D file and are read by the game I3D loader code and are used automatically to create physics objects. Trigger callback functions can be added to an object inside the game, which are called as soon as the player enters the region defined by the geometry of the object.

Material properties are read directly from the I3D file, such as diffuse, normal and specular textures or specular color and cosine power. The appropriate shaders are generated dynamically based on the given attribute of the material and the environment properties such as fog, lights and shadow maps. Loading of lights and cameras specified inside Maya is supported as well.

1.2 Character

A character controller is implemented which allows moving, jumping and automatic stepping on low platforms. It is based on the code provided by the BEPU Physics Samples. Additionally forces are added to the objects the character is walking on, to make some physics puzzles possible, such as the shelf puzzle. A third person camera controller was added, which collides with the environment and smooths the transitions.

We have adjusted the character mesh from a previous game project we have started. Additionally we have created a new texture and improved the animations.

1.3 Environment interaction

To solve the physics puzzles, the player has the possibility to pick up the objects around. Once an object is picked up, it can be moved relative the to player with the Directional Pad. Currently there is no collision of the picked objects with the environment.

1.4 Animation

We have implemented a keyframe animation support, which moves any scenegraph object around defined by a series of keyframe. The attributes supported are translation, rotation, scale and visibility. Additionally these keyframes are arranged in several clips, which can be played back at different times with different

speeds. Up to four clips can be blended together, for example to create smooth transitions between different clips. E.g blending from a walk to a jump cycle.

1.5 Map Creation

We have created a map to test the different puzzles and game mechanics. It currently is a very rough prototype with no advanced geometries or textures. The whole map is created inside maya and is exported to a single I3D file.

1.6 Puzzles

We have implemented several puzzles. These are memory, spikes, stack them up and shelves. The physics simulation turned out to be quite stable, which enables us to create the physics puzzles as planned. We also have implemented a fail restore mechanism, which allows the player to reset the character or even all objects of the current puzzle, if something goes wrong. Puzzles can also define fail points, which automatically trigger a reset of the character position.

2 Development Schedule

2.1 Tasks

2.1.1 Functional Minimum

#	Task	Length	Member	Description	Done
1	Initial setup	2 D	Stefan	Startup the game and initialize and track the current state	x
2	Scenegraph implementation	4 D	Thomas	Create the implementation of a basic scenegraph with support for transformgroups, shapes, light and cameras	x
3	I3D loader	4 D	Thomas	Create the implementation to load a scenegraph from an i3d file	x
4	Physics integration	4 D	Stefan	Add a physics engine and run it in a separate thread. Commands should be buffered and executed once per frame	x
5	Render system	4 D	Stefan	Travers the scenegraph and render the visible object. Setup the material and shader parameters	x
6	Character controller	4 D	Melanie	Implement a character controller to move and jump	x
7	Camera controller	2 D	Melanie	Implement a simple camera controller	x
8	Environment interaction	2 D	Melanie	Implement a simple model to pick up objects in the world	x
9	Character modelling	4 D	Stefan	Model a simple mesh of the player character	x
10	Test world modelling	4 D	Stefan	Model a simple test world with two puzzles	x
11	Shelf and Memory puzzle	2 D	Melanie	Implement the shelf and memory puzzle code	x

2.1.2 Low Target

#	Task	Length	Member	Description	Done
1	Keyframe Animation	4 D	Thomas	Update the scenegraph transform groups involved in a keyframe animation	x
2	Material/Shader system	4 D	Stefan	Dynamically create the shaders needed for a given shape and environment (lights, fog..)	x
3	Character animation	8 D	Stefan	Create a simple animation of the character model	x
4	Puzzles code	8 D	Melanie	Implements more puzzles	x
5	Puzzles modeling	8 D	Melanie	Create models of more puzzles	x
6	Map modeling	8 D	all	Create models for the environment	x
7	Simple in-game GUI	4 D	Thomas	Create a simple in-game gui to show help and status	x
8	Fail restore points	2 D	Melanie	Add points that the player is placed if a puzzle was failed	x

2.1.3 Desirable Target

#	Task	Length	Member	Description	Done
1	Camera controller improvements	4 D	Melanie	Improve the camera controller to collide with the environment and do some smoothing	x
2	Character controller improvements	4 D	Melanie	Improve the character controller to support rope swining and more advanced environment interactions	
3	Shadow mapping	4 D	Stefan	Add shadow mapping support to rendering	x
4	Game menu GUI	2 D	Thomas	Add a nice game menu GUI	
5	Character sounds	2 D	-	Add sounds directly triggered by the player (footsteps, collisions...)	
6	Background sound	4 D	-	Add sound that runs in the background to improve the atmosphere	
8	Advanced rendering effects	4 D	Stefan	Reflection/Refraction, Postprocessing effects (e.g. intoxication)	

2.1.4 High Target

#	Task	Length	Member	Description	Done
1	Random special events	4 D	-	Add events that happen randomly (e.g. falling stones)	
2	Split-screen cooperative play	8 D	-	Add cooperative play and add special two-player puzzles.	
3	Simple savegames	2 D	-	Store the current progress between the levels	
4	Idle animations	4 D	-	Play random animations if the player is idling	
4	Special sounds	4 D	-	Add sounds that are played if a player does something wrong or right	

2.1.5 Extras

#	Task	Length	Member	Description	Done
1	Control setting GUI	4 D	-	Add a control setting GUI	
2	Random characters sounds	2 D	-	Add sounds that are played randomly by the player (cough...)	
3	Network cooperative play	10 D	-	Allow mulitple players over the network to play in cooperative mode	
4	Collectables	4 D	-	Add special items that can be collected while playing the game to receive a special award.	

2.2 Schedule

Date	Melanie	Stefan	Thomas
08.03.		Initial setup	
15.03.	Proposal Final		
21.03.	Character controller	Physics integration	Scenegraph implementation
22.03.	Physical Prototype, Final In-class Pitch		
26.03.	Camera controller	Render system	I3D loader
01.04.	Environment interaction	Character modelling	Keyframe Animation
08.04.	Puzzles code/Puzzles modeling	Material/Shader system/Map modeling	Simple in-game GUI
19.04.		Character Animation	Fail restore points
19.04.	Interim Report, Presentation/Game Demo		
23.04.	Camera controller improvements / Character controller improvements	Shadow Mapping	Game menu GUI
05.05.	More puzzles	Advanced Rendering effects	Character Sounds / Background Sounds
10.05.	Bug fixing		
10.05.	Alpha Release		
17.05.	Bug fixing, Layer 4		
17.05.	Playtesting		
25.05.	Bug fixing, Layer 4, polishing		
25.05.	Final Presentation Debriefing		
29.05.	Conclusion, Final Digital Video		
01.06.	Final Presentation		