Project Proposal - Game Programming Lab '07

Group members: Benjamin Schindler, Basil Fierz, Henning Avenhaus

Proposal „Gravity bound“

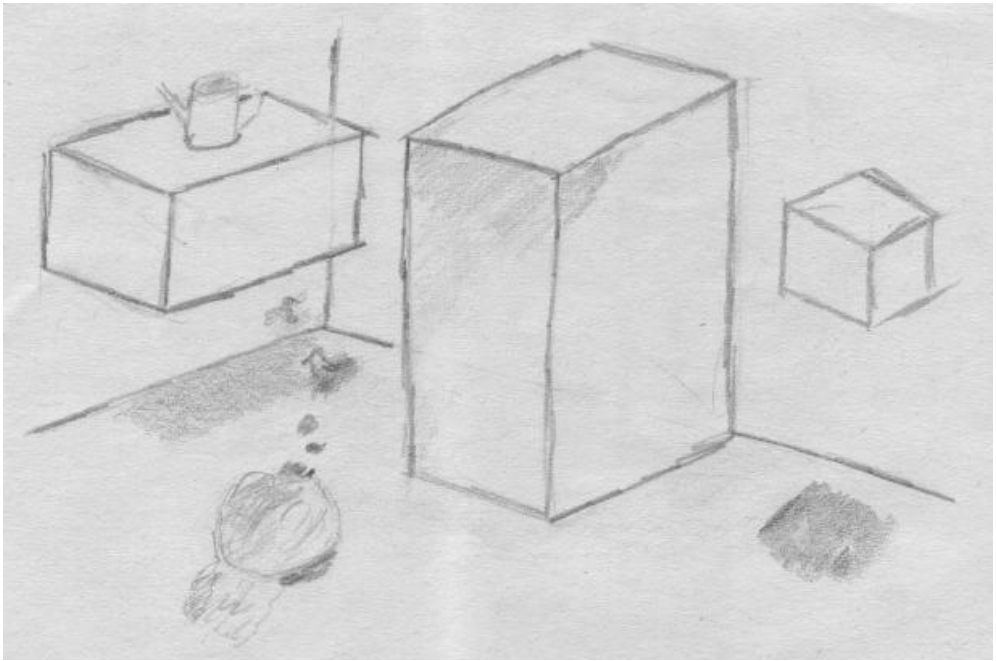
Game description

The following game description contains the whole game up to the last detail. This means that all parts are described; the ones we plan to implement for the project, and the ones that go beyond the scope of this laboratory. What exactly will be implemented is described in the development schedule and the development targets.

Summary

The game is an arcade style game, which orients itself loosely on the old and famous arcade game „Bubble Bobble“. “Gravity bound” is about a heroic drop of oil, named “Blob”. He lives in a three-dimensional cubic world together with some monsters. He has several possibilities to move. He can “slime” forward, jump, crawl up walls and obstacles and he can slide over the ground.

But Blob doesn’t live on his own. He has to shares his home with monsters, which try to kill him. To find his peace again Blob has to defend his world by chasing and killing the monsters. Fortunately he is not unarmed. Blob can shoot oil bullets to paralyze his opponents. The ammunition, which he shoots, comes from himself. The more Blob shoots, the more he shrinks. If he shrinks below a certain volume he can’t shoot anymore and thus is unprotected further on.



Monsters Blob has paralyzed, he can eat and devour. If the monsters by any chance are not paralyzed and Blob touches them he loses dies.

Game play details

To make the game more appealing for non Blobs, it is designed as a 3D arcade style game. The goal of “Gravity bound” is to score points. The more you score, the better you are. This can be achieved with two means; killing enemies and collection bonus items.

Blob has more than one live and consists of a variable amount of oil, depending on the amount of shots he has emitted and the amount of oil he has collected through bonus items, monsters.

“Gravity bound” as the name says is about gravity. The player can tilt the box world by 90° degrees in each direction. Turning the cube not only changes the view, it also changes the gravity. Blob and the monsters will be affected by the changing gravity, but not the bonus items.

Oil

As stated earlier the amount of oil of which Blob consists is variable. It reduces when he shoots and it increases when he eats monsters or collects bonus item. Blob can also recollect the ammunition he emitted by going over them, or by using an attraction force field. This force field acts on the oil particles and bring them back to Blob.

Shooting

Blob shoots always straight forward, depending on the view direction. This way it is possible that he can also shoot upwards. The shots are affected by the gravity.

Bonus items

The game contains a few items, which Blob can collect:

- Oil cane: Add new oil to Blob.
- Special oil: A shot uses less oil, but has the same efficiency at least for a certain amount of time.
- Shooting enhancer: Blob can shoot faster.
- Shooting stabilizer: Blob can shoot farther away.
- Live: Blob gains one additional live.
- Shield: Blob is not vulnerable for some time.
- Jump enhancer: Blob can jumper higher.

Enemies

In his box world Blob has to face two types of enemies. Walking ones and flying ones. Blob loses one live if he touches an opponent, if they aren't paralyzed. The monsters can move around in the level and have a simple sense of where Blob is, so they won't walk around completely randomly.

If flying monsters make any sense in the game has to be evaluated during game play testing.

Level elements

There are some elements:

- Trampolines: Blob and monsters make huge leaps up words (normal of the ground).
- Beamers: Transports Blob at a random point of the level.

Score system

Score points is possible in several ways. The enemies Blob eats add points to the score. He can collect items, which also add points. At the end of each level the oil volume above the initial mark are also added to the score. Because the score system is the most crucial part of the game (scores are used by players to compare each other), it is hard to balance. Thus the final amount of points the player gets for an action will be determined during play testing.

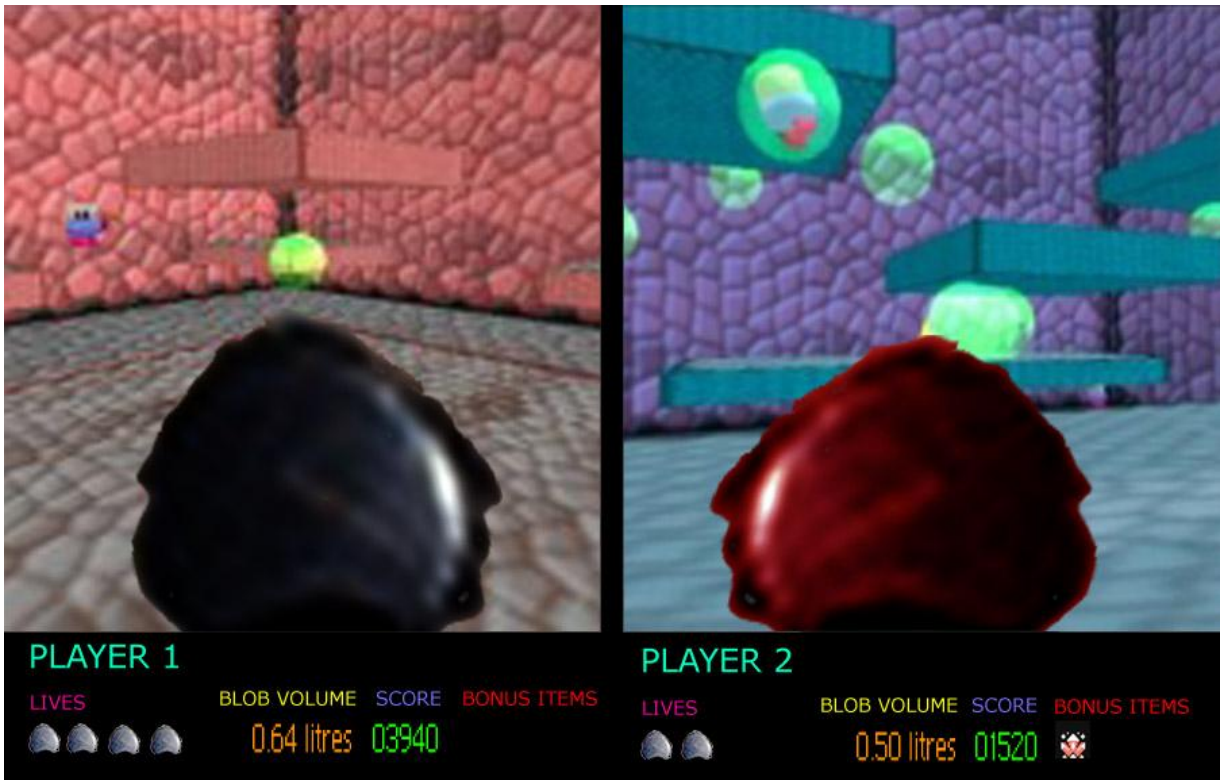
Controls

The game can only be played using an XBox360 controller (although it will also run on the PC). The controls are configurable using an on screen configuration panel.

The controls are similar to the ones of many 3D person view games. One steering cross is used to maneuver the main character (forward/backward/sideward) and one to change the view direction/camera.

Multiplayer

As most arcade games “Gravity bound” can also be played by two or four players. The multi player mode will use split screen technology. All players play on the same screen.



The players can collect not only the own shots and eat the monsters they paralyzed themselves, but also the ones from the other players.

The gravity feature of the game undergoes some special treating in this mode. Since all players hunt in the same world, it probably will be very annoying if any player could change the gravity at any time. Thus there are several alternatives.

- The world tilts itself randomly after a fixed or random time span.
- The players tilt the world, but have to wait for a certain time span between two tilts.
- We introduce a new level element that tilts the world when it is touched.

Which options will be the best for game play, fun and balance has to be shown in game testing.

Technical details

Graphical user interface

The game contains a simple graphical user interface with buttons, labels and a onscreen keyboard. Navigation will be done using the Xbox360 controller. The two player version of the GUI has to support a split screen mode.

3D graphics

The oil drop (Blob) is illuminated using environmental mapping and will be shinny and reflective. Other than that the game will be kept simple in remembrance of old fashion arcade games.

Sound

There are sounds for the blob, the level and bonus items and the monsters. The sound effects will be three dimensional. If something happens behind the player he should hear it from behind.

Background music will be a part of the full game play experience.

Physics

The crucial part of the game technology are physically based simulations. On important part of that is collision detection. Although this topic is rather complex and much work, we can reduce the amount of features needed to a smaller subset. All items except Blob can be sufficiently well be modeled as boxed.

Blob on the other hand is modeled as mass spring or FEM system to make it wobbly and jelly like.

One additional feature is a simple type of force fields. This is also sufficiently simple, because it affects only a small subset of elements (oil ammunition).

Content

Monsters and bonus items are simple 3D models. Blob uses simple triangle mesh, which is animated by the simulation described above.

For the level no special models are needed for they consist of a bunch of box alike textured elements.

Monster AI

The monsters have to be able to orient themselves in the level and walk through the level. They can jump and actively move in direction of Blob.

Camera

A third-person camera will be used. Special care has to be taken if Blob is near walls. The camera‘s up direction will point in the opposite direction of the gravity.

Development Stages

Functional Minimum

Graphics

No textures, simple point light in the middle of the world, Sphere for Blob, simple geometric figures (sphere, cube, ...) for the monsters.

Sound

No Sound

Physics

Working collision detection. Spherical model for Blob. No mass-spring or FEM techniques. Working gravity in one direction. Shots not affected by gravity. Shooting direction not controllable.

Game content

Blob and monsters present. No bonus items.

Monster AI

Stupid monsters. Monsters walk around and change direction if they meet a wall.

Camera

The camera has to work even in the functional minimum. It does not have to work perfectly close to walls.

Game logic

Shooting and hitting monsters possible.

Low target

Graphics

Basic textures, lighting with light sources (no shadows), triangle model for Blob (which is a mass-spring system), simple monster models (non-animated, hopefully from libraries). Working menu and score / status display

Sound

Basic sound effects for Blob and events like hitting a monster.

Physics

Collision detection. Mass-spring or FEM model for the blob. Gravity which can change direction. Shots affected by gravity. Shooting direction (up/down) can be controlled.

Game content

Blob, monsters, and bonus items present.

Monster AI

Stupid monsters. Monsters walk around and change direction if they meet a wall or at random. Also, flying monsters which change direction if they hit a wall.

Camera

Working camera, even close to walls. Camera angle dependent on shot angle.

Game logic

Implemented score system. Full game play with one level.

Desirable Target

Graphics

Textured levels, advanced lighting for Blob (shiny with environment mapping), simple animations for monsters only if available from libraries. Textured monster models (hopefully from libraries). Graphical effects when killing monsters etc. Animated tilting of the level.

Sound

Sound effects for the blob and for events like hitting a monster. Some monsters produce sounds. Bonus items produce sounds when hitting the ground. Background music (which can be shut off).

Physics

Sticking to walls and sliding.

Game content

Bonus items that provide special skills (faster / further shooting).

Monster AI

Better monster AI which can for example stay on platforms, hunt the player etc.

Camera

Working camera when sticking to walls.

Game logic

Fully functional split-screen multiplayer. Simple attractors for oil traces.

High Target

Graphics

More graphical effects when killing monsters. Hard shadows. Animated menu and score / status bar. Animated monsters.

Sound

Background to effect volume balance.

Game content

More specific bonus skills from items (invulnerability / higher jumps etc.)

Monster AI

“Intelligent” monsters.

Game logic

More levels.

Our Extras

Graphics

Graphical effects which produce colored light to light up the environment. Particle effects for the graphical effects. Multi-colored rendering of blobs (see game content).

Sound

3D Sound rendering.

Physics

Fluids in the levels. SPH simulation for the traces of oil in the level (better with the attractors). Physics for jump pads etc. Shots that bounce off walls.

Game content

AI controlled other blobs. Non-mixing colors if traces from other blobs (in multiplayer) are collected (gives multicolor blobs). Jump pads etc. Ability to fall out of holes in the bottom of the level and re-appear from the top.

Monster AI

Group behavior of monsters.

Game logic

Fancy and more advanced attractors for oil traces around the level.

Development Schedule

20.3.2007 – 3.4.2007

Task	Person	Time
Project proposal	All	15 h

3.4.2007 – 1.4.2007

Task	Person	Time
Work into XNA framework	All	4 h
Project setup	All	2 h
Reviews of other games proposals	All	5 h
Basic game scene setup (ready to add items)	Basil	3 h
Study Xbox360 control system	Benjamin	3 h
Monster movement (VERY basic)	Henning	3 h

10.4.2007 – 17.4.2007

Task	Person	Time
Basic graphic framework (box, sphere, boxes for monsters, status texts, one point light)	Basil	6 h
HUD with labels and containers	Basil	4 h
Collision detection (only outer box)	Benjamin	7 h
Blob (ball) physics	Henning	7 h
Controls	Benjamin	5 h
Monster movement (VERY basic)	Henning	2 h

Milestone: Very simple yet playable version

17.4.2007 – 24.4.2007

Task	Person	Time
Internal boxes within the big "world"-box	Basil	2 h
Simple monster graphics	Basil	5 h
Level format definition	All	3 h
Shooting for the blob	Benjamin	5 h
Collision detection with internal boxes	Benjamin	5 h
Camera (beginning)	Basil	4 h
Simple monster AI	Henning	5 h
Blob physics advanced (collision with internal boxes, ….)	Henning	7 h

Milestone: Functional minimum reached

24.4.2007 – 1.5.2007

Task	Person	Time
Camera (continued)	Basil	6 h
Level loader	Basil	5 h
Texture collection	Benjamin	5 h
Sound (basic)	Benjamin	10 h
Mass-spring or FEM model for blob (beginning)	Henning	10 h
Simple monster AI	Henning	5 h

1.5.2007 – 8.5.2007

Task	Person	Time
Light sources	Basil	3 h
Menu / status bar	Basil	8 h
Shooting direction for blob	Henning	4 h
Mass-spring or FEM model for blob (advanced, interaction with walls, ….)	Henning	10 h
Gravity change	Benjamin	3 h
Shots affected by gravity	Benjamin	2 h
Sound	Benjamin	6 h

8.5.2007 – 15.5.2007

Task	Person	Time
Complete low target issues	All	15 h

Milestone: Low target reached

15.5.2007 – 22.5.2007

Task	Person	Time
Multiplayer split screen	Basil	12 h
More Sound effects	Benjamin	4 h
Background music	Benjamin	8 h
Oil attractors	Henning	6 h
Better monster AI	Henning	6 h

22.5.2007 – 29.5.2007

Task	Person	Time
Environmental mapping for Blob	Basil	5 h
Multiple light sources	Basil	4 h
Background music	Benjamin	10 h
Better monster AI	Henning	6 h
More bonus items	Henning	4 h

29.5.2007 – 5.6.2007

Task	Person	Time
More level textures	Basil	5 h
Climbing, sliding	Henning	10 h
Camera for climbing	Basil	5 h
Better textures and models	Benjamin	10 h

Milestone: Alpha release; ready for play testing

5.6.2007 – 12.6.2007

Task	Person	Time
Collect feedback from other people, collect issues, and priories them	All	15 h

12.6.2007 – 19.6.2007

Task	Person	Time
Fix final issues	All	10 h
Prepare final presentation	All	5 h

Milestone: Final game release

Assessment

In contrast what game publishers think, games don't sell because of graphics. They sell because their fun to play. Fun has many faces. Gravity Bound is an arcade game and touches the player by their basic instincts; Hunting and Collecting. There are also some graphical goodies. Blob shiny and wobbly nature is an attraction for the eyes. Every game should have some feature that makes it unique and recognizable under other games.

Gravity Bound is about gravity. The variable direction of the gravity makes the game more diverse than other games. New players may just ignore it, but more advanced players will be able to use the gravity changing feature to maximize the gaming experience. They can turn the world to faster get to monsters and items, or just to have fun watching all the items fall through the level.

The game has also a challenging feature. Blob has only a limited amount of ammunition. So the player has to shoot precisely or otherwise he will be helpless. Alternatively the player can also recollect what he shot.

An further fun point is the multi player option. Everyone wants to compete against other players. Just playing over the network would be boring, for you don't have the pleasure of seeing your opponents. Thus we offer the option to compete against other human players in direct split screen competition. Since all players play in the same world the "gravity feature" makes even more fun than in single player mode. While in single player mode you can use the changeable gravity to your advantage. In multi player mode you can also use it to give the other players a disadvantage and disturb their movements.

Development Team

Benjamin Schindler

Semester: 8
Course of studies: Informatics, Visual Computing Track A
Courses taken: Introduction to Computer Graphics, Physically based Simulations in Computer Graphics, Visual Computing (core subject)

Basil Fierz

Semester: 7
Course of studies: Informatics, Visual Computing Track A
Courses taken: Graphische Datenverarbeitung I, Visual Computing (core subject)

Henning Avenhaus

Semester: 8
Course of studies: Computer Science and Engineering
Courses taken: Graphische Datenverarbeitung I, Physically based Simulations in Computer Graphics