TERRACOM

By Cory Thomas

Summary of Contents

Summary of Contents ____ 2
Table of Contents ____ 3
Overview ____ 9
Characters ____ 11
Storyline ____ 13
Levels ____ 24
Bosses ____ 32
Enemies ____ 38
Special Enemies ____ 47
Interfaces ____ 48
Sprites ____ 64
Music ____ 71
Production Schedule ____ 73
Terminology ____ 75

Table of Contents

Summary of Contents 2
Table of Contents 3
Overview 9
About the game 9
Premise 10
Characters 11
About this section 11
Lieutenant Daemon 11
Sarah McNelson 11
Commander Nelson 12
Terracom AI 12
Storyline 13
About this section 13
Storytelling method 13
Scene 1 14
Scene 2 15
Scene 3 16
Scene 4 17
Scene 5 17
Scene 6 19
Scene 7 21
Scene 8 23

Table of Contents

Levels 24

About this section 24

Level 1A 24

Level 1B 25

Level 1C 25

Level 2B 26

Level 2C 27

Level 3A 28

Level 4A 29

Level 5A 30

Level 6A 31

Bosses 32

About this section 32

Battle Chopper 32

Security Door 33

Laser Hunter 34

Shadow Hunter 35

Mine Layer 36

Shield Bit 37

Table of Contents

Enemies 38

Jet A 38

Jet B 39

Boulder 40

Terracom Fighter A 41

Terracom Fighter B 42

Gun Turret 43

Security Camera 44

Laser Beam 44

Closing Security Door 44

Mine 45

Circuit Cube 46

Destructible Circuit Cube 46

Special Enemies 47

Standard Projectiles 47

Simple Obstacles 47

Table of Contents

Interfaces ______________________ 48

About this section 48

Loading Screen 49

❖ Frame Actions 51

Game Over Screen 52

❖ Frame Actions 53

Title Screen 54

❖ Frame Actions 55

Story Scene 56

❖ Frame Actions 57

Level 58

❖ Level Objects 59

❖ Triggers 60

❖ Background 60

❖ SCROLL_SPEED 61

❖ display() 61

❖ Heads-up Display 62

❖ sprites 62

❖ Frame Actions 63

Table of Contents

Sprites 64

About this section 64

Sprite structure 65

triggers 66

object 66

checkCollision() 67

PARENT_SPRITE 68

xPos 68

yPos 68

zPos 68

zVelocity 69

hp 69

MAX_HP 69

enterFrame() 70

Music 71

About this section 71

Title / Credits 72

Story Scene 72

Level 72

Boss Battle 72

Production Schedule 73

About this section 73

Weekly schedule 73

Daily schedule 74

Table of Contents

Terminology ____________________ 75

About this section 75

AI 75

Interface 75

Sprite 75

Sprite 75

Terracom 75

Hunter 76

Hacker 76

Game sequence 76

Force field / Shield 76

Overview

About the game

Terracom is a 3D shooter in the style of Starfox. This Adobe Flash game utilizes pre-rendered graphics and scales them in 1-point perspective to create the illusion of 3D. Unlike Starfox, Terracom focuses a lot more on the story and characters.

There are 9 levels, with story scenes placed in between many of them.

To streamline the online experience, levels load in the background while the game is being played.

The feature set is deliberately kept to a minimum in order to allow more focus on the quality of each feature, and to help accommodate the very limited amount of development time. This game is designed to be created in only one month.

Overview

Premise

This is an early rendering.

An alien, artificially intelligent satellite called Terracom visits the Earth and abducts a hacker named Sarah McNelson for study. After an experiment which endangers her life, Sarah gets very angry and takes control of the satellite, reprogramming all of the security to hunt the Terracom AI. To protect itself, the AI goes into hiding somewhere in the satellite.

Meanwhile, at Nickelson Airforce Base, Commander McNelson sends Lieutenant Daemon on a mission to rescue his granddaughter from Terracom. They are both unaware that she has taken control of the satellite.

Characters

About this section

In this section, the role and personality of each character is described.

Lieutenant Daemon

Daemon is an experienced air force pilot at Nickelson Air Force Base, who takes orders from Commander McNelson. He is cool-headed and adapts well to unusual circumstances.

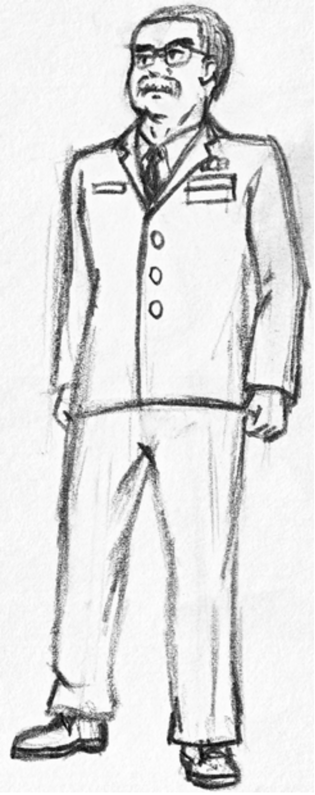
The player plays as this character.

Sarah McNelson

Sarah McNelson, alias: Data Sarah, is a very skilled hacker, who has been abducted by Terracom. Her grandfather is Commander McNelson. She is reckless and quick to anger. After being abducted, she takes control of the Terracom satellite and reprograms all of the security to hunt the Terracom AI.
All units hacked by Sarah, have a red color in their windows or eyes.

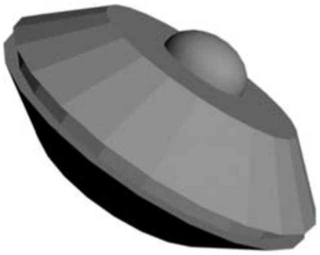
Characters

Commander McNelson



Commander McNelson works at Nickelson Air Force Base and is the grandfather of Sarah McNelson, though he is lothe to admit it. After Sarah is abducted, he assigns Lieutenant Daemon the mission of rescuing her.

Terracom AI



This is an early rendering.

Terracom is a satellite of alien origin. The AI is a separate entity acting as the “brain” of the satellite. It brought the satellite to earth on a journey of exploration and study.

After abducting Sarah for study, it accidently endangers her life in an experiment. As a result, she gets very angry and hacks the Terracom satellite’s security, causing the AI to go into hiding to protect itself. Hacked units are referred to as Hunters.

Storyline

About this section

In this section, the sequence and content of the story scenes are detailed.

Storytelling method

The story is told using cut-scenes between levels. During these scenes, dialogue is exchanged between characters and a picture is shown.

The picture usually portrays the character who is speaking, but is sometimes used to convey the situation instead.

The dialogue is portrayed using scrolling text.

Storyline

Scene 1

When: After the title screen.

Picture: Terracom Satellite.

Mission Briefing

Nickelson Air Force Base 13:00 hours.

Seven hours ago, an artificial structure of unknown origin dropped into Earth's orbit.
We have dubbed this presence "Terracom."

At first, there was every indication that Terracom was a benign entity up until 2 hours ago, when one of our former programmers was abducted.

The programmer in question is Sarah McNelson, who was terminated only yesterday after we discovered that she had hacked into 5 classified military databases.

She possesses strategic information which must not be allowed to fall into the wrong hands.

We have set up a makeshift launch-pad on the roof of a skyscraper in the city. It's designed for the prototype space-jet that we have been working on.

Storyline

Scene 1 (continued)

Lieutenant, your mission will be to quickly make your way to the launch site.
After liftoff, you will infiltrate Terracom, extract Sarah McNelson, and return to Nickelson Air Force Base.

Be on the lookout for foreign aircraft and any other hostiles. We do not know what Terracom is up to yet.

Level 1A occurs - Grassy fields

Scene 2

When:	After level 1A (the first level).
Picture:	A squadron of jets taking off.

Lieutenant, 15 minutes ago, a large squadron of experimental fighter jets took off from Nickelson Air Force Base on their own!

The bogeys you just encountered were
some of them.

We can only assume that Terracom is behind this, and that it is somehow utilizing Sarah's hacking skills.

Storyline

Scene 2 (continued)

The rest of the squadron is headed for the launch site. You must get there before they do!

Take the underground tunnel. It's a shortcut that neither Sarah nor Terracom know about.

Level 1B occurs – Underground cave

Level 1C occurs – Skyscraper

Scene 3

When: After level 1C (the skyscraper level).
Picture: The space-jet launching into space.

Lieutenant, you're on your own from here.
We do not have the equipment to contact you in outer space.

Make your way to the Terracom satellite, but be careful. Our spy satellites have detected a large number of unidentified objects coming from the Terracom satellite.

Storyline

Level 2B occurs – Outer space

Level 2C occurs – corridor outside Terracom

Scene 4

When: After level 2C (the corridor level).

Picture: The AI attached to the space-jet.

That's strange. It felt like something just hit my plane. But there doesn't seem to be any damage.

Uh oh, I see incoming bogeys up ahead.

Level 3A occurs – Holding cells

Scene 5

When: After level 3A (the holding-cell level).

Picture: Sarah McNelson.

Are you the programmer, Sarah McNelson?

That's HACKER extraordinaire to you!

Never mind, I'm here to rescue you.

Storyline

Scene 5 (continued)

Rescue!? Hah!

You have no idea what's going on, do you?

I have taken over the satellite!

Why?

Why!? For my own safety, that's why!

The Terracom's AI tried to electrocute me!

But I showed that stupid machine.

I've turned the whole satellite against it!

So why is everything attacking me?

Duh! Because the AI latched onto your ship the moment you got here!

So that's what it was.

If you hadn't showed up, my hunters would've destroyed the AI by now!

Is that why you sent those aircraft after me back on Earth?

Storyline

Scene 5 (continued)

Naturally, for your own protection.

They tried to shoot me down.

Of course! Another presence here would only confuse my hunters. Fortunately, they think your plane is the AI.

So how do I get rid of it?

Head for the laboratory section and download my virus from the computer there.
That outta fry its circuits!

Level 4A occurs - Laboratory

Scene 6

When: After level 4A (laboratory).
Picture: The Terracom AI.

Incoming transmission...

What? From who?

Please... save me...

Storyline

Scene 6 (continued)

Who is this!?

I am… this place.
I am… the mind.

Terracom?

Yes… save me.

Why should I?
You tried to kill Sarah!

A mistake.
Humans are unexpectedly fragile.
I came to study… not to harm.

How do you expect me to save you!?
You pissed off a hacker and everything on the satellite is after you!

Then take me outside the station.
There can be no hunt… when there is no prey.
There can be no conflict… when there is no enemy.

Storyline

Scene 6 (continued)

Why don't we just go talk to Sarah?

She is "pissed off" … maybe crazy.

Hmm… you have a point there.
She'll have to calm down
before you can talk to her.

Level 5A occurs – Escape the satellite

Scene 7

When: After level 5A (escape).
Picture: Sarah being chased by her hunters.

Incoming transmission…

NOW who is it!?

Yo, pilot dude! Come in!

Sarah!?

Get your hunk-a-junk back in here!
With the AI gone, my hunters think I'm the
"foreign presence!!"

Storyline

Scene 7 (continued)

And how do I stop the hunters?

Go to… the core.

What!? That chip-head's still with you!?

I did not intend to harm you.

Yeah, you're real sorry now, aren't ya!

A mistake…
But I could deactivate the hunters from the
satellite's core computer.

Fine! Pilot dude, keep an eye on chip-head there!
Make sure it doesn't try anything funny.

Level 6A occurs – The core

Storyline

Scene 8

When:	After level 6A (the core).
Picture:	A star-filled night sky on Earth.

So ol' chip-head was tellin' the truth!
It only came here to study us.

It said that it was leaving.
That humans are much too terrifying.

Nah! Humans ain't terrifying… I am!!

You're just crazy.

Levels

About this section

In this section, the sequence of story scenes and levels are shown. The levels are described in detail with a description of their setting, the boss and enemies present, and a description of the gameplay involved in the level. Information about the bosses is also summarized here, with detailed information in the Bosses section of this document.

Scene 1 occurs

Level 1A – Grassy fields

In this level, the space-jet is traveling extremely fast, as though it's using its afterburners.

The enemies are hacked air force jets.

Setting:	Gigantic grass-covered farm fields.
Enemies:	Jets that approach from behind you. Jets in the distance that rush forward.
Strategy:	Avoid the jets from behind you. Shoot down distant jets before they rush forward. Dodge farm silos on the ground. Shoot scarecrows to release items that replenish hit points.
Boss:	N/A

Scene 2 occurs

Levels

Level 1B – Underground Cave

There are no enemies in the level, but the player must survive an obstacle course of hazards.

Setting:	An underground cave.
Enemies:	Falling rocks. Stalagmites and Stalactites.
Strategy:	Shoot and dodge falling boulders. Dodge stalagmites and stalactites. Shoot falling boulders to release HP.
Boss:	N/A

Level 1C – Skyscraper

The space-jet is flying up the side of a very tall skyscraper, while missiles shower the side of the building.

Setting:	The side of a skyscraper.
Enemies:	Air force jets. Missiles.
Strategy:	Dodge missiles. Dodge or shoot kamikaze jets.
Boss:	It’s a Battle Chopper hovering over the roof. Its weak point is the center of its propellers, which flashes red.

Levels

Scene 3 occurs

Level 2B - Outer space

The player is in space, flying over the Earth, with the Terracom satellite in the distance gradually getting closer.

Setting:	In space, over the Earth.
Enemies:	Meteors. Terracom fighters.
Strategy:	Dodge incoming meteors. Shoot meteors to release HP. Snipe distant Terracom fighters before they come close to shoot at you.
Boss:	N/A

Levels

Level 2C – Corridor

The space-jet is flying along a corridor on the surface of the Terracom satellite.
This level resembles a scene from Star Wars.

Setting: In a technological corridor.

Enemies: Wall-mounted gun turrets.
Terracom fighters which approach from behind.

Strategy: Shoot distant turrets before they fire.
Avoid fighters coming from behind.

Boss: It’s a secured door with gun turrets.
Destroy all the turrets to win.
The turrets open and close, and are only vulnerable while they are open.

Scene 4 occurs

Levels

Level 3A – Holding cells

The player is flying through the section of Terracom where specimens are stored.

Setting:	Holding cells.
Enemies:	Blocking gates. Cross-beams. Security cameras. Terracom fighters.
Strategy:	Avoid being seen by security cameras. Each time the player is seen, fighters are summoned. Gates block the path and must be shot. Cross-beams must be dodged.
Boss:	Laser boss. Avoid the targeting reticule to avoid getting blasted by a huge laser beam.

Scene 5 occurs

Levels

Level 4A - Laboratory

The player travels through the laboratory section of the satellite where experiments take place.

Setting:	Laboratory.
Enemies:	Regular Terracom fighters. Strong Terracom fighters. Moving laser beams. Blocking gates.
Strategy:	Dodge moving laser beams. Shoot blocking gates. Snipe distant Terracom fighters. Fight cunning Terracom fighters.
Boss:	Shadow boss. The laboratory computer is destroyed by the boss when it arrives. The boss creates false copies of itself. Shoot the correct one to damage it. The correct one shoots missiles. The copies shoot lasers.

Scene 6 occurs

Levels

Level 5A – Escape

The satellite is on high alert. The space-jet is flying at breakneck speed through the holding cells.

Setting: The holding cells.

Enemies: Strong Terracom fighters.
Mines.
Closing doors.

Strategy: Fly through narrow holes in the security doors while they're closing.
Snipe the mines, or give them a wide berth.
Fight cunning Terracom fighters.

Boss: Mine layer.
This boss releases mines, occasionally turning around to look at the player.
This boss is only vulnerable while it's facing the player

Scene 7 occurs

Levels

Level 6A – The core

The space-jet is flying through a circuit-laden area. Circuit cubes are abundant and some move unexpectedly.

Setting: Inside the core computer.

Enemies: Moving laser beams.
Moving circuit cubes.
Destructible circuit cubes.
Strong Terracom Fighters.

Strategy: Dodge fast-moving laser beams.
Fight cunning Terracom fighters.
Fly around moving circuit cubes.
Shoot through destructible cube walls. (It's like digging)

Boss: Shield Bit.
This boss moves around erratically while its force field is up.
It is vulnerable when it lowers its shield to fire at you.
When firing, it shoots 3 shots at a time, doing this twice.

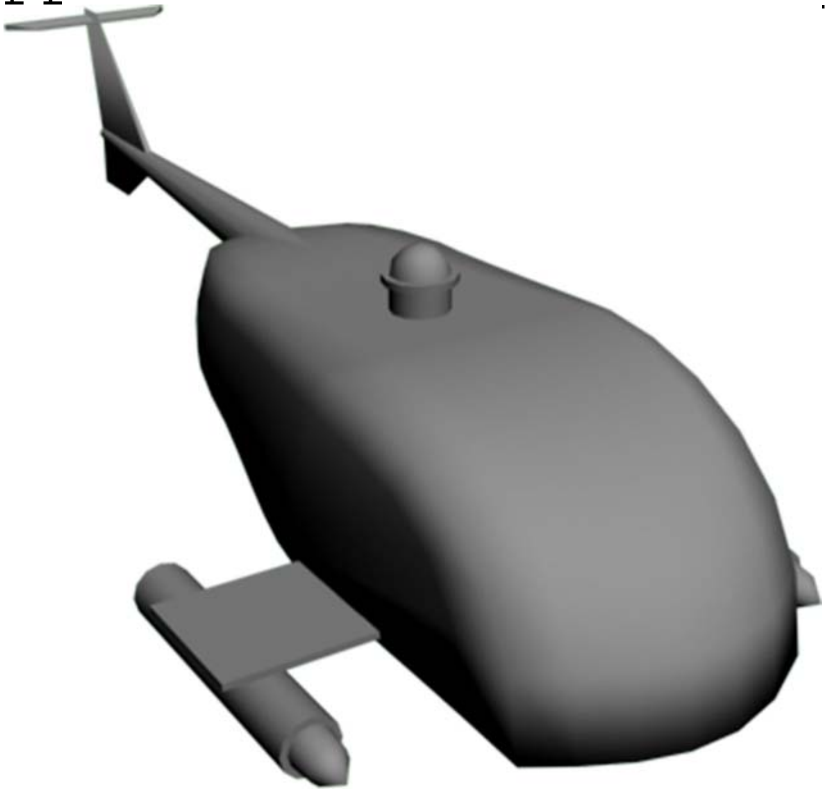
Scene 8 occurs

Bosses

About this section

In the Bosses and Enemies sections, all of the game's adversaries and hazards are described. Their movement patterns are explained, and the levels they appear in are listed.

Battle Chopper



This is an early rendering.

This boss fires a rocket at the player, then moves suddenly to another location. It does this three times. Afterwards, it waits a bit longer than normal, then fires 2 rockets simultaneously.

This enemy is found in level: 1C

Bosses

Security Door

This is a boss.

There are 3 gun turrets mounted around its perimeter. They are different from normal gun turrets in that they emerge and hide. They do this in a clockwise order.

When hidden, they cannot be damaged.

While revealed, they are vulnerable.

When all three of them are destroyed, the door explodes and the battle is over.

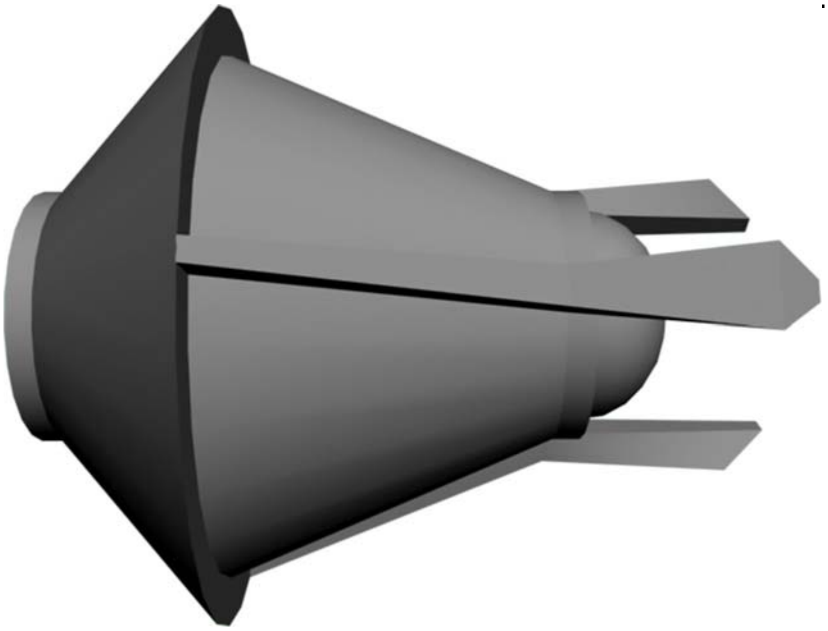
When a turret emerges, it waits briefly, fires 2 lasers at the player, then closes.

After that, the next turret emerges and follows the same pattern.

This enemy is found in level: 2C

Bosses

Laser Hunter



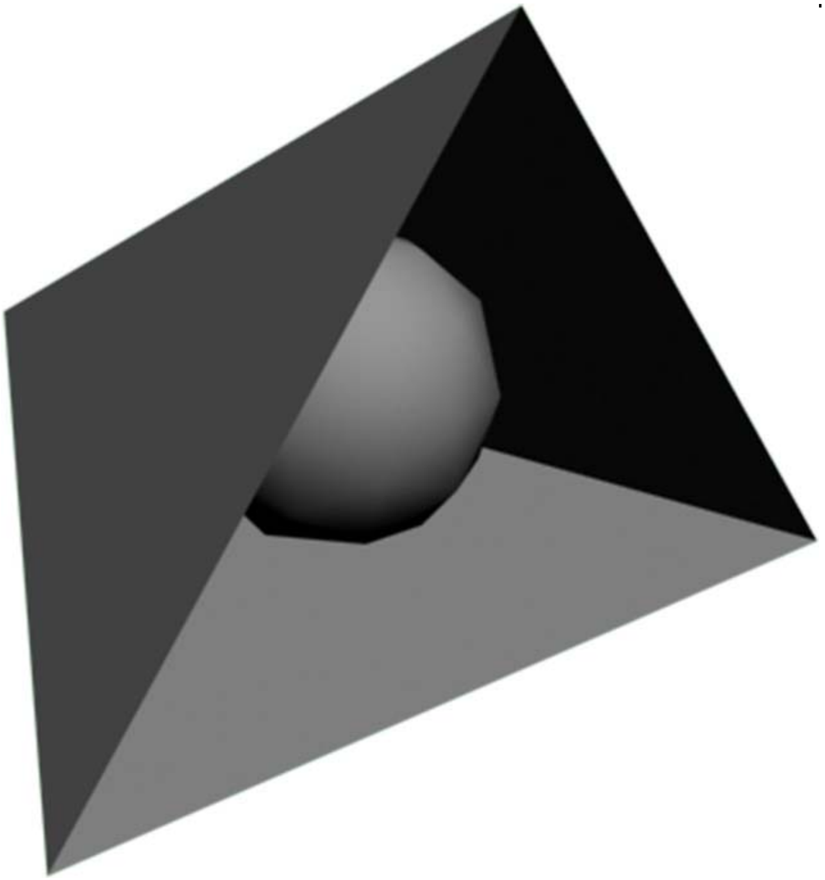
This is an early rendering.

This boss moves slowly and randomly.
Its targeting reticule follows the player around.
When the reticule touches the player, the boss briefly charges up its laser and fires a huge blast at that location.

This enemy is found in level: 3A

Bosses

Shadow Hunter



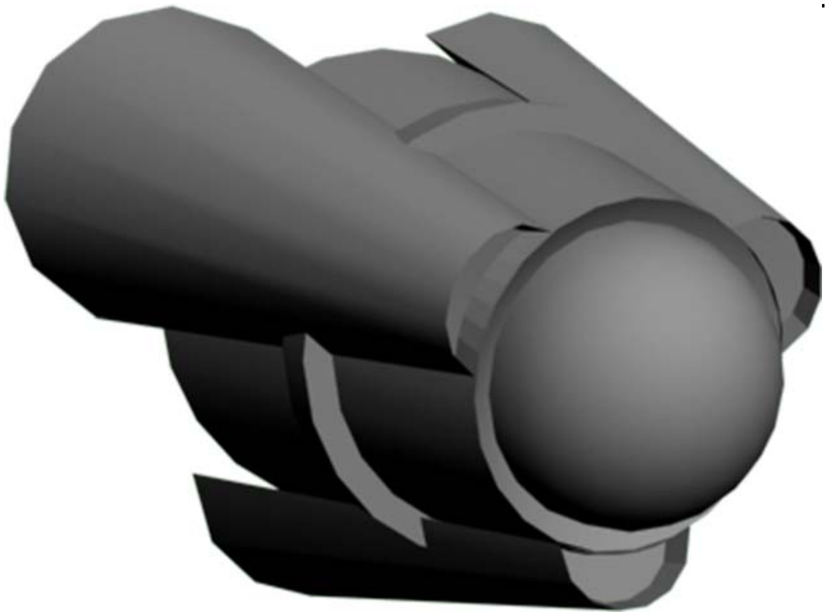
This is an early rendering.

The front of this boss ripples like fabric. It waits a moment, then splits into 3 images of itself. Only one of them is vulnerable to damage. The vulnerable one fires rockets at the player. The false copies shoot lasers at the player. When it takes damage, the copies disappear, and the remaining boss moves to the center. Then the pattern repeats.

This enemy is found in level: 4A

Bosses

Mine Layer



This is an early rendering.

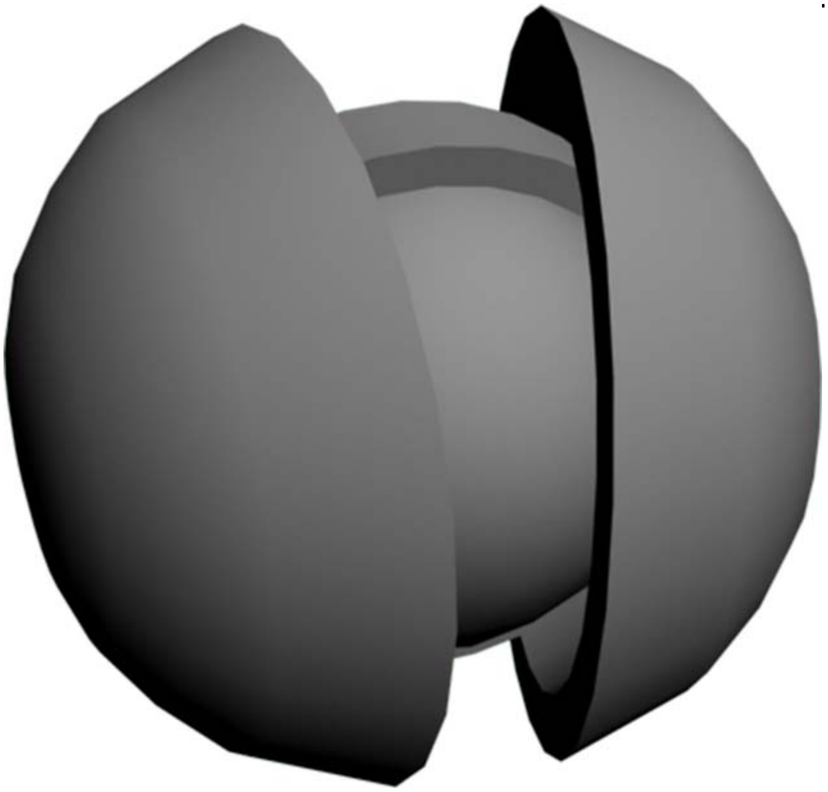
This boss has its back to the player most of the time. It moves to a random position, then releases a mine. It does this 3 times. Then it turns around to face the player, pauses, fires a rocket, turns away, and repeats its pattern.

It is only vulnerable when it is facing the player.

This enemy is found in level: 5A

Bosses

Shield Bit



. This is an early rendering.

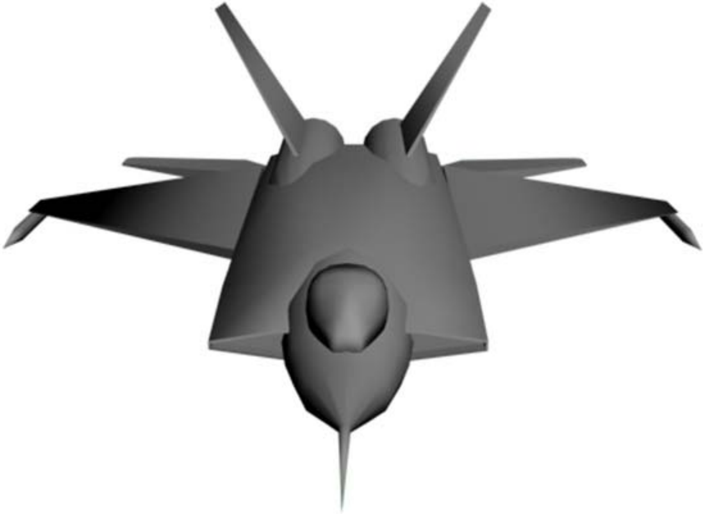
This boss is only vulnerable when it lowers its force field to fire at the player.

With its shield on, it moves around erraticly, then suddenly stops, switches off its shield, and fires three simultaneous lasers. After firing 2 times quickly, it repeats the pattern.

This enemy is found in level: 6A

Enemies

Jet A



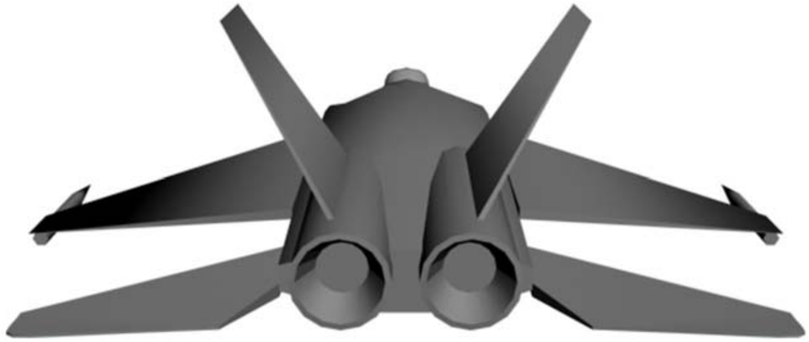
This is an early rendering.

This foe dives down in the distance, then rushes forward.

This enemy is found in levels: 1A, 1C

Enemies

Jet B



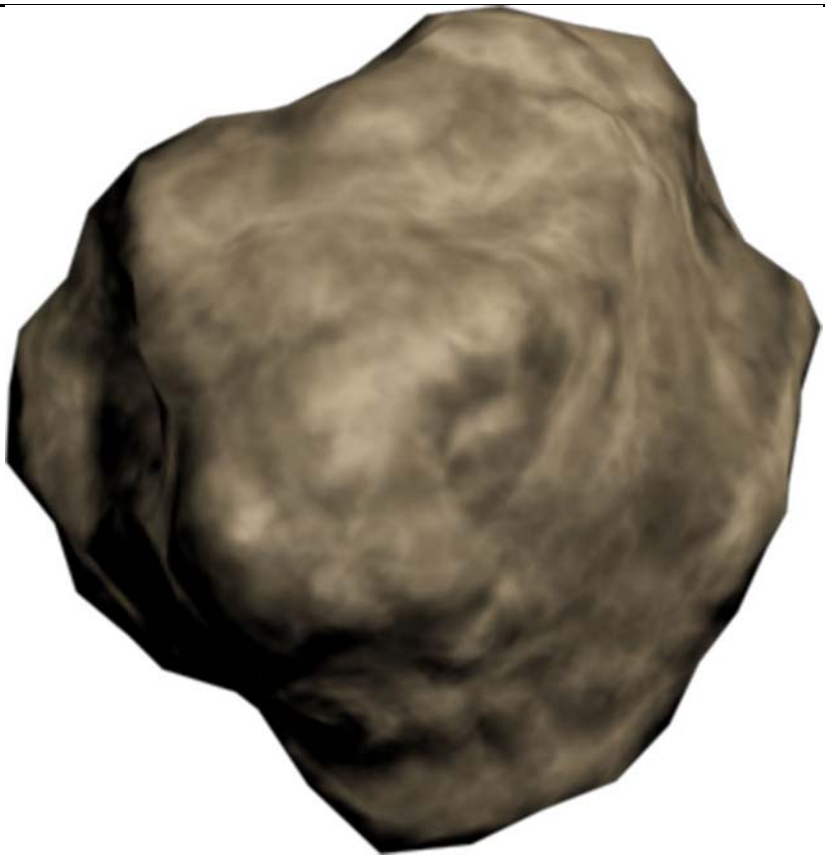
This is an early rendering.

This foe approaches from behind the player.
It waits there a moment, flies past, turns around,
fires a rocket, waits a moment, then rushes forward.

This enemy is found in levels: 1A, 1C

Enemies

Boulder



This is the final rendering.

This hazard falls from above and breaks upon impact with anything.

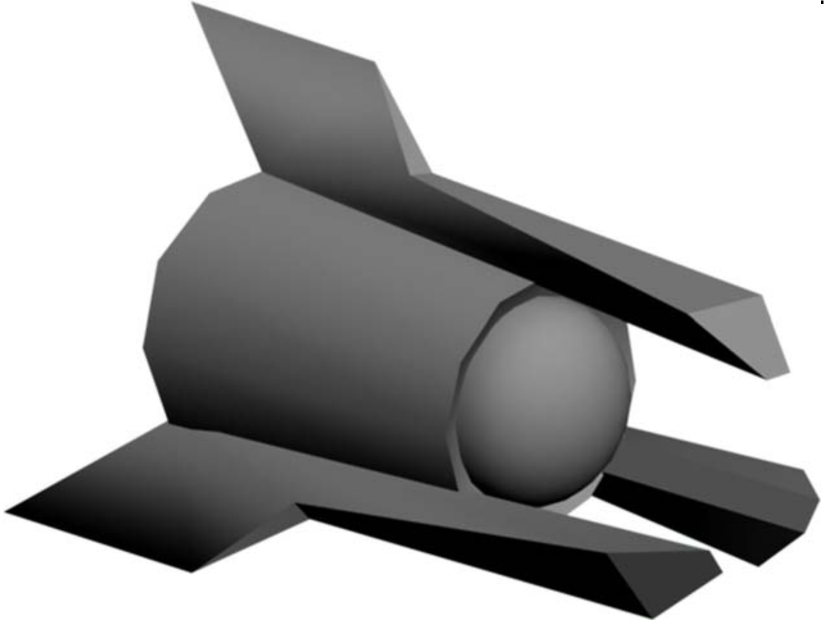
Occasionally, it'll release an item which can replenish the player's health.

The AI of this enemy is also used for the missile shower in level 1B.

This enemy is found in level: 1C

Enemies

Terracom Fighter A



. This is an early rendering.

It waits in the distance, approaching slowly.

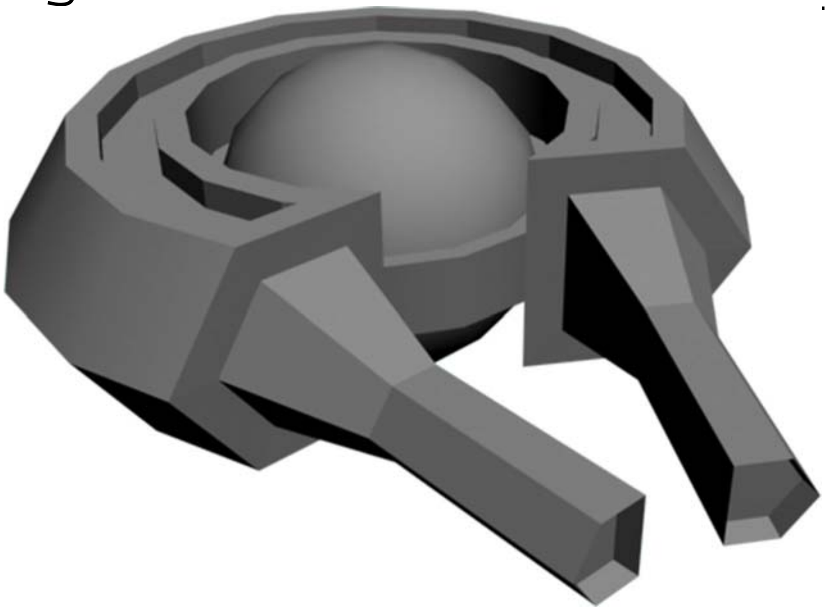
Then it rushes up close, fires some lasers, and retreats into the distance again.

From there, it repeats the pattern.

This enemy is found in levels: 2B, 4A

Enemies

Terracom Fighter B



This is an early rendering.

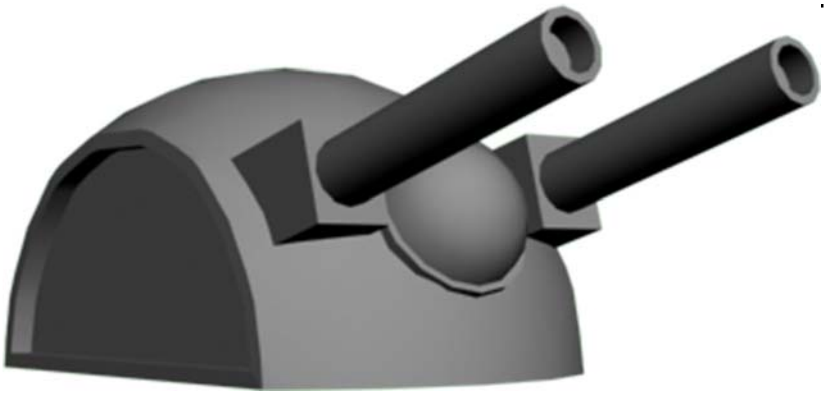
This version of the enemy follows the same movement pattern as “Jet B,” except that it also fires a laser before moving in front of the player.

This enemy shoots lasers instead of rockets.

This enemy is found in levels: 2C, 3A

Enemies

Gun Turret



This is an early rendering.

As it draws close, it fires 3 lasers at the player. Without interruption, it continues its movement until it passes the player and disappears.

This enemy should appear to be moving with the level, as though it is attached to the
floor, wall, or ceiling.

This enemy is found in level: 2C

Enemies

Security Camera

This is an early rendering.

This enemy cannot be harmed.

When the player flies in front of it,

an alarm sounds, and a “Terracom Fighter B” arrives.

This enemy is found in level: 3A

Laser Beam

This hazard moves either up and down,

or left and right, in a repeating pattern.

It cannot be damaged.

This enemy is found in levels: 4A, 6A

Closing Security Door

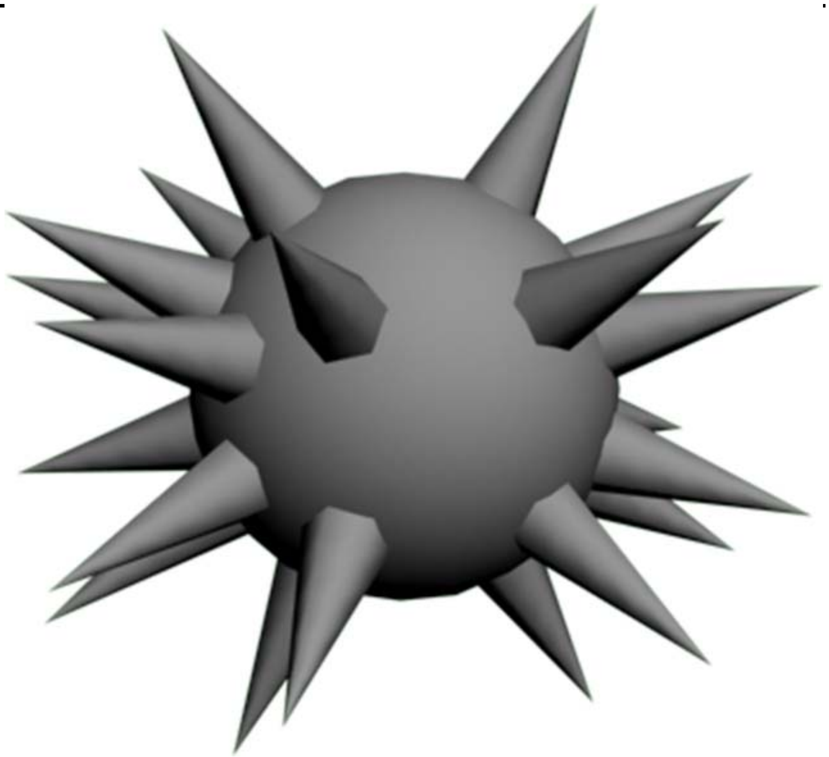
This hazard cannot be damaged. It slowly closes,

leaving only a small hole to fly through.

This enemy is found in level: 5A

Enemies

Mine



This is an early rendering.

If the player is near it when it gets close, the mine explodes. Shooting it will cause it to explode prematurely.

These are scattered throughout one of the levels. The Mine Layer boss also releases these.

This enemy is found in level: 5A

Enemies

Circuit Cube

This hazard cannot be damaged.

Some circuit cubes move, others don't.

The moving ones are each programmed to move to a specified location after waiting a specified amount of time, thereby creating unexpected and interesting obstacle courses to dodge through.

This enemy is found in level: 6A

Destructible Circuit Cube

This hazard is different from regular Circuit Cubes.

It's an obstacle and doesn't change position.

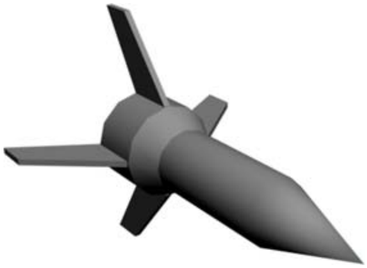
Unlike regular Circuit Cubes,

Destructable Circuit Cubes look different and can take damage.

This enemy is found in level: 6A

Special Enemies

Standard Projectiles



This is an early rendering.

There are 2 kinds of projectiles. Lasers and Missiles. These are actually simple enemy sprites that initially set their direction toward the player, and then blindly move in that direction until they collide with something or go offscreen.

The difference between these two projectiles is that Missiles leave a smoke trail behind them.

Simple obstacles

This is an early rendering of a farm silo.

Simple obstacles are objects such as farm silos, meteors, stalagtites, cross-beams, blocking gates, scarecrows, and non-moving circuit cubes.

These objects simply approach in sync with the level’s motion. Most of the time, they appear to be attached to part of the level, such as the floor, walls, or ceiling.

Some can be destroyed, some cannot.

Those that can be destroyed, might release a healing item for the player to catch.

Interfaces

About this section

The game is separated into units called “interfaces.” There are 5 types of interfaces:

- Loading Screen
- Game Over
- Title screen
- Story scene
- Levels

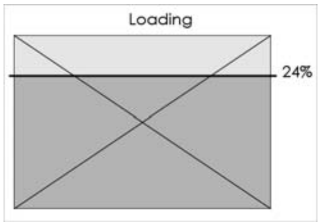
This section describes the appearance and general programming of each type of interface. Interfaces are self-contained SWF files, which are loaded in the background while the previous interface plays out. For example, while Level 1A is being played, Level 1B is loading in the background. After loading, it waits in standby mode.

Each interface exists as a “movieClip” object at the “_root” level. Each new interface is layered on top of the previous one. When an interface completes its program, it tells the next interface to run and erases itself. The game is basically a chain reaction of interfaces running one at a time, in sequence.

The programming for each interface is organized into a series of frames on its timeline.

Interfaces

Loading Screen



This is the rough layout

This interface is visible if an interface is needed, but hasn't finished loading yet.

Under ideal conditions,

this interface would never be seen.

It consists of:

- A picture of a 3D wireframe of an object
- A picture of the full-rendered 3D object
- A mask
- A line
- A percentage indicator
- The text, "Loading"

This is the interface's structure:

```
_root
 └ Loading Screen
    ├ Wireframe Image
    ├ Rendered Image
    ├ Mask
    ├ Line
    ├ Percentage indicator
    └ Loading text
```

Interfaces

Loading Screen (continued)

When this interface is placed, the instance name of another interface is passed to it.
The Loading Screen evaluates that interface and displays its loading progress.

Initially, the wireframe image is visible and the fully-rendered picture is not.
As the specified interface loads, the mask scales downward, revealing the rendered picture on top of the wireframe. This creates the illusion that a 3D object is being rendered.

While this happens, the line follows the top of the mask, accentuating the division between the two pictures.
Next to the line is the percentage indicator, which constantly updates to show how much of the interface has been loaded.

Interfaces

Loading Screen – Frame Actions _

1 – Setup

Check whether or not the specified interface has finished loading.

Loaded - Go to Ending

Not done loading - Go to Main Program

2 – Main Program

Scale the mask, based on the load progress.

Place the line on top of the mask.

3 – Loop

Do the Main Program until the interface is fully loaded.

4 – Ending

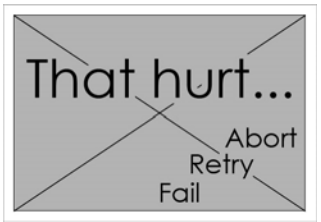
Tell the specified interface to begin playing.

5 – Self-destruct

Delete the Loading Screen interface.

Interfaces

Game Over Screen



This is the rough layout

This interface appears when the player presses ESC, or when the player runs out of hit points.

It consists of:

- A background image
- 3 buttons
- The text, “That hurt...”

This is the interface’s structure:

```
_root
 └ Game Over Screen
    ├ Background
    ├ Text
    ├ Abort Button
    ├ Retry Button
    └ Fail Button
```

Each button changes what it says when the player mouses over it. The three buttons are:

- Abort / Abort Game
- Retry / Retry Level
- Fail / Return to Title Screen

The frame actions are shown on the next page.

Interfaces

Game Over Screen – Frame Actions

1 – Standby

This frame stops the timeline.
When it’s time for this interface to be used,
a play() command is issued to it from
the “player” sprite.

2 – Setup

Fade-in the Game Over screen.

3 – Main Program

Program the buttons:
- Abort: Closes the entire program
- Retry: Restarts the level from the beginning
- Fail: Loads & plays the Title Screen

4 – Ending

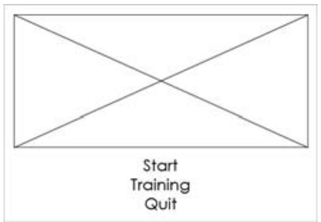
Deactivate the buttons.
Flash & Fade-out the Game Over screen.

5 – Self-destruct

Delete the Game Over interface.

Interfaces

Title Screen



This is the rough layout

This interface only appears once:

When the game is first started.

It consists of a picture and two buttons.

The buttons are:

- ❖ Start
- ❖ Training
- ❖ Quit

This is the interface's structure:

```
_root
 └ Title Screen
   ├ Background Picture
   ├ Start Button
   ├ Training Button
   └ Quit Button
```

The frame actions are shown on the next page.

Interfaces

Title Screen – Frame Actions

1 – Standby

This frame stops the timeline.
When it's time for this interface to be used,
a play() command is issued to it from
its associated Load Screen interface.

2 – Setup

Begin loading the interface: Story Scene 1.
Begin loading the interface: Level 1A.
Fade-in the title screen.

3 – Main Program

Start button: Create a Load Screen interface
to evaluate and run Story Scene 1.
Training Button: Create a Load Screen
interface to evaluate and run Training Level.
Quit button: Close the program.

4 – Ending

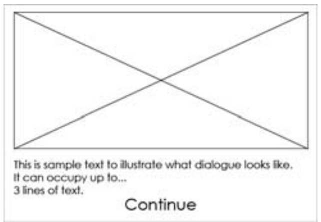
Fade-out the title screen.
Deactivate the buttons.

5 – Self-destruct

Delete the Title Screen interface.

Interfaces

Story Scene



This is the rough layout

This interface appears after the Title Screen and between levels.

It consists of a picture,
vertically scrolling text,
and a “Continue” button.

This is the interface’s structure:

```
_root
 └ Story Scene
    ├ Picture
    ├ Scrolling Text
    └ Continue Button
```

The frame actions are shown on the next page.

Interfaces

Story Scene – Frame Actions

1 – Standby

This frame stops the timeline.
When it’s time for this interface to be used,
a play() command is issued to it from
its associated Load Screen interface.

2 – Setup

Fade-in the Story Scene.

3 – Main Program

Scroll the text until it reaches the end,
then stop.
Continue Button: Go to Ending

4 – Loop

Go to the Main Program.

5 – Ending

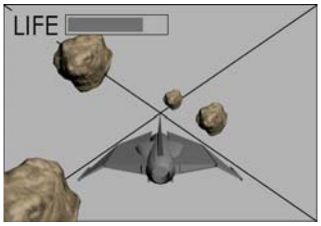
Fade-out the Story Scene.
Deactivate the “Continue” button.
Create a Load Screen interface for
the next interface in the game sequence.

6 – Self-destruct

Delete this Story Scene interface.

Interfaces

Level



This is the rough layout

A level can occur after other levels and after story scenes.

Levels preceding a story scene often have a boss enemy at the end of them.

Levels are labeled based on the last story scene that was played.
For example: Levels 1A, 1B, and 1C occur after story scene 1.
Levels 2B, and 2C occur after story scene 2.

Due to time constraints, one or two levels have been cancelled from the production of this game.
Cancelled levels: 2A

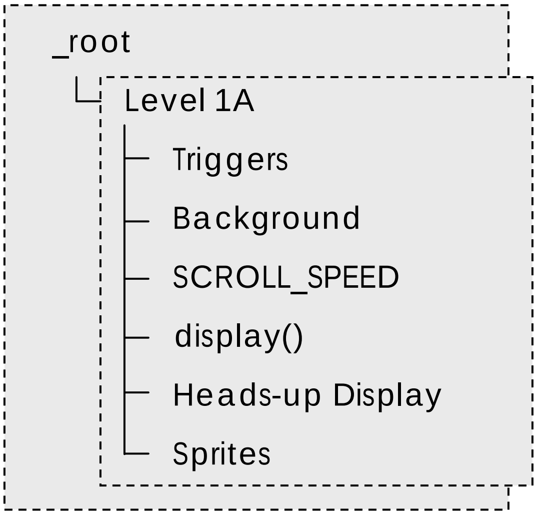
Interfaces

Level Objects

A level consists of the following objects:

- (movieClip) Triggers
- (movieClip) Background
- (static variable) SCROLL_SPEED
- (function) display()
- (movieClip) Heads-up Display
- (movieClip) Sprites

This is the interface's structure:



Interfaces

Level Objects - Triggers

As the timeline of this movieClip plays,
frame actions create enemies and obstacles.
After being created, enemy and obstacle sprites
control themselves.
When a boss battle occurs, the timeline enters
a loop until the boss is defeated,
then continues playing.
When the timeline finishes playing, it stops and
creates a Load Screen interface for the next Level
or Story Scene in the game sequence.

The first sprite that is created is the “player” sprite,
which is the space-jet controlled by the player.

Level Objects - Background

This movieClip contains an embedded video which
usually loops endlessly.

Some levels halt the animation of the background
after a stationary boss appears.
Some levels have more than one video embedded
into their background. This is utilized when a more
seamless transition between two levels is desired.
In both cases, a change in the behavior of this
movieClip is initiated from the “Triggers” movieClip.

Interfaces

Level Objects - SCROLL_SPEED

This is a static variable. Therefore its name is all caps with an underscore seperating the words in accordance with the standard naming conventions of programming.

This variable specifies the rate of movement along the Z-axis that would move a sprite at the same apparent speed as the background animation. Sprites that approach at this speed, seem to be attached to the background. This is desirable for enemies such as gun turrets and cameras.

Level Objects - display()

This is a function. Therefore its name ends with a pair of paranthesis in accordance with the standard naming conventions of programming.

This function places, scales, and sorts the depth of all the sprites according to each one’s Z coordinate. This creates the illusion of sprites moving in 3D space. The actual movement of each sprite is controlled the sprites themselves.

Interfaces

Level Objects - Heads-up Display

This movieClip displays the hit points of the player. Hit points are set to their maximum at the beginning of each level. When the player’s hit points are zero, the player dies and must restart the current level from the beginning.

This movieClip contains a frame-based loop which allows it to constantly check the hit-points of the “player” sprite and update its display accordingly.

Level Objects - sprites

All sprite movieClips are placed inside of this movieClip in order to make it easy to loop through them. This is necessary when performing hit-detection between sprites.

A sprite is any enemy, obstacle, visual effect,
or player-controlled object.
Visual effects are temporary animations
such as explosions.

Due to their complexity, Sprite objects are discussed in more detail later in the document.

Interfaces

Level – Frame Actions

1 – Standby

This frame stops the timeline.

When it’s time for this interface to be used,
a play() command is issued to it from
the previous interface in the sequence.

2 – Setup

Fade-in the Level.

Begin loading the next level.

Begin loading the next story component
if it’ll be needed after this level.

3 – Main Program

display()

4 – Loop

Do the Main Program.

5 – Ending

Fade-out the level.

Tell the next interface to begin playing.

6 – Self-destruct

Delete the current level interface.

Sprites

About this section

A sprite is any enemy, obstacle, visual effect, or player-controlled object in a level. Visual effects are temporary animations such as explosions.

This section describes what sprites are, what they consist of, and how they work.

Sprites

Sprite structure

A sprite cosists of:

- ❖ (layer) triggers
- ❖ (movieClip) object
  - ➢ (movieClip) collision
- ❖ (function) checkCollision()
- ❖ (function) enterFrame()

This is a sprite's structure:

```
_root
  └ Level Interface
      └ Sprites
          ├ Sprite Instance A
          │   └ triggers
          │     object
          │          Image
          │          collision
          │     checkCollision()
          │     PARENT_SPRITE
          │     _z
          │     zVelocity
          │     hp
          │     MAX_HP
          │     enterFrame()
          │
          └ Sprite Instance B
```

Sprites

triggers

This timeline layer directly controls the sprite's attacks, movement along the Z-axis, and poses.

Boss sprites have long and complex triggers.

Enemy sprites have moderately complex triggers.

Obstacle sprites have very simple triggers.

object

This movieClip is a container
which holds two movieClips:

- ❖ image
- ❖ collision

The "image" movieClip is the sprite's graphic. Multiple images can be stored in this movieClip if the sprite has more than one pose. When that's the case, each image is on a separate frame.

The "collision" movieClip is used by the checkCollision() function when checking for collisions between sprites.

This movieClip can have more than frame if the sprite is not always vulnerable to damage,
or has changing weak points.
Invincible sprites have empty "collision" movieClips.

Sprites

Object (continued)

The sprite's horizontal and vertical position is manipulated by animating the location of the "object" movieClip in the sprite's timeline.

The animation of the "object" movieClip and the frame actions of the "triggers" layer jointly control the movement of the sprite.

checkCollision()

This function is called externally by the "main program" frame actions of a Level interface.

When called, this function loops through all but two of the other sprites and checks to see it any of them collide with this sprite.
The looping check ignores two sprites:

- The current sprite
- The sprite name specified by PARENT_SPRITE

When a collision is detected,
this sprite's "hp" variable is reduced by one.

Sprites

PARENT_SPRITE

This is a static variable that is passed to a sprite if it is created by another sprite.

This is typically used for a projectile sprite to prevent it from damaging the ship that shot it.

xPos

This variable stores the horizontal position of the "object" movieClip.

yPos

This variable stores the vertical position of the "object" movieClip.

zPos

This variable represents the sprite's coordinate along the z-axis. The higher the value, the farther away the sprite will appear.

This variable is manipulated by
the sprite's "triggers" movieClip.

Sprites

zVelocity

This variable represents the speed that the sprite is traveling along the z-axis.

A positive value makes the sprite move away.
A negative value makes the sprite move closer.

This variable is manipulated by the sprite's "trigger" frame actions.

hp

This variable represents the number of collisions that this sprite can sustain before self-destructing.

"hp" stands for "hit points."

MAX_HP

This static variable represents the maximum value allowed for the "hp" variable.

Sprites

enterFrame()

This function does four things.

It calls checkCollision() to determine whether or not this sprite is colliding with another sprite in the level.

It checks the “hp” variable of this sprite.
If the “hp” is at or below zero, a visual effect is created at the sprite’s position, and the sprite is deleted. This allows enemies to explode when defeated.

It adds the current zVelocity variable the zPos variable of this sprite, allowing a “triggers” action to easily move the sprite along the z-axis.

It copies the horizontal and vertical coordinates of the “object” movieClip into the xPos and yPos variables respectively.

Music

About this section

This section lists what music is in the game, and when they are used.

The game has four pieces of music:

- Title / Credits
- Story Scene
- Level
- Boss

The number of songs is mainly limited due to time constraints.

All of the music used by an interface is stored inside of it. Because of this, it remains possible for each level to have different music.

The style and usage of each song is described on the next page.

Music

Title / Credits

This song is played at the title screen and during the ending credits.

It's style is inspired by the song "Life on Strings."

Story Scene

This song is played whenever a "Story Scene" interface is running.

It's style is inspired by the music in the
Super Nintendo game, "Wing Commander."

Level

This song is played whenever a "Level" interface is running.
However, it will stop playing when a boss appears.
This is because boss battles have their own music.

It's style is inspired by the music in the
Sega CD version of the game, "Shadow Run."

Boss Battle

This song begins playing when a boss enemy appears.

It's style is inspired by the music in the
Playstation game, "Metal Gear Solid."

Production Schedule

About this section

Terracom is designed to be produced in only one month by a single student; Cory Thomas.

As a result, it has a very tight production schedule, which is shown here.

Weekly schedule

Each week is spent on a specific aspect of the game's production.

Spring break	Story & Design Document
Week 1	Programming
Week 2	Programming & Graphics
Week 3	Graphics
Week 4	Music & Sound
Week 5	Project Due

Production Schedule

Daily schedule

The spring break was spent designing the game.

To ensure that all of the necessary components were fleshed out, a daily schedule was followed.

Monday	Code structure & Methodology
Tuesday	Premise & Story
Wednesday	Concept Art
Thursday	Concept Art
Friday	Design Document
Saturday	Design Document
Sunday	Design Document

Terminology

About this section

This section explains the definitions of some of the custom terminology that is used in this document.

AI

This is an acronym for "Artificial Intelligence."

Interface

Interfaces are self-contained SWF files, which are loaded in the background while the previous interface plays out. For example, while Level 1A is being played, Level 1B is loading in the background.

Sprite

A sprite is a specially-structured movieClip that is used for any enemy, obstacle, visual effect, or player-controlled object in a level.

Visual effects are temporary animations such as explosions.

Terracom

Terracom is the name of this game.
The game is named after one of its characters, an artificially intelligent satellite of alien origin.

The word itself is completely made-up.

Terminology

Hunter

A hunter is any enemy that has been programmed to hunt the Terracom AI.

Hacker

In this game, the term “hacker” is used in the more connotative sense to refer to a computer genius who is capable of circumventing security and creating viruses.

This definition technically fits the meaning of the word “cracker,” but most audiences are unfamiliar with that term.

Game sequence

This refers to the sequential order in which the game’s interfaces are run.

Although each interface is self-contained and can be run on its own, the story is only coherent when the interfaces are experienced in
the right sequence.

Force field / Shield

Both of these terms refer to energy-generated barriers. There is a boss in the game which uses one.

Projectile

This is a simple sprite that looks like a fired shot.