



An Atmospheric Adventure  
to the Core of the Earth

# Contents

<b>1</b>	<b>What is Activated Core?</b>	<b>5</b>
1.1	<i>Basic Story</i> . . . . .	5
<b>2</b>	<b>How do I Play Activated Core?</b>	<b>6</b>
2.1	<i>Installing Activated Core</i> . . . . .	6
2.1.1	Installing From Source . . . . .	6
2.1.2	Precompiled Package . . . . .	6
2.2	<i>Starting Activated Core for the First Time</i> . . . . .	7
2.3	<i>Standard Controls</i> . . . . .	8
2.4	<i>Mission Begins – Playing the First Level</i> . . . . .	9
<b>3</b>	<b>Activated Core Gameplay</b>	<b>10</b>
3.1	<i>Difficulty Settings</i> . . . . .	10
3.1.1	Quick Comparison . . . . .	11
3.1.2	Normal . . . . .	12
3.1.3	High School . . . . .	12

3.1.4	Masters Degree . . . . .	12
3.1.5	Professor . . . . .	12
3.2	<i>Mixed Gameplay</i> . . . . .	14
3.2.1	Platforming . . . . .	14
3.2.2	Action . . . . .	14
3.2.3	Puzzle . . . . .	14
<b>4</b>	<b>Developing Own Game Content for Activated Core</b>	<b>15</b>
4.1	<i>Creating Maps/Levels</i> . . . . .	15
4.1.1	Main Mapfile . . . . .	16
4.1.2	Available Basic Objects . . . . .	17
4.1.3	Information Objects and Weapons . . . . .	18
4.1.4	Moving Objects . . . . .	18
4.1.5	Pickup Items and General Upgrades . . . . .	19
4.1.6	Main Map Graphics (skin) . . . . .	20
4.1.7	Background Graphics (optional) . . . . .	20
4.1.8	Main Collision Map . . . . .	20

**5 Tips from the Professor** **21**

5.1 *Cracking the Codes* . . . . . 21

# 1 What is Activated Core?

*Activated Core* is a game which intends to mix several game genres, including platforming, action, puzzle and more. The first version was only a tile based platformer, which then expanded beyond the tiles with large bitmapped maps. This allowed for fast rendering which was needed back in the days. The game was originally programmed in Visual Basic, using the extraordinary powers of DirectDraw 4 in the DirectX library. This was year 2004. By the means of switching permanently to Linux, the development of this version faded gradually and eventually died out, until May 20:th 2010. The game development was then restarted, resulting in a completely rewritten game engine in C/C++ using OpenGL and GLUT.

## 1.1 Basic Story

---

Our character, **Include Character Name**, was given a mission to explore a distant planet. On her way back to earth, she discovered at a distance that the earth atmosphere now have a purple hue. As she is getting closer and closer she picks up a transmission from earth. . .

The distress call mentioning something about foreign lifeforms converting earth into a machine, before it vanishes. When entering through the atmosphere, a virus enters the onboard computers which causes the shields to go down. There's only a short matter of time before the ship breaks down and crashes.

By entering a cave, near the crash site, she discovers a hive of alien lifeforms, operating on the earth's soil. She begins an infiltration process of an alien vessel buried deep inside the cave.

## 2 How do I Play Activated Core?

*Time to prepare yourself for an adventure to the core of the earth!*

### 2.1 *Installing Activated Core*

---

**Activated Core** is somewhat self contained, and does therefore not need to be installed.

#### 2.1.1 **Installing From Source**

If you are installing the game from source, there is a makefile provided. Just make sure you have the OpenGL and GLUT development libraries on your computer. You can make the executable file by typing,

```
> make -f Makefile
```

in your terminal. Make sure you are located in the `src` directory when performing the command.

#### 2.1.2 **Precompiled Package**

Life is easy. Just extract the `.tar.gz` to anywhere.

## 2.2 *Starting Activated Core for the First Time*

---

After installation, you can start the game either from the terminal by,

```
> ./activate
```

or by just running the **activate** file in the installation directory.

After watching through (or skipping by using **ESC**) the intro, you are presented with the profile screen. Here you can select which profile to use, or create a new one. Proceed with a choice and enter the main menu. From here you can access the main options, explained shortly below,

<b>Start Game</b>	<i>Starts a new game from the beginning.</i>
<b>Continue</b>	<i>Continues the game from a save point. This menu item is unavailable until the first save.</i>
<b>Settings</b>	<i>Adjust settings of the game or selecting a different player profile.</i>
<b>Review Intro</b>	<i>Start the Intro from the beginning.</i>
<b>Quit</b>	<i>Exits <b>Activated Core</b></i>

## 2.3 *Standard Controls*

---

Below is the default keyboard bindings. These settings can be changes from the settings menu.

<b>Command</b>	<b>Key</b>	<b>Description</b>
Exit	Esc	<i>Exit terminals and menus</i>
Confirm	Return	<i>Accepts terminal commands and menu selections</i>
Left	A	<i>Move player to the left</i>
Right	D	<i>Move player to the right</i>
Jump	L	<i>Jump - Hold for higher jumps</i>
Crouch	S	<i>Crouch to be able to reach narrow spaces</i>
Fire	K	<i>Fire the selected weapon</i>
Next Weapon	R	<i>Selects next available weapon modification</i>
Previous Weapon	E	<i>Selects previous weapon modification</i>
Flashlight	F	<i>Toggles flashlight (if equipped)</i>
Enter Terminal	W	<i>Try to enter a terminal</i>



## 2.4 *Mission Begins – Playing the First Level*

---

The first level is trying to introduce all styles of gameplay to the player at the first time. The briefing informs that the virus that entered the ship computers shut down the shields. This is making the ship burn to much when entering through the earth modified atmosphere.

The virus also triggered the self destruction sequence, which is standard equipment on all intergalaxy explorer boats.

Given the problems at hand, the mission of the first level is to disable the automatic destruct sequence, and escape the ship with an escape pod. This is a quite hard task, but there is no time limit, and the challenge should give a fair introduction to all gameplay elements.

The virus also made the helper droid in the base of the ship to go haywire. There are two solutions on stopping the self destruction, either go into one of the terminals and stop the countdown by entering the abort code (which can be found by reading the contents of a special data disc), or get past **Helpy**, the droid, to deactivate the destruction hardware.

## 3 Activated Core Gameplay

*Platforming, Action, Puzzle. It's all part of the game.*

### 3.1 *Difficulty Settings*

---

There are several difficulty settings to select from when starting a game. The selection is unlike most existing game difficulties. The difficulty setting for **Activated Core** affects all gameplay elements described on the next page.

The main settings are,

1. Normal
2. High School
3. Masters Degree
4. Professor

### 3.1.1 Quick Comparison

Variable	Normal	High School	Masters Degree	Professor
Story Instructions	Clear	Plenty	Some	None
Platforming	Easy	Normal	Hard	Easy
Action	Easy	Normal	Hard	Easy
Puzzle	Very easy	Easy	Challenging	Impossible

The elaborate the puzzle differences between the different settings one can say that if the target is to open a door by using a password, the *Normal* mode would be to get a data disc (item) and use it on a terminal to open the door. Very much like any key in another game. For the *High School* settings, the same action would require the user to enter a command into the terminal to read the disc. To find out the command is a one time challenge, but will be the same for all terminals which uses this system. The *Masters Degree* might require the player to enter a `/dev/disc0` folder inside a terminal, open a binary file and extract the password, then use this together with a command. The difference is not that much, but for a user not familiar with terminal environments, this might be hard enough. The task of opening a door at *Professor* level might require some cracking of the code as well, by using the tools available in the terminal. As the toolset varies between terminals, and the cracking require different solutions and sometimes not even a software based solution, this is very hard to figure out at occations, and should only be played by people who likes impossible puzzles and challenges!

A description of all difficulty settings follow on the next few pages.

### 3.1.2 Normal

This is the standard gameplay of **Activated Core**. The enemies are not very touch, indicating a low level artificial intelligence. The puzzles are easy to figure out, meaning that it's enough to find a certain data disc to operate on a terminal and open a door. Anyone should be able to play through **Activated Core** on this level, but some elements may be pretty touch anyway.

### 3.1.3 High School

The *High School* difficulty of **Activated Core** introduces smarter and stronger enemies. This is the main difference from *Normal* difficulty. However, some puzzles may require some commands given to a terminal in order to operate on the data disc.

### 3.1.4 Masters Degree

Select this setting if you have the courage of taking on really strong enemies and is pretty familiar with the unix environment to get through the puzzle parts. This is actually the overall hardest difficulty setting of the game.

### 3.1.5 Professor

For this mode, the enemies are only as strong as in the *Normal* difficulty setting, but the puzzles are very hard. In this mode, there are no given clues on how to proceed through the levels, and the player must figure this

out on his/her own. Given this, the *Professor* setting for **Activated Core** gives more emphasis on the puzzle elements and exploration, and might give a more rewarding experience to smart people.

## 3.2 *Mixed Gameplay*

---

### 3.2.1 **Platforming**

The main gameplay is platforming elements. This is all exploration of new grounds, progressing through the story. The difficulty of the platforming increases fast, with the first few levels fairly small and easy to find the way through, with quite obvious puzzle elements. But pretty soon, it becomes a maze of survival, with insane leaps, traps and more. There is a mix of fast paced platforming and slow atmospheric exploration.

### 3.2.2 **Action**

Blast your way through the levels, either if it's platforming, shooter etc. Intense action is there to give contrast to the other elements.

### 3.2.3 **Puzzle**

My idea for this is mainly hacking sequences. Either if it is cracking a code, password, or hacking terminals to open doors. Some parts may require some really tough thinking when playing on hard difficulty settings.

## 4 Developing Own Game Content for Activated Core

*Inteded for online multiplayer use, but I might consider some code changes to make it work for singler player stories as well.*

### 4.1 *Creating Maps/Levels*

---

The files for a map has to be specified as,

[mapname]	<i>The main mapfile</i>
[mapname]gfx.tga	<i>Main map graphics (skin)</i>
[mapname]gbg.tga	<i>Background graphics (optional)</i>
[mapname]col.tga	<i>Main collision map</i>
[mapname]ext	<i>Animated map graphics definitions</i>

### 4.1.1 Main Mapfile

The first 5 lines of the mapfile has to be, in order of appearance,

```
Level Name  
Author  
Width  
Height  
Music Index
```

**Level Name** and **Author** can both be 128 characters wide. **Music Index** is an internal index for a song. The **Width** and **Height** fields are defined as a number of 32x32 tiles, so a map of 1024x1024 pixels would translate to 32 and 32 respectively. Example mapfile header:

```
My Cool Map!  
Jiisuki  
32  
64  
3
```

The example above would load the map *My Cool Map!*, created by *Jiisuki*. The map is 32x64 in tilesize, which is 1024x2048 pixels in size. The music used is the first song, *Mission Begins*.



The lines that follow lists all objects on the map. An object is defined by one description line which makes the loading interpreter go into the correct segment. The Object name comes first, and the syntax on the second line. For example,

```

TERMINAL
176 118 0 -1 128 42 0
General Utility Terminal
> Welcome Line 1
> Welcome Line 2
> Welcome Line 3

```

#### 4.1.2 Available Basic Objects

Object	Syntax	Object Description
TELEPORT	X Y ToID Key SpawnLeft	<i>A teleport. The teleports get a unique ID when created. The first ID is 0, the next is 1 etc. Key specifies if the terminal is locked for input until a Data Disc is collected. SpawnLeft specifies if you spawns at the target teleport on the left side</i>
SPAWN	X Y Team Time	<i>A position where the player starts. A spawn is also a save point.</i>
TERMINAL	X Y ToID Key W H GfxType	<i>A Hacker terminal, see above example on how to define it, as it has more lines than a normal object</i>
SWITCH	X Y ToID Key GfxType	<i>A switch to operate doors, elevators etc.</i>

### 4.1.3 Information Objects and Weapons

<b>Object</b>	<b>Syntax</b>	<b>Object Description</b>
DISC	X Y ID Respawn	<i>Data disc which contain terminal codes and other information.</i>
BLASTER	X Y InitAmmo Respawn	<i>The first weapon of the game.</i>
GRENADERS	X Y InitAmmo Respawn	<i>Grenade Launcher modification.</i>
RAILGUN	X Y InitAmmo Respawn	<i>Railgun modification.</i>
TRAILGUN	X Y InitAmmo Respawn	<i>Modification (upgrade) to an existing railgun modification.</i>
FLASHDISC	X Y Strength	<i>A flashdisc used for destroying terminals and open doors.</i>

### 4.1.4 Moving Objects

<b>Object</b>	<b>Syntax</b>	<b>Object Description</b>
DOOR	X Y Height Dir Type Key	<i>A door. Can be given a key needed for opening. Type sets lock type.</i>
ELEVATOR	X1 Y1 X2 Y2 Speed Type Key	<i>Elevator, syntax similar to door.</i>

### 4.1.5 Pickup Items and General Upgrades

Object	Syntax	Object Description
HEALTH	X Y Strength Respawn Time	<i>Gives a health replenish.</i>
BATTERY	X Y Strength Respawn Time	<i>Replenish battery power for weapon upgrades.</i>
ARMOR	X Y Strength Respawn Time	<i>Gives some armory upgrade specified by Strength.</i>
HEARTBEAT	X Y Strength Respawn Time	
CAP	X Y Strength Respawn Time	
AMMOGRENADE	X Y Strength Respawn Time	

#### **4.1.6 Main Map Graphics (skin)**

This file contains the graphical overlay for the collision map. The format should be 32-bit TGA, with reference to the Bottom Left and using RLE compression. The last 8 bits (alpha channel) is used for transparency.

#### **4.1.7 Background Graphics (optional)**

This is a background image, which should be twice as large as the main map graphics and collision map! The background graphics scrolls slower in the game, so it has to be larger to not disappear when at the edges of the screen. This graphics is optional.

#### **4.1.8 Main Collision Map**

This is a background image, which should be twice as large as the main map graphics and collision map! The background graphics scrolls slower in the game, so it has to be larger to not disappear when at the edges of the screen. This graphics is optional.

## 5 Tips from the Professor

*Need some tips for beating the game at the Professor level? The **Activated Core** Professor gives you some pointers here!*

### 5.1 *Cracking the Codes*

---