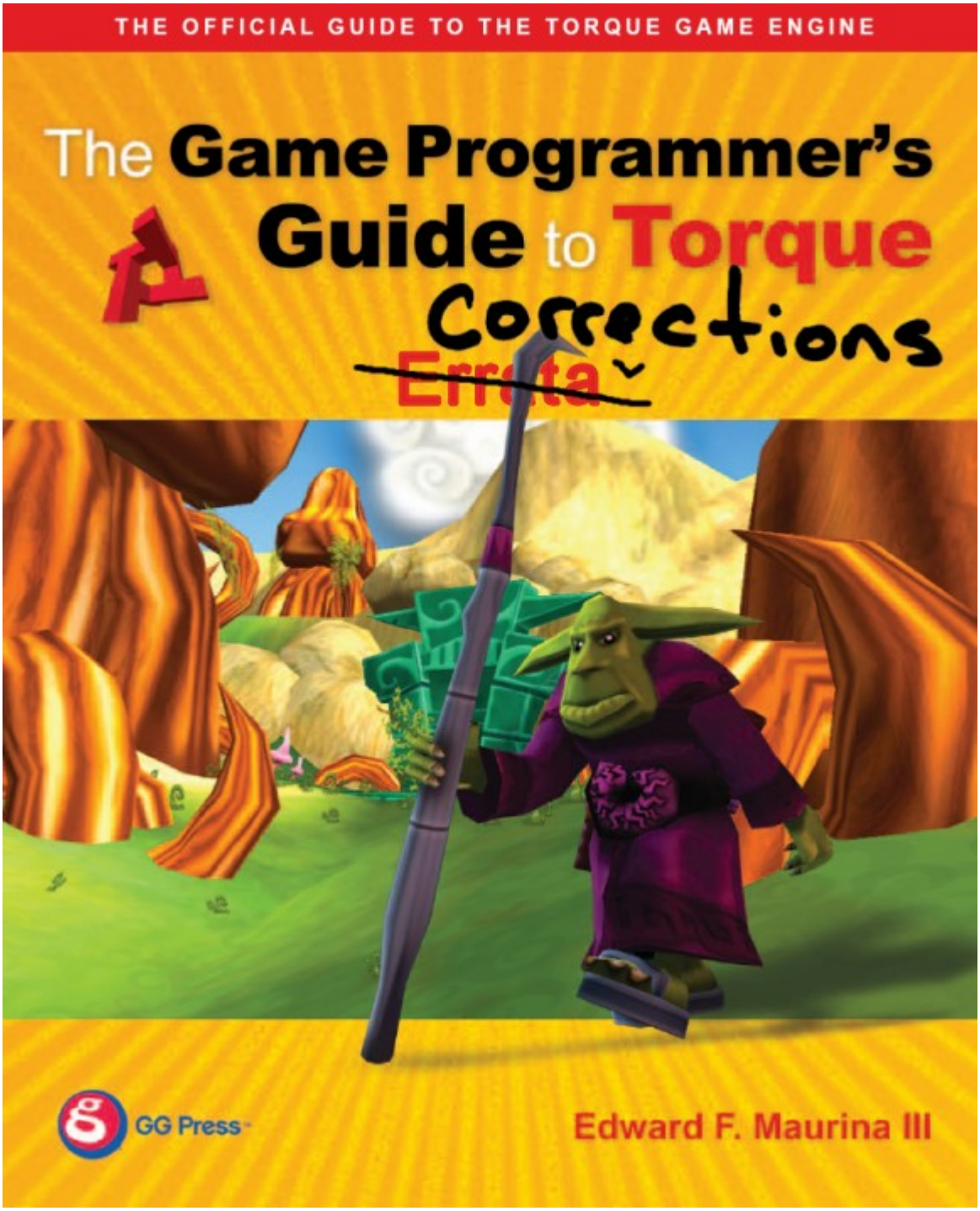
© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.



~~Errata~~ Corrections as of 17 May 2006

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Table of Contents

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 90 – MainMenu<u>Gui</u>.gui vs. MainMenu.gui

Bug Credit: Hokuto

Description:

@ Ed Maurina: I think there is a problem in Chapter 3 - Page 90/91

Page.90

Last bullet point says: *"File->SaveGUI, select MainMenuGui.gui".*

That won't work, I think you should save to *"mainMenu.gui'* instead, then it will work.

Response:

Hokuto is correct.

The Main Menu interface in the GPGT lesson kit has a top-level GUI named MainMenu. Therefore, the engine will assume you want to save it in the file MainMenu.gui. In fact, I did save it in the file "MainMenu.gui".

This problem is a result of my switching to using the GPGT lesson kit as the example base versus using starter.fps. If you are using the starter.fps MOD as a base, the Main Menu interface has a top-level GUI named MainMenuGui. Therefore, the engine will assume you want to save the file as MainMenuGui.gui.

So, if you are playing around with the GPGT lesson kit, save your file as "MainMenu.gui". If you are playing around with the FTS Starter Kit (starter.fps), save the Main Menu as "MainMenuGui.gui".

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 91 – MainMenu.gui Path Is Wrong

Bug Credit: Hokuto

Description:

Page.91

1. Open the file "gpgt\client\Interface\mainMenu.gui"

it should be:

"gpgt\client\Interface\mainMenu\mainMenu.gui"

Response:

Yes, I moved files around a bit during the design of the kit and it seems I missed this path update in the book. The Main Menu interface for the GPGT lesson kit is located under:

"gpgt\client\Interface\mainMenu\mainMenu.gui"

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 94 – Incorrect Path

Bug Credit: Roto

Description:

Roto notes that the following statement (on page 94 near the top of the page) is incorrect:

> wake it. This loading is done in various places. The organization of GUI loading scripts is beyond the scope of this section, but for completeness, I'll show you how to get your new GUI loaded.
> **Open the file "egt\client\init.cs" and search for the following code.**

Correction:

The last line should read:

"Open the file "GPGT LessonKit\gpgt\client\init.cs" and search for the following code."

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 107 – Missing Commas

Bug Credit: Hokuto

Description:

In Chapter 4, on page 107, there is a syntax error in two lines of code:

```
//bt06();
$a = "This is a regular string";
$b = 'This is a tagged string';
echo("Regular String: " $a);
echo("Tagged String: " $b);
```

There is a comma missing in both of the bold lines of code above.

Correction:

The code should read as follows:

```
//bt06();
$a = "This is a regular string";
$b = 'This is a tagged string';
echo("Regular String: ", $a);
echo("Tagged String: ", $b);
```

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 110 – Incorrect Example Output Given

Bug Credit: Justin Mosiman

Description:

In Chapter 4, on page 110, A snippet of code is provided:

```
//bt12();
$srcRay = "1.0 0.0 1.0";
$destRay = "1.0 6.0";
echo ( VectorAdd( $srcRay , $destRay ) );
```

Below this, I say that the output will be: **1 6 1**

This is incorrect.

Correction:

The correct output for this code snippet will be: **2 6 1**

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 119 – Sample Unclear

Bug Credit: Hokuto

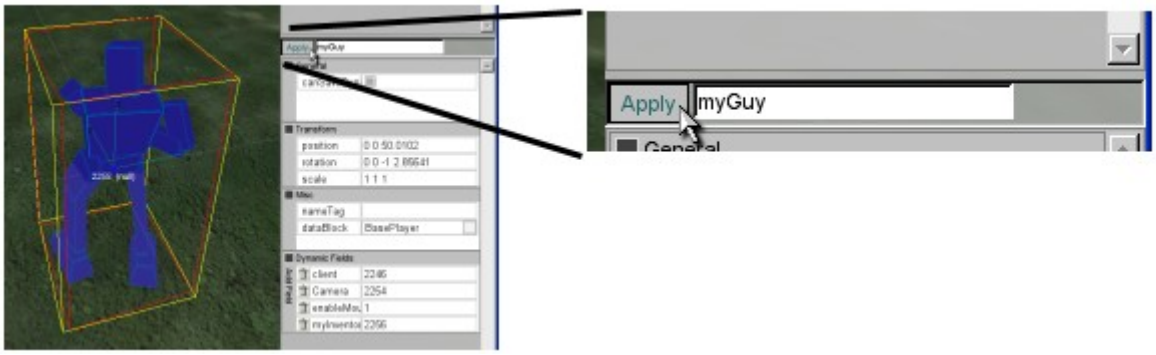
Issue Description: Hokuto notes the following:

> *"If I attempt to follow what the book asks to do on one of the GPGT lesson Kit's missions, the code won't work.*
>
> *When you type:*
>
> *$Player_id = $Player_name.getID(); //second line of code*
>
> *I get <input> (0): unable to find object: 'myGuy' attempting to call function 'getId'"*

Issue Cause: The instructions are probably a little unclear. As well, if you are skimming (which is perfectly reasonable) you might miss the instructions a few lines above.

Issue Resolution: To get this example to work, you must first do the following:

1. Open the mission editor (F11) and switch to the Inspector (F3).
2. Toggle to 'Free Camera' mode (ALT+C)
3. Move a little ways back so you can see the player.
4. Select the player.
5. Name the character by placing the string "MyGuy" next to the Apply Button in the inspector and then clicking Apply. (see image below).



6. Now, you may proceed with the sample.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 129 – Note Block Sentence Incomplete

Bug Credit: Justin Mosiman

Issue Description: Justin notes the following about the note block on the right side of the page:

It reads:

"For example, a p-zone can be used to change the gravity and/or apply a force and/or modify an object's current velocity when the object passed into or through the area encapsulated by the"

and then it stops.

Correction: The sentence should read:

"For example, a p-zone can be used to change the gravity and/or apply a force and/or modify an object's current velocity when the object passed into or through the area encapsulated by the ***physical zone's bounds.****"*

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 132 – Incorrect Instructions in Lesson #2

Bug Credit: Hokuto

Description: Hokuto found that on page 132 (in Maze Runner Lesson #2), there is the following mistake:

> It says:
>
> "2. Now, edit the function onServerCreated() in the file **"\MazeRunner\prototype\server\game.cs"**"
>
> It should say:
>
> **"\MazeRunner\prototype\server\scripts\game.cs"**

Correction: As can be seen, Hokuto has supplied us with the correction. The supplied path (in the book) is in fact missing the "scripts" portion of the path.

© Copyright 2000-2006 GarageGames

Page 133 – Lesson #2 Contains Unclear Steps

Bug Credit: Hokuto

Issue Description: Hokuto notes that the following statement is misleading:

> *pg. 133, paragraph 2 (after code at top of page)*
>
> *"To test for successful load, simply start the kit and load the "MazeRunner"*

Issue Cause:

This statement may well be unclear because the statement uses the word '**kit**'. The use of the word kit seems to imply the use of either the lesson kit or maybe the starter.fps kit. In fact, you are supposed to start your prototype.

Issue Resolution:

1. Be sure that you have followed all of the directions in section "14.4 Setting Up Or Workspace", starting on page 574 and ending on page 577. If you have not created a working area for your prototype, none of the lessons will work.

2. Be sure that you followed lesson #1 on page 72.

3. For all subsequent numbered lessons be sure to run your prototype, not the GPGT lesson kit.

© Copyright 2000-2006 GarageGames

Page 133 – BT18(a) and BT18(b)

Bug Credit: Mark Barner

Description: These two scripts, as supplied on the accompanying and shown on page 133 do not run correctly, producing errors similar to these:

```
==>bt18(a);
~~~~~~~~~~~~~~~~~~~~~
Object 'SimpleTarget1' is not a member of the 'GameBaseData' data block class
gpgt/SampleScripts/stsBasicScripting.cs (461): Register object failed for object CoolTarget of
class StaticShape.
gpgt/SampleScripts/stsBasicScripting.cs (466): Unable to find object: '0' attempting to call
function 'DoIt'
```

```
==>bt18(b);
~~~~~~~~~~~~~~~~~~~~~
Object 'SimpleTarget1' is not a member of the 'GameBaseData' data block class
gpgt/SampleScripts/stsBasicScripting.cs (461): Register object failed for object CoolTarget of
class StaticShape.
gpgt/SampleScripts/stsBasicScripting.cs (470): Unable to find object: '0' attempting to call
function 'DoIt'
Calling StaticShape::DoIt() ==> on object 0
gpgt/SampleScripts/stsBasicScripting.cs (474): Unable to find object: 'CoolTarget' attempting to
call function 'DoIt'
gpgt/SampleScripts/stsBasicScripting.cs (476): Unable to find object: 'CoolTarget' attempting to
call function 'DoIt'
gpgt/SampleScripts/stsBasicScripting.cs (485): Unable to find object: '0' attempting to call
function 'getId'
Syntax error in input.
gpgt/SampleScripts/stsBasicScripting.cs (491): Unable to find object: '0' attempting to call
function 'getId'
<input> (0): Unable to find object: '' attempting to call function 'DoIt'
gpgt/SampleScripts/stsBasicScripting.cs (497): Unable to find object: '0' attempting to call
function 'getId'
Calling StaticShape::DoIt() ==> on object
```

Problem: The sample code on the disk does uses a datablock that was not defined.

Resolution:

1. Open file "/gpgt/sampleScripts/stsBasicScripting.cs" in text editor.
2. Locate *function bt18(%sampleCase)*
3. Modify code as shown below:

```
//bt18(a);
%myTarget = new StaticShape( CoolTarget )
{
    position = "0 0 0";
    // dataBlock = "SimpleTarget1";
   dataBlock = "BaseStaticShape"; // Uses base static shape datablock.
};
```

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 143 – Missing Image?

Bug Credit: Roto

Issue Description: Roto notes:

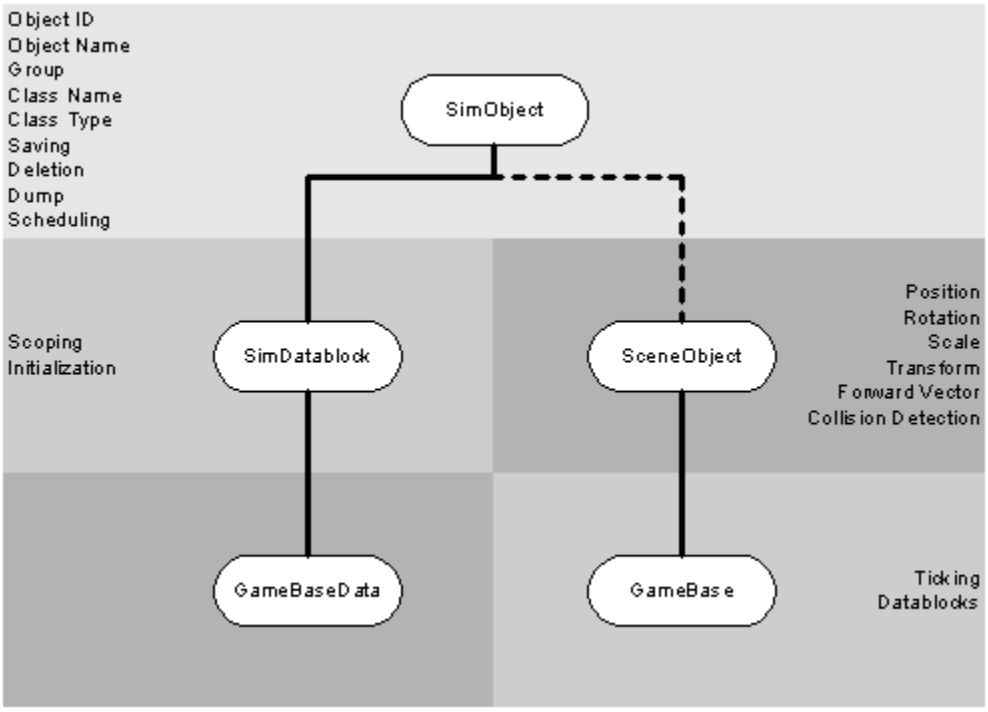
Page 143 the sentence just before section **5.1 SimObject**:

"Please note that the dotted line in Figure 5.1 indicates that there is a class between the two connected classes that we are not discussing"

There is no Figure 5.1. In fact there is not Figures in chapter 5 at all, only tables.

Response: Roto is correct. We pulled that image at the last minute and the text did not get updated.

Here is the image:



© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 181 – Caption Missing Ending Text

Bug Credit: Mark Barner

Description: The Note Caption on page 181 ends in an incomplete sentence.

"This is quite useful, as such mounted shapes can temporarily shield an item from contact and thus from "

Problem: It looks like some words got cut off in copy-editing.

Resolution:

The sentence should read: "This is quite useful, as such mounted shapes can temporarily shield an item from contact and thus from **pickup**."

The idea is that you can temporarily keep items from being picked up by mounting objects or shields onto them. This isn't particularly elegant, but it could have some interesting uses.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 186 – Expert Tip (Reloaders)

Bug Credit: Mark Barner

Description: Mark asks, "The caption reads,"

```
function rldfb() {
    exec("./fadeblocks.cs");
}
```

"but the code shows, "

```
function rldfade() {
    exec("./fadeblocks.cs");
}
```

"Can you name the function anything you want?"

Resolution:

Yes. You may name reloaders anything you want as long as you remember what they are. :) Obviously, I forgot what I named that one.

In all seriousness, I use reloaders while testing code in SinglePlayer mode. This saves me a lot of time without going to the trouble of hooking up a script debugger.

Page 187 – Syntax Error In Sample Code

Bug Credit: Mark Barner

Description: Mark found this syntax error:

```
%object = new TSStatic() {
    position = "0 0 0" <--- is missing ";" in book
    rotation = "1 0 0 0";
    scale = "1 1 1";
    shapeName = "~/data/Shapes/Lessons/GeneralLessonShapes/egg.dts";
};
```

© Copyright 2000-2006 GarageGames

Pages 223 & 226 – Maze Runner Lesson #6 Has Double Copy

Bug Credit: Mark Barner

Description: Mark finds this problem:

Quote (from page 223):

Copy Required Files
From the accomanying disk, please copy the file "**\MazeRunner\Lesson_006\mazerunnerplayer.cs**" into "**\MazeRunner\prototype\server\scripts\MazeRunner**".

...
exec("./MazeRunner/mazerunnerplayer.cs"); //MazeRunner

Quote (from page 226)

To get the entire datablock, please copy "**\MazeRunner\Lesson_006\MazeRunnerPlayer.cs**". to the "**\MazeRunner\prototype\server\scripts\MazeRunner**" directory.

Loading the Datablock

...
exec("./MazeRunner/MazeRunnerPlayer.cs"); // MazeRunner

Problem: The statement on page 226 can be ignored. If you followed the directions on page 223, you already have the file in place. No further changes are needed at this point in the construction of Maze Runner.

Resolution: Ignore all text on page 226 between:

'To get the entire datablock' and ' ...\MazeRunner" directory.'

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 246 – Book vs. Kit Code Mismatch

Bug Credit: Mark Barner

Description: Mark notes:

Chapter 7, section 7.7.1, page246 - 248. Says to follow along with the script "SimpleInventoryGeneral.cs". The variable in the book is "**%theSimpleInventory**" and in **SimpleInventoryGeneral.cs** the variable is "**%theInventory**".

Problem: I caused this issue by renaming some files just prior to shipping the Kit to the publisher. This change came about because late in the editing process, I shortened the names of some files because we ran into a issue compressing them to the ZIP format. There is a fixed limit on the length of a path and filename that can be zipped and later unzipped.

There may be other instances of this type of error.

Resolution: There is no resolution. I will have to correct this issue in later printings and must apologize to those who have a printing with this typo.

© Copyright 2000-2006 GarageGames

Page 257 – Book vs. Kit Code Mismatch

Bug Credit: Mark Barner

Description: Mark notes:

Chapter 7, section 7.7.5, page 257

> Quote:
>
> ---
>
> Take a look in the file " **\MazeRunner\prototype\server\scripts\GPGTBase\Player\PlayerDataConsoleMethods.cs**". It contains the...
>
> ---

The script file on disk is **PlayerDataMethods.cs** that is copied over not **PlayerDataConsoleMethods.cs**.

Problem: I caused this issue by renaming some files just prior to shipping the Kit to the publisher. This change came about because late in the editing process, I shortened the names of some files because we ran into a issue compressing them to the ZIP format. There is a fixed limit on the length of a path and filename that can be zipped and later unzipped.

There may be other instances of this type of error.

Resolution: There is no resolution. I will have to correct this issue in later printings and must apologize to those who have a printing with this typo.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 278 – Spelling Error

Bug Credit: Mark Barner

Description: Mark notes:

"Chapter 8, section 8.3.4 Maze Runner Lesson #8, page 278, table 8.3. Parameters: surfaceTexture, shoreTexture, specularMaskTex values = "**starter.fps/data/GPGTBase/water/...**" should read "**prototype/data/GPGTBase/water/...**" for the Maze Runner lesson. "

Problem: In the original lessons, I used an unmodified version of the FPS starter kit as a base. Subsequently, I chose to use a modified version and supply in under the name 'prototype'. During technical edits, we should have replaced all references of 'starter.fps' with 'prototype', but it looks like this one got by.

Resolution: In all lessons, if you find the word 'starter.fps' treat it as if the word 'prototype' were used instead.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 285 – cloudHeightPer

Bug Credit: Mark Barner

Description: Mark asks:

Chapter 8, section 8.4.9 Maze Runner Lesson #9, page 285, step 3.

> Quote:
>
> 3. Using the Inspector, be sure that the sky has the settings shown in Table 8.4.

Only one of the three **cloudHeightPer** parameters can be change in the Inspector in TGE 1.4. You have to go into the mission file and change the parameters. It must be a TGE 1.4 bug or am I missing something?

Response: No, Mark you are not missing anything. This is in fact a bug with the engine/editors. The cloudHeightPer array is wrongly exposed to the inspector and thus cannot be edited correctly.

The real challenge here is not in fixing the bug, but in making it backward compatible. For now, this is on my list of 'Torque Issues To Resolve/Address'.

Resolution: The only resolution at this time is to hand edit your mission, or to edit the engine code.

Unfortunately, I cannot post an answer in this document as that would violate the engine EULA. However, I can give you a hint.

1. Open the file sky.cc
2. Find the routine "void Sky::initPersistFields()" on about line 279.
3. Compare how 'mCloudSpeed[0]' ... 'mCloudSpeed[2]' are exposed vs. how 'mCloudHeight' is exposed.

During this examination, you may wonder why the 'mCloudSpeed[0]' ... 'mCloudSpeed[2]' is exposed as 'cloudSpeed1' ... 'cloudSpeed3'. Yup. Another legacy bug. On my own projects I always change this to 'cloudSpeed0' ... 'cloudSpeed2'. This too is on my list.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 296 – Thunder Missing From Lesson #11

Bug Credit: Mark Barner

Description: Mark asks:

Chapter 8, section 8.6.6 Maze Runner Lesson #11, page 296, **Adding the Lightning and Thunder.**

Was there supposed to be thunder added in this lesson or is it going to be discussed in a later chapter?

Response: There should be lighting added in that lesson. I'll check into this when I receive my author's copies. Meanwhile, take a look at the 'Lightning' 3D Lesson in the GPGT Lesson Kit if you need to learn about adding thunder.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 325 – Bad Table Reference

Bug Credit: Mark Barner

Description: Mark notes:

Chapter 8, section 8.9.10, page 325, Other Culling Features.

> Quote:
>
> ---
>
> ... When an object is at the perimeter of the outer sphere it will begin to fade, fading completely out at the perimeter of the inner sphere. See **Table 8.25** and Figure 8.27.
>
> ---

It should read **Table 8.26**.

Response: Thanks Mark. I'll get this fixed in a subsequent revision.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Pages 329 thru 331 – Table Numbers Off By -1

Bug Credit: Mark Barner

Description: Mark discovered that starting on page 330, the tables in chapter 8 are incorrectly numbered for two pages.

Specifically, the tables on pages 330 and 331 are numbered 8.26 thru 8.32.

This means chapter 8 has two tables numbered 8.26 and is completely missing 8.33.

This is further confused by the statement on page 329:

"Animations provide the parameters in Table 8.27. Table 8.28-8.33 list the specific parameters for color, brightness, rotation, size, azimuth, and elevation, respectively."

Correction: Never fear. The tables you need are all present. The tables on pages 330 and 331 should be numbered: **8.27 thru 8.33**. The quote from page 329 is in fact referring to the tables on pages 330 and 331.

Page 372 – Wrong File Reference Lesson #17

Bug Credit: Mark Barner

Description: Mark notes:

Chapter 9, section9.5.10 Maze Runner Lesson # 17,page 372, Copy Required Files.

> Quote:
>
> ---
>
> From the accompanying disk, please copy the file "\MazeRunner**Lesson_12\teleporters.cs**" into ....
>
> ....
> exec("./MazeRunner/levelloader.cs"); //MazeRunner
>
> ---

Lesson_12\telporters.cs should be **Lesson_17\levelloader.cs**.

Response: Thanks Mark. I'll get this fixed in a subsequent revision.

Page 384 – TS10()

Bug Credit: Mark Barner

Description: This script fails to execute, producing an error like this:

```
==>ts10();
~~~~~~~~~~~~~~~~~~~~~
gpgt/SampleScripts/stsAdvancedScripting.cs (470): Unable to find object: '' attempting to call
function 'delete'
```

Problem: The sample depends on the presence of a datablock that has no longer available.

The GPGT Lesson Kit script "stsAdvancedScripting.cs" requires that the '3D Lessons' mission be running to execute the samples. This is required because datablocks are used in some examples, which requires certain engine features to be active. Running a mission activates these parts of the engine.

The problem exists because the script "stsAdvancedScripting.cs" is loaded when the application starts. At this time, the datablock '*myTestDatablock*' is created. Subsequently, when you start the '3D Lessons' mission, the kit first deletes all datablocks in preparation to load new ones. If you consider this, you will understand that by the time the mission starts '*myTestDatablock*' has been destroyed. This causes the lesson to fail.

Resolution:

The resolution to this problem is simple, but can be misleading. So, please the caution at the end of this fix.

1. Open file "/gpgt/sampleScripts/stsAdvancedScripting.cs" in text editor.
2. Locate *function ts18( )*
3. Modify code as shown below:

```
// Comment the old datablock definition
//datablock StaticShapeData( myTestDatablock : BaseStaticShape ) {
//   category  = "LessonShapes";
//};

function ts10() {
    if( ! sscls() ) return;

    // Copy the same datablock definition into the body of the sample function
    datablock StaticShapeData( myTestDatablock )
    {
        category  = "LessonShapes";
    };

   %obj = new StaticShape( testObject ) {
      datablock = "myTestDatablock";
   };

   %obj.delete();
}
```

With these fixes you should now get:

```
~~~~~~~~~~~~~~~~~~~~~~
A new object: testObject was created with the datablock: myTestDatablock
Deleting: testObject created with the datablock: myTestDatablock
```

Cautionary Note:

First, this is and advanced topic, so if yo have not read about datablocks yet and if you have not done some samples on your own, this may not mean a lot to you.

In the above fix, we declared a datablock within the body of a function. Doing so is completely legal, but will have certain sideffects, depending upon the game type you are writing. If you are writing a SinglePlayer game, then the engine will automatically find and use this new datablock. However, if you are writing a MultiPlayer game and you create a new datablock **after** datablocks have already been transmitted to all clients, then ONLY THE SERVER will be aware of new datablock and creating objects with it will crash the clients.

I have ascertained (experimentally) that it is possible to create datablocks on the fly after a mission load and then to force them to be transmitted to clients. Again. This is an advanced topic, and I won't be discussing the exact technique until I have verified it works consistently.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 574 – Incorrect Path

Bug Credit: Roto

Description:

Roto notes that the following statement (on page 574 near the top of the page) is incorrect:

> he 3rd bulleted item reads"
>
> **"\LessonKit"**. This directory contains the

Correction:

Instead, this should read:

> he 3rd bulleted item reads"
>
> **"\GPGT LessonKit"**. This directory contains the

© Copyright 2000-2006 GarageGames

Page 576 – Confusing Directions Section 14.4.4

Bug Credit: Roto

Description:

Roto noted that the instructions about updating main.cs are ambiguous. Specifically, there is the possibility of confusing main.cs in our game's root directory with the one in the main.cs in the prototype directory.

Clarification:

In the pages prior to section 14.4.4 we have done the following:

1. Installed a fresh copy of the Torque Game Engine Demo. Let's assume we installed it in **C:\TGE**.
2. Copied the entire installation to a new directory. In my instructions, I said that we could call it: **C:\MazeRunner**.
3. Wrote some cleanup scripts.
4. Copied the mod directory from in the included disk. Assuming that our CD drive is Z:, in section 14.4.3, we would copy Z:\MazeRunner\A_SettingUp\prototype into C:\MazeRunner.

At the end of step 4 (above) we will have something like this structure on our C:\ drive:

C:\MazeRunner
|
|--main.cs (edit this main.cs while reading 14.4.4)
|
|--\prototype
|
|--main.cs

In section 14.4.4 on page 576, I say to edit main.cs and change:

$defaultGame = "demo";

to

$defaultGame = "prototype";

The main.cs file you should be edit is C:\MazeRunner\main.cs.

© Copyright 2000-2006 GarageGames Product of Hall Of Worlds, LLC.

Page 576 – Incorrect Filename

Bug Credit: Roto

Description:

Roto notes that the filename provided for the simple task manager is incorrect:

In the book, it is:

exec("./EGSystems/SimpleTaskMgr/egs_SimpleTaskMgr.cs");

, when it should be:

exec("./EGSystems/SimpleTaskManager/egs_SimpleTaskManager.cs");