«Green Ball»
Game Engine documentation

Marcin Natanek

May 1, 2015

Contents

Chapter 1

Introduction

The «Green Ball» Game Engine is written in C++ with use of OpenGL rendering API. I started it to pursue my dream about using my spatial imagination for programming and developing video games. It turned out to be very absorbing and I started separating game engine from the game itself, to make adding new features easier. With this project I learned a lot about abstract programming, development and design patterns.

1.1 License

The project is free software released under GPLv2 license.

1.2 Contributing

If you ever happen to use my code please don't be afraid to show your work. I will appreciate all patches, features and improvements.

Chapter 2

Development state

Currently you can move around and push boxes.

2.1 Goals

1. ~~general code cleanup DONE, the poject compiles~~
2. debug loading maps in Map class, currently causing segfaults. valgrind output:

```
==10121==    at 0x409184: void std::vector<game_obj*,
   std::allocator<game_obj*> >::emplace_back<game_obj*>(game_obj*&&)
   (vector.tcc:94)
==10121==    by 0x40882F: std::vector<game_obj*,
   std::allocator<game_obj*> >::push_back(game_obj*&&) (in
   /mnt/Horyzont Zdarzen/PROGRAMOWANIE/SDL i OpenGL/GreenBall/test)
==10121==    by 0x404E4E: Map::load_map(char const*) (Map.cpp:72)
==10121==    by 0x40476A: Map::Map() (Map.cpp:14)
==10121==    by 0x406625: main (game.cpp:472)
```

   New hint: It is probably caused by player not initialized properly due to not being global anymore.

3. rethink the map format and way to mark which Switch opens which Door
4. create Trigger - Responder system for opening doors
5. implement collecting Gems and display score properly

2.2 Known bugs

None yey

2.3 Naming conventions

Yet to be implemented, borrowed from Ogre3D rendering engine.

1. Classes, types and structures must be title case (MyNewClass).
2. Methods and local variables must be camel case (myNewMethod).
3. Member variables should be prefixed with 'm' (mInstanceVar), static member variables should be prefixed 'ms' (msStaticMemberVar). Do not use any other prefixing such as Hungarian notation.
4. Preprocessor macros must be all upper case.
5. Enums should be named in title case, enum values should be all upper case.

2.4 Ultra distant future

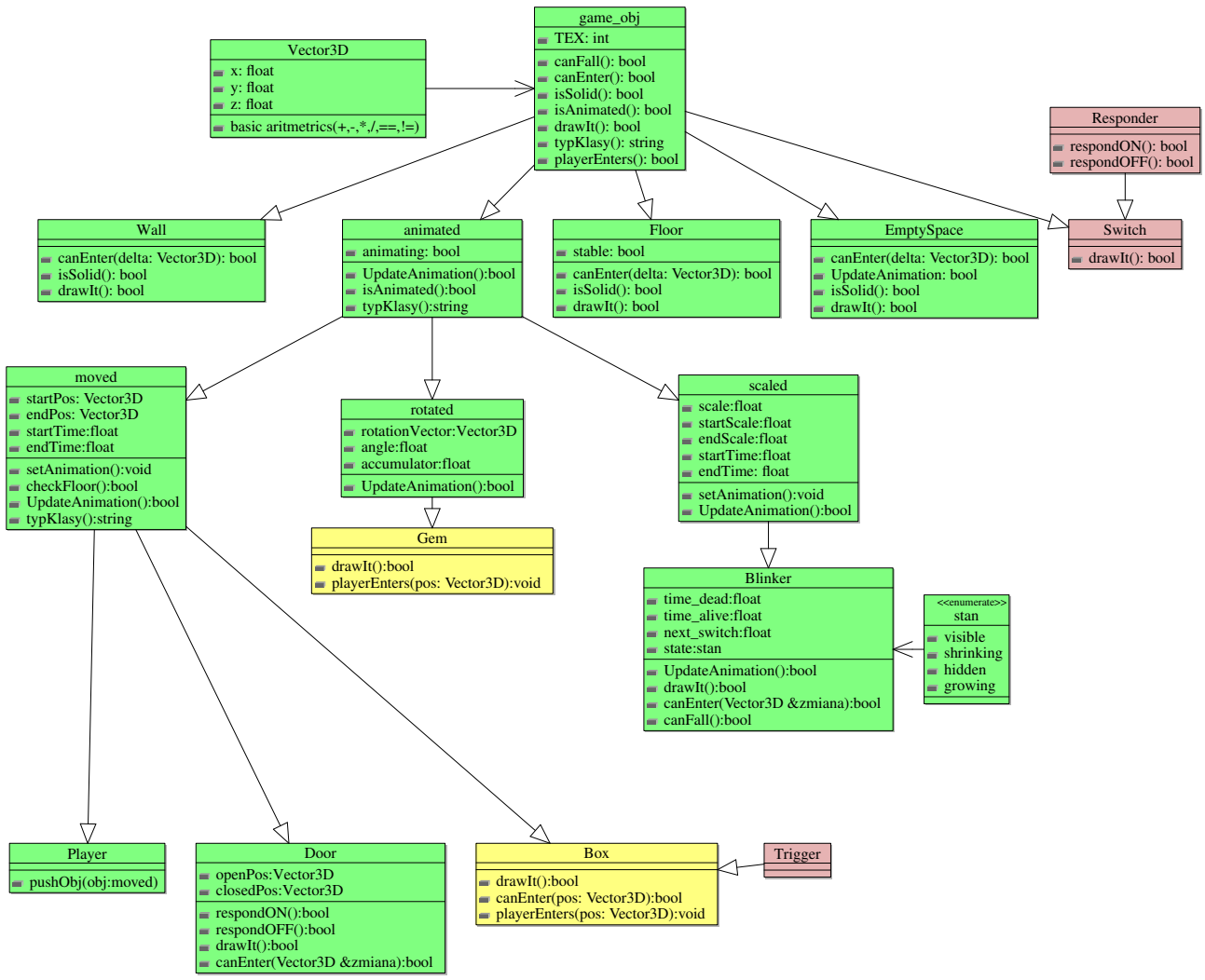
1. Use Doxygen for generating documentation.
2. Use Ogre3D rendering engine.
3. Use Bullet physics engine.

Chapter 3

UML diagrams

Legend:
Green classes are considered ready,
yellow partially done and
red ones lack their main features.

3.1 Game objects inheritance diagram



3.2 Game engine diagram