

1. Record Nr.	UNINA9910955053203321
Autore	Mitchell Shawn Ross
Titolo	SDL game development : discover how to leverage the power of SDL 2.0 to create awesome games in C++ // Shaun Ross Mitchell
Pubbl/distr/stampa	Birmingham, : Packt Pub., 2013
ISBN	9781849696838 1849696837
Edizione	[1st ed.]
Descrizione fisica	1 online resource (256 p.)
Collana	Community experience distilled
Disciplina	ELECTRONIC BOOK
Soggetti	Multimedia systems Video games - Programming
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Intro -- SDL Game Development -- Table of Contents -- SDL Game Development -- Credits -- About the Author -- About the Reviewers -- www.PacktPub.com -- Support files, eBooks, discount offers and more -- Why Subscribe? -- Free Access for Packt account holders -- Preface -- What this book covers -- What you need for this book -- Who this book is for -- Conventions -- Reader feedback -- Customer support -- Downloading the example code -- Errata -- Piracy -- Questions -- 1. Getting Started with SDL -- Why use SDL? -- What is new in SDL 2.0? -- Migrating SDL 1.2 extensions -- Setting up SDL in Visual C++ Express 2010 -- Using Mercurial to get SDL 2.0 on Windows -- Cloning and building the latest SDL 2.0 repository -- I have the library -- now what? -- Hello SDL -- An overview of Hello SDL -- SDL initialization flags -- SDL renderer flags -- What makes up a game -- Breaking up the Hello SDL code -- What does this code do? -- The Game class -- Fullscreen SDL -- Summary -- 2. Drawing in SDL -- Basic SDL drawing -- Getting some images -- Creating an SDL texture -- Source and destination rectangles -- Animating a sprite sheet -- Flipping images -- Installing SDL_image -- Using SDL_image -- Tying it into the framework -- Creating the texture manager -- Using texture manager as a singleton -- Summary -- 3. Working with Game Objects -- Using inheritance -- Implementing polymorphism -- Using abstract base classes -- Should we always use inheritance? -- Could the same

thing be achieved with a simpler solution? -- Derived classes should model the "is a" relationship -- Possible performance penalties -- Putting it all together -- Summary -- 4. Exploring Movement and Input Handling -- Setting up game objects for movement -- What is a vector? -- Some common operations -- Addition of two vectors -- Multiply by a scalar number -- Subtraction of two vectors. Divide by a scalar number -- Normalizing a vector -- Adding the Vector2D class -- Adding velocity -- Adding acceleration -- Creating fixed frames per second -- Input handling -- Creating our input handler class -- Handling joystick/gamepad input -- SDL joystick events -- Initializing joysticks -- Listening for and handling axis movement -- Dealing with joystick button input -- Handling mouse events -- Using mouse button events -- Handling mouse motion events -- Implementing keyboard input -- Wrapping things up -- Summary -- 5. Handling Game States -- A simple way for switching states -- Implementing finite state machines -- A base class for game states -- Implementing FSM -- Implementing menu states -- Function pointers and callback functions -- Implementing the temporary play state -- Pausing the game -- Creating the game over state -- Summary -- 6. Data-driven Design -- Loading XML files -- Basic XML structure -- Implementing Object Factories -- Using Distributed Factories -- Fitting the factory into the framework -- Parsing states from an XML file -- Loading the menu state from an XML file -- Loading other states from an XML file -- Loading the play state -- Loading the pause state -- Loading the game over state -- Summary -- 7. Creating and Displaying Tile Maps -- What is a tile map? -- Getting familiar with the Tiled application -- Parsing and drawing a tile map -- Creating the TileLayer class -- Creating the LevelParser class -- Parsing tilesets -- Parsing a tile layer -- Drawing the map -- Scrolling a tile map -- Parsing object layers -- Developing the ObjectLayer class -- Summary -- 8. Creating Alien Attack -- Using the SDL_mixer extension for sound -- Creating the SoundManager class -- Setting up the basic game objects -- GameObject revamped -- SDLGameObject is now ShooterObject -- Player inherits from ShooterObject -- Lots of enemy types. Adding a scrolling background -- Handling bullets -- Two types of bullets -- The BulletHandler class -- Dealing with collisions -- Creating a CollisionManager class -- Possible improvements -- Summary -- 9. Creating Conan the Caveman -- Setting up the basic game objects -- No more bullets or bullet collisions -- Game objects and map collisions -- ShooterObject is now PlatformerObject -- The Camera class -- Camera-controlled map -- The Player class -- Possible additions -- Summary -- Index.

Sommario/riassunto

Written as a practical and engaging tutorial, SDL Game Development guides you through developing your own framework and the creation of two engaging games. If you know C++ and you're looking to make great games from the ground up, then this book is perfect for you.
