Main + main() + logError(exception : string)

Engine &audioEngine : AudioEngine - &videoEngine : VideoEngine - mLevel : Level - mPlaver : Plaver mlsRunning : bool = false - mMainMenu : bool = true - mSerializer : GameSaveSerializer - mCurrentLevelID : int + EngineState(&audioEngine : AudioEngine, &videoEngine : VideoEngine) + run() - getMouseCoordinates(): Point - getMouseClickState(): int - onPlayAudio(&actor : BaseActor, value : int) onLoadLevel(&actor : BaseActor, value : int) - onChangeScene(&actor : BaseActor, value : int) - handleEvent(&actor : BaseActor, actorEvent : ActorEvent) render() - onExitGame(&actor : BaseActor, value : int) onNewGame(&actor : BaseActor, value : int) - onDisplayText(&actor : BaseActor, value : int) - onLoadGame(&actor : BaseActor, value : int) - onStreamAudio(&actor : BaseActor, value : int)

Engine subordinate audio/video modules in separate chart

GameSaveSerializer

- slot : int

+ load(&levelID : int, &sceneID : int, &inventory : vector<int>)

+ save(&levelID : int. &sceneID : int. &inventory : vector<int>)

+ saveInThread(levelID : int, sceneID : int, inventory : vector<int>)

+ setSlot(slot : int)

- fileLock : mutex

Plaver - mRegion : Region

- mLaver : int

- mTextureID : int - mPosition : Point

- mInventory : vector<int> - mCursorStyle : CursorStyle

+ hasItem(itemType : int) : bool

+ addltem(itemType : int)

+ setCursor(cursorStyle : CursorStyle)

+ setPosition(x: int, y: int)

+ getPosition() : Point

+ getInventory() : vector<int>

+ setInventory(inventory : vector<int>)

+ getCursorTextureID(): int

«enum»

CursorStyle CURSOR NONE CURSOR SELECT

CURSOR LEFT CURSOR RIGHT CURSOR UP CURSOR DOWN

Point + x : int = 0+ y : int = 0

Level scenes : map<int, vector<BaseActor>> - activeScene : int

- levelID : int

+ Level(levelID: int. &audioEngine: AudioEngine. &videoEngine: VideoEngine)

+ getScene(): int

+ changeScene(sceneID : int)

+ clearAll()

+ getActors()

+ getBGImageID(): int

+ loadLevel(path: string, audioEngine: AudioEngine, videoEngine: VideoEngine): int



+ onHover(player : Player) : ActorEvent + step(player&: Player): ActorEvent

+ isInBounds(point : Point) : bool + setRegion(region : Region)

+ getRegion(): Region

+ getLayer(): int

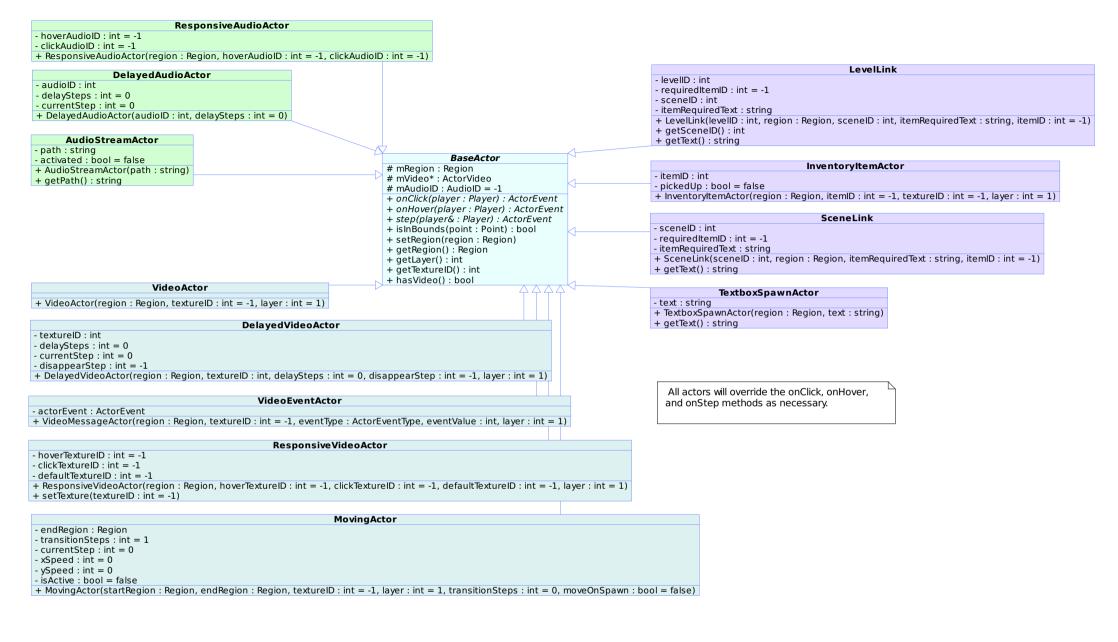
+ getTextureID(): int + hasVideo(): bool

BaseActor inheritance in separate chart

«enum» ActorEventType ChangeScene LoadLevel

PlayAudio Nil NewGame LoadGame DisplayText

ExitGame StreamAudio



Engine + EngineState(&audioEngine : AudioEngine, &videoEngine : VideoEngine) + run()

AudioEngine

- coreResources : vector<AudioID>
- levelResources : vector<AudioID>
- audioPlayer : AudioPlayer
- + loadSound(&path : string, id : AudioID)
- + playSound(id : AudioID)
- + playStream(&path : string)
- + deleteSound(id : AudioID)
- + deleteSoundGroup(resourceGroup : ResourceGroup)

AudioPlayer

- audio_rate : int = 44100 - audio_channels : int = 1
- audio buffer : int = 4096
- *audio_samples : unordered_map<AudioID, Mix_Chunk*>
- *audio_stream : Mix_Music = NULL
- + load stream(&file path : string)
- + stream audio(loops : int = -1)
- + load_sample(id : AudioID, &file_path : string)
- + play sample(id : AudioID)
- + delete sample(id : AudioID)
- + clear_samples()

VideoEngine

- layers[] : TextureID

- coreResources : vector<TextureID>
- levelResources : vector<TextureID>
- videoContext : VideoContext
- textboxActive : bool = false
- textboxRegion : Region
- textboxPadding : Region
- + VideoEngine(&title : string, width : unsigned int, height : unsigned int)
- + loadTexture(&path: string, id: TextureID, resourceGroupName: ResourceGroup)
- + render(®ion : Region, layer : unsigned int, textureID : TextureID)
- + renderRotate(®ion : Region, layer : unsigned int, textureID : TextureID, angle : float = 0.0)
- + isShowingTextbox(): bool
- + showTextbox(&text : string)
- + hideTextbox()
- + deleteTexture(textureID : TextureID)
- + deleteResourceGroup(resourceGroup : ResourceGroup)
- + display()
- + clearLayers()

VideoContext

- *renderer : SDL Renderer
- *font : TTF Font
- *videoDisplay : VideoDisplay
- *textures : unordered map<TextureID, SDL Texture*>
- + VideoContext(&title: string, width: unsigned int, height: unsigned int)
- + display(
- + load texture(id: TextureID, &image path: string, blend: BlendMode = BLENDMODE BLEND): Region
- + load font(&font path : string, font size : int)
- + create texture(id: TextureID, width: int, height: int, blend: BlendMode = BLENDMODE BLEND): Region
- + fill texture(id : TextureID, r : int, g : int, b : int, a : int)
- + render(id : TextureID, *dest : Region, *src : Region)
- + render_onto(dest_id : TextureID, src_id : TextureID, *dest_region : Region, *src_region : Region)
- + render_rotate(dest_id : TextureID, src_id : TextureID, *dest_region : Region, *src_region : TextureID, angle : double = 0.0, flip : Flip = FLIP_NONE)
- + render clear()
- + render_clear(id : TextureID)
- + render_text(dest_id : TextureID, *dest_region : Region, &text : string, r : Uint8, g : Uint8, b : Uint8, a : Uint8)
- + delete_texture(id : TextureID)

VideoDisplay

- *window: SDL_Window
- + VideoDisplay(&title: string, width: unsigned int, height: unsigned int)
- + *get renderer(): SDL Renderer