

MainGUI
<pre> IGUI *(* _make)(void); void(* _destroy)(IGUI *gui); Data *_gameData; IGame *(* _makeGame)(void); void (* _destroyGame)(IGame *); ISound *(* _smake)(void); void(* _sdestroy)(ISound*); ISound *_sound; IGUI *_gui; IGame *_game; int _gameNumber; int _libNumber; std::string getScore(const std::string &amp;); std::vector&lt;std::string&gt; _guiLib; std::vector&lt;std::string&gt; _gameLib; std::map&lt;std::string, std::pair&lt;size_t, size_t&gt;&gt; _thumbnail; std::map&lt;std::string, std::string&gt; _scoreFile; std::map&lt;std::string, std::string&gt; _scoreMax; int _soundMain; int _soundPM; int _soundArrow; int _libIndex; int _gameIndex; int _lastGameScore; int _isWriting; int _i; int _j; int _k; IObject::Coordonnes_t _c; LLoader *_loader; LLoader *_soundLoader; LLoader *_gameLoader; Data *_data; IObject *_bg; IObject *_gt; IObject *_arcadetitle; IObject *_libtitle; IObject *_gametitle; IObject *_score; IObject *_maxScore; IObject *_playerName; IObject *_leftArrow; IObject *_rightArrow; IObject *_currentPlayer;  MainGui(const std::string libName);/ ~MainGui();/ void run(); void loadDL(); void loadThumbnail(); void loadScore(); void startGame(std::string, std::string); void launchGame(); void loadScoreMax(); void writeScore(); void navigateUp(); void navigateDown(); </pre>

Arcade
MainGui *_mainGui
<pre> +Arcade(); +~Arcade() = default; +void start(int ac, char **av) ; +void checkParams(int ac, char **) const; </pre>

LLoader
<pre> std::string _libName; void *_dl; </pre>
<pre> LLoader(); LLoader(const std::string &amp;libName); ~LLoader(); void loadLib(); void *get_sym(const std::string sym); void closeLib(); void reload(const std::string &amp;); std::string getLib() const; </pre>

AException
<pre> -std::string _msg; </pre>
<pre> +AException(const std::string &amp;msg); +virtual ~AException() throw() {} -virtual const char* what() const throw(); </pre>

PFont_t
<pre> +unsigned char r; +unsigned char g; +unsigned char b; +unsigned char a; +int size; +std::string font; </pre>

IObject
<pre> Data *_data; PFont_t _font; Coordonnes_t _coord; Coordonnes_t _togo; Coordonnes_t _imgCoord; std::string _termpic; Coordonnes_t _size; </pre>
<pre> virtual ~IObject() {}; virtual void update() = 0; virtual Coordonnes_t getCoord() const = 0; virtual Coordonnes_t getTogo() const = 0; virtual Coordonnes_t getCoordImg() const = 0; virtual std::string getTermpic() const = 0; virtual Coordonnes_t getSize() const = 0; virtual PFont_t getFont() const = 0; virtual void setFont(PFont_t font) = 0; virtual void setCoord(const Coordonnes_t &amp;coord) = 0; virtual void place(const int &amp;x, const int &amp;y) = 0; virtual void go(const Coordonnes_t &amp;coord) = 0; virtual void setCoordImg(const Coordonnes_t &amp;coord) = 0; virtual void setTermpic(const std::string &amp;term) = 0; virtual void setSize(const Coordonnes_t &amp;t) = 0; </pre>



AObject
<pre> bool _solid; </pre>
<pre> AObject(); AObject(const Coordonnes_t &amp;c); AObject(const Coordonnes_t &amp;co, const Coordonnes_t &amp;ci, const std::string &amp;c); virtual ~AObject() {}; virtual void update() = 0; Coordonnes_t getCoord() const; Coordonnes_t getCoordImg() const; std::string getTermpic() const; Coordonnes_t getSize() const; Coordonnes_t getTogo() const; PFont_t getFont() const; void setFont(PFont_t font); void place (const int &amp;x, const int &amp;y); void go(const Coordonnes_t &amp;coord); void setCoord(const Coordonnes_t &amp;coord); void setCoordImg(const Coordonnes_t &amp;coord); void setTermpic(const std::string &amp;term); void setSize(const Coordonnes_t &amp;t); bool isSolid() const; void setSolid(const bool &amp;solid); </pre>

```
void loadScoreMax();
void writeScore();
void navigateUp();
void navigateDown();
void navigateRight();
void navigateLeft();
void blitAndUpdate();
void switchLibLeft();
void switchLibRight();
void initRun();
void writeName();
void switchGameLeft();
void switchGameRight();
void reloadGame();
```

