Adding new lib:

New lib should implement the function

IDisplayModule *entryPoint(Arcade *arcade)

It should initialize the new lib and return it.

Graphical Lib API

Graphical lib should implement:

void loadElemToDisplay(std::vector<IDisplayElem*> &elements)

This function is called when loading new assets. Old assets should always be cleared when called.

void getMoved(std::vector<IMovedElem*> &elements)

This function is called when moving assets.

void getInput(void)

This function is called for getting input. Should call giveInput of Arcade for each input.

void display (void)

This function is called for updating display.

Graphical lib can use from arcade class:

void giveInput(char input)

This function should be called for each input.

Game Lib API

Game lib should implement:

std::vector<IDisplayElem*> *getGraphElem(void)

This function should return all the assets that the game need.

std::vector<IMovedElem*> *giveMoved(void)

This function should return moved objects during the loop

void getInput(char input)

This function is called for input.

void gameTick(void)

This function is called at each game turn.

Class List

Position_t {

unsigned int x.

unsigned int y.

}

IDisplayElem {

int getId(void) const

This function return the id of the asset. Id should be unique

bool getVisible(void) const

This function return the display. If false, asset should not be drawn

position_t getPosition(void) const

std::string getFilePath(void) const

This function should return the file to the asset for graphical lib

char getCharDisp(void) const

This function should return the character for terminal lib

int chgVisible (bool visible)

int chgPosition(position_t &newPos)

}

IMovedElem {

int getId(void) const

This function return the id of the concerned asset.

position_t getPosition(void) const

bool getVisible(void) const

}