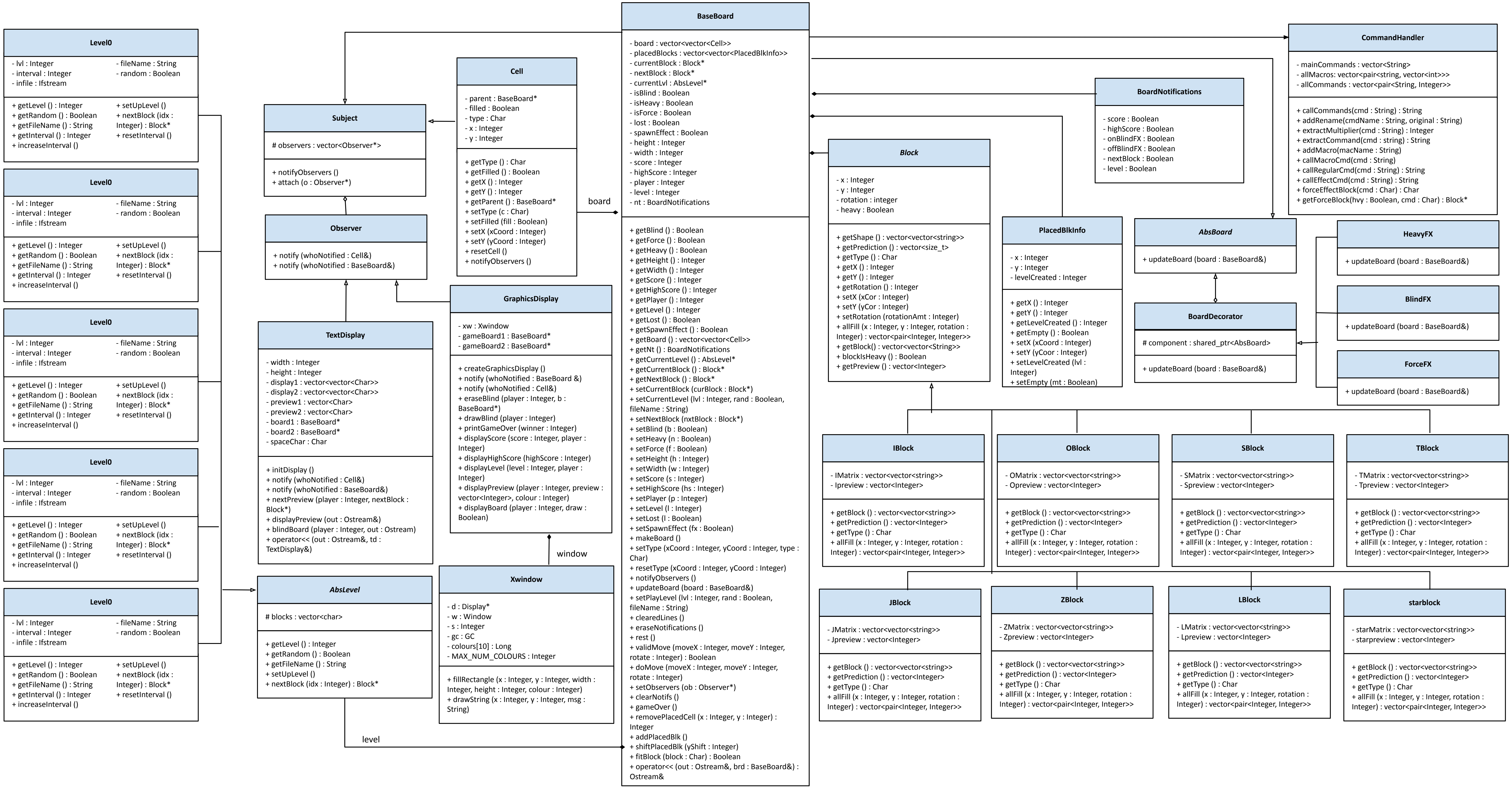
Biquadris Project UML Diagram



Level0

- lvl : Integer
- interval : Integer
- infile : Ifstream
- fileName : String
- random : Boolean

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ getInterval () : Integer
+ increaseInterval ()
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*
+ resetInterval ()

Level0

- lvl : Integer
- interval : Integer
- infile : Ifstream
- fileName : String
- random : Boolean

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ getInterval () : Integer
+ increaseInterval ()
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*
+ resetInterval ()

Level0

- lvl : Integer
- interval : Integer
- infile : Ifstream
- fileName : String
- random : Boolean

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ getInterval () : Integer
+ increaseInterval ()
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*
+ resetInterval ()

Level0

- lvl : Integer
- interval : Integer
- infile : Ifstream
- fileName : String
- random : Boolean

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ getInterval () : Integer
+ increaseInterval ()
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*
+ resetInterval ()

Level0

- lvl : Integer
- interval : Integer
- infile : Ifstream
- fileName : String
- random : Boolean

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ getInterval () : Integer
+ increaseInterval ()
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*
+ resetInterval ()

Subject

# observers : vector<Observer*>

+ notifyObservers ()
+ attach (o : Observer*)

Observer

+ notify (whoNotified : Cell&)
+ notify (whoNotified : BaseBoard&)

Cell

- parent : BaseBoard*
- filled : Boolean
- type : Char
- x : Integer
- y : Integer

+ getType () : Char
+ getFilled () : Boolean
+ getX () : Integer
+ getY () : Integer
+ getParent () : BaseBoard*
+ setType (c : Char)
+ setFilled (fill : Boolean)
+ setX (xCoord : Integer)
+ setY (yCoord : Integer)
+ resetCell ()
+ notifyObservers ()

TextDisplay

- width : Integer
- height : Integer
- display1 : vector<vector<Char>>
- display2 : vector<vector<Char>>
- preview1 : vector<Char>
- preview2 : vector<Char>
- board1 : BaseBoard*
- board2 : BaseBoard*
- spaceChar : Char

+ initDisplay ()
+ notify (whoNotified : Cell&)
+ notify (whoNotified : BaseBoard&)
+ nextPreview (player : Integer, nextBlock : Block*)
+ displayPreview (out : Ostream&)
+ blindBoard (player : Integer, out : Ostream)
+ operator<< (out : Ostream&, td : TextDisplay&)

GraphicsDisplay

- xw : Xwindow
- gameBoard1 : BaseBoard*
- gameBoard2 : BaseBoard*

+ createGraphicsDisplay ()
+ notify (whoNotified : BaseBoard &)
+ notify (whoNotified : Cell&)
+ eraseBlind (player : Integer, b : BaseBoard*)
+ drawBlind (player : Integer)
+ printGameOver (winner : Integer)
+ displayScore (score : Integer, player : Integer)
+ displayHighScore (highScore : Integer)
+ displayLevel (level : Integer, player : Integer)
+ displayPreview (player : Integer, preview : vector<Integer>, colour : Integer)
+ displayBoard (player : Integer, draw : Boolean)

AbsLevel

# blocks : vector<char>

+ getLevel () : Integer
+ getRandom () : Boolean
+ getFileName () : String
+ setUpLevel ()
+ nextBlock (idx : Integer) : Block*

Xwindow

- d : Display*
- w : Window
- s : Integer
- gc : GC
- colours[10] : Long
- MAX_NUM_COLOURS : Integer

+ fillRectangle (x : Integer, y : Integer, width : Integer, height : Integer, colour : Integer)
+ drawString (x : Integer, y : Integer, msg : String)

BaseBoard

- board : vector<vector<Cell>>
- placedBlocks : vector<vector<PlacedBlkInfo>>
- currentBlock : Block*
- nextBlock : Block*
- currentLvl : AbsLevel*
- isBlind : Boolean
- isHeavy : Boolean
- isForce : Boolean
- lost : Boolean
- spawnEffect : Boolean
- height : Integer
- width : Integer
- score : Integer
- highScore : Integer
- player : Integer
- level : Integer
- nt : BoardNotifications

+ getBlind () : Boolean
+ getForce () : Boolean
+ getHeavy () : Boolean
+ getHeight () : Integer
+ getWidth () : Integer
+ getScore () : Integer
+ getHighScore () : Integer
+ getPlayer () : Integer
+ getLevel () : Integer
+ getLost () : Boolean
+ getSpawnEffect () : Boolean
+ getBoard () : vector<vector<Cell>>
+ getNt () : BoardNotifications
+ getCurrentLevel () : AbsLevel*
+ getCurrentBlock () : Block*
+ getNextBlock () : Block*
+ setCurrentBlock (curBlock : Block*)
+ setCurrentLevel (lvl : Integer, rand : Boolean, fileName : String)
+ setNextBlock (nxtBlock : Block*)
+ setBlind (b : Boolean)
+ setHeavy (n : Boolean)
+ setForce (f : Boolean)
+ setHeight (h : Integer)
+ setWidth (w : Integer)
+ setScore (s : Integer)
+ setHighScore (hs : Integer)
+ setPlayer (p : Integer)
+ setLevel (l : Integer)
+ setLost (l : Boolean)
+ setSpawnEffect (fx : Boolean)
+ makeBoard ()
+ setType (xCoord : Integer, yCoord : Integer, type : Char)
+ resetType (xCoord : Integer, yCoord : Integer)
+ notifyObservers ()
+ updateBoard (board : BaseBoard&)
+ setPlayLevel (lvl : Integer, rand : Boolean, fileName : String)
+ clearedLines ()
+ eraseNotifications ()
+ rest ()
+ validMove (moveX : Integer, moveY : Integer, rotate : Integer) : Boolean
+ doMove (moveX : Integer, moveY : Integer, rotate : Integer)
+ setObservers (ob : Observer*)
+ clearNotifs ()
+ gameOver ()
+ removePlacedCell (x : Integer, y : Integer) : Integer
+ addPlacedBlk ()
+ shiftPlacedBlk (yShift : Integer)
+ fitBlock (block : Char) : Boolean
+ operator<< (out : Ostream&, brd : BaseBoard&) : Ostream&

CommandHandler

- mainCommands : vector<String>
- allMacros: vector<pair<string, vector<int>>>
- allCommands : vector<pair<String, Integer>>

+ callCommands(cmd : String) : String
+ addRename(cmdName : String, original : String)
+ extractMultiplier(cmd : String) : Integer
+ extractCommand(cmd : string) : String
+ addMacro(macName : String)
+ callMacroCmd(cmd : String)
+ callRegularCmd(cmd : String) : String
+ callEffectCmd(cmd : String) : String
+ forceEffectBlock(cmd : Char) : Char
+ getForceBlock(hvy : Boolean, cmd : Char) : Block*

BoardNotifications

- score : Boolean
- highScore : Boolean
- onBlindFX : Boolean
- offBlindFX : Boolean
- nextBlock : Boolean
- level : Boolean

Block

- x : Integer
- y : Integer
- rotation : integer
- heavy : Boolean

+ getShape () : vector<vector<string>>
+ getPrediction () : vector<size_t>
+ getType () : Char
+ getX () : Integer
+ getY () : Integer
+ getRotation () : Integer
+ setX (xCor : Integer)
+ setY (yCor : Integer)
+ setRotation (rotationAmt : Integer)
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>
+ getBlock() : vector<vector<String>>
+ blockIsHeavy () : Boolean
+ getPreview () : vector<Integer>

PlacedBlkInfo

- x : Integer
- y : Integer
- levelCreated : Integer

+ getX () : Integer
+ getY () : Integer
+ getLevelCreated () : Integer
+ getEmpty () : Boolean
+ setX (xCoord : Integer)
+ setY (yCoor : Integer)
+ setLevelCreated (lvl : Integer)
+ setEmpty (mt : Boolean)

AbsBoard

+ updateBoard (board : BaseBoard&)

BoardDecorator

# component : shared_ptr<AbsBoard>

+ updateBoard (board : BaseBoard&)

HeavyFX

+ updateBoard (board : BaseBoard&)

BlindFX

+ updateBoard (board : BaseBoard&)

ForceFX

+ updateBoard (board : BaseBoard&)

IBlock

- IMatrix : vector<vector<string>>
- Ipreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

OBlock

- OMatrix : vector<vector<string>>
- Opreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

SBlock

- SMatrix : vector<vector<string>>
- Spreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

TBlock

- TMatrix : vector<vector<string>>
- Tpreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

JBlock

- JMatrix : vector<vector<string>>
- Jpreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

ZBlock

- ZMatrix : vector<vector<string>>
- Zpreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

LBlock

- LMatrix : vector<vector<string>>
- Lpreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>

starblock

- starMatrix : vector<vector<string>>
- starpreview : vector<Integer>

+ getBlock () : vector<vector<string>>
+ getPrediction () : vector<Integer>
+ getType () : Char
+ allFill (x : Integer, y : Integer, rotation : Integer) : vector<pair<Integer, Integer>>