{sf::Drawable ll}

{BlockHandler
ll+ ~BlockHandler()
+ BlockHandler(const int id, const int xOff, const int yOff, TileHandler *tileHandlerIn)
+ void generateBlock()
+ void loadBlocks()
+ void userMove(string const &direction)
+ void draw(sf::RenderTarget &window, sf::RenderStates states) const
+ void update()
+ void transferBlocks (Block *block)
+ void deleteBlock(Block *block)
+ vector< Tile * > getCurrentBlockTiles()
+ void setPongBlock(PongBlock *pongBlock)
+ void hasLost(int id) const
}

#blockHandler

{TileHandler
ll+ ~TileHandler()
+ TileHandler(const int id, const float xOffset, const float yOffset)
+ void update()
+ void move(string moveInput)
+ void draw(sf::RenderTarget &window, sf::RenderStates states) const
+ vector< Tile * > getTiles()
+ void appendTile(Tile *tile)
+ void clearRow()
+ int getPointCount()
+ void setPointsCleared (int value)
+ BlockHandler * getBlockHandler () const
}

{vector< Tile * > ll}

+tiles

#tileHandler

{Block
ll+ ~Block()=default
+ Block()
+ Block(vector< Tile * > const &tilesIn, TileHandler *tileHandlerIn)
+ virtual bool willLand () const
+ bool hasOverlap()
+ virtual void move(string const &direction)
# virtual bool willCollideWithTile(string const &moveInput) const
# bool willCollideWithWall (string const &moveInput) const
}