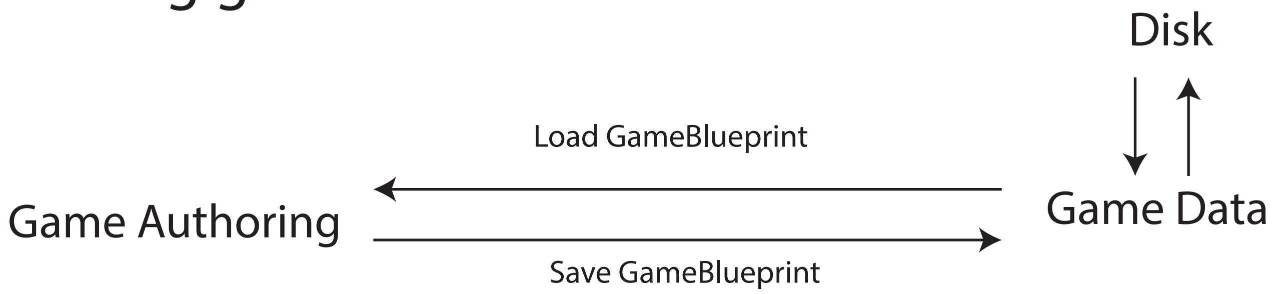
Authoring game



Disk

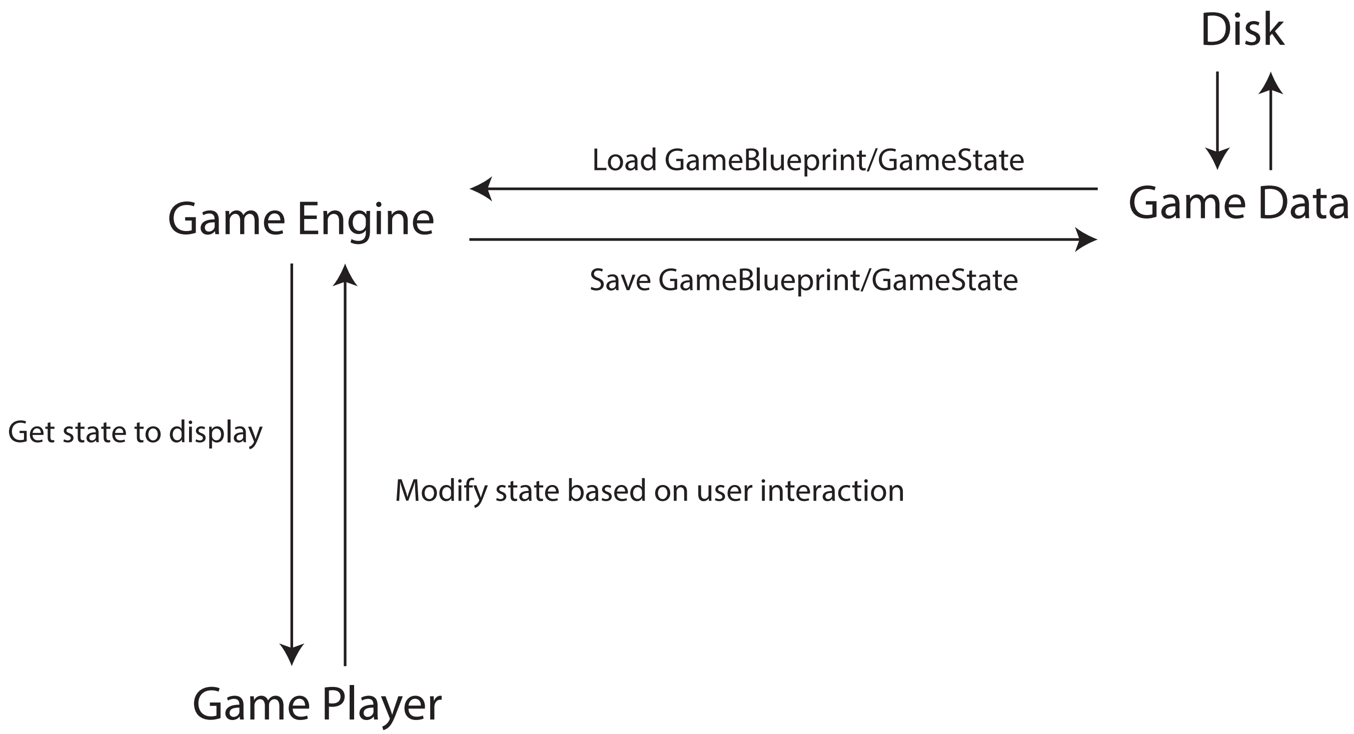
Load GameBlueprint

Game Authoring

Game Data

Save GameBlueprint

Playing game



Disk

Load GameBlueprint/GameState

Game Engine

Game Data

Save GameBlueprint/GameState

Get state to display

Modify state based on user interaction

Game Player

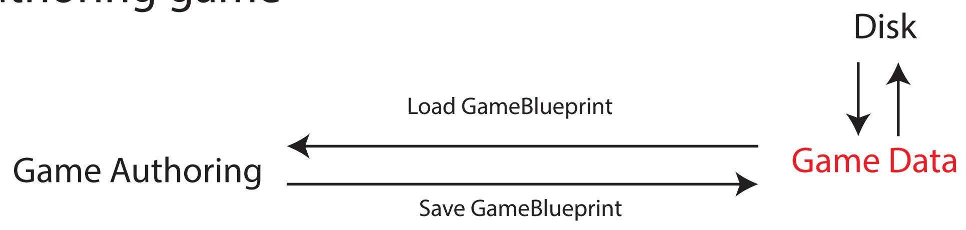
GameBlueprint: container for Schemas

Schema is a design for different object types
Common to both author and engine
Map-based = extensible (adding features does not change API)

- GameSchema
- TowerSchema
- MonsterSchema
- WaveSpawnSchema

GameState is engine-only
List of built towers, score, etc

Authoring game



Disk

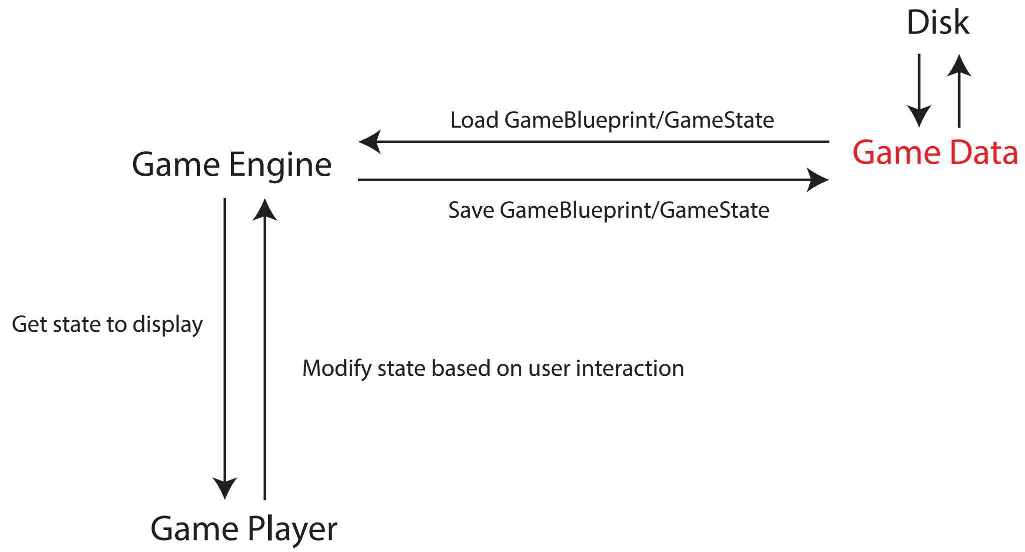
Load GameBlueprint

Game Authoring

Game Data

Save GameBlueprint

Playing game



Disk

Load GameBlueprint/GameState

Game Engine

Game Data

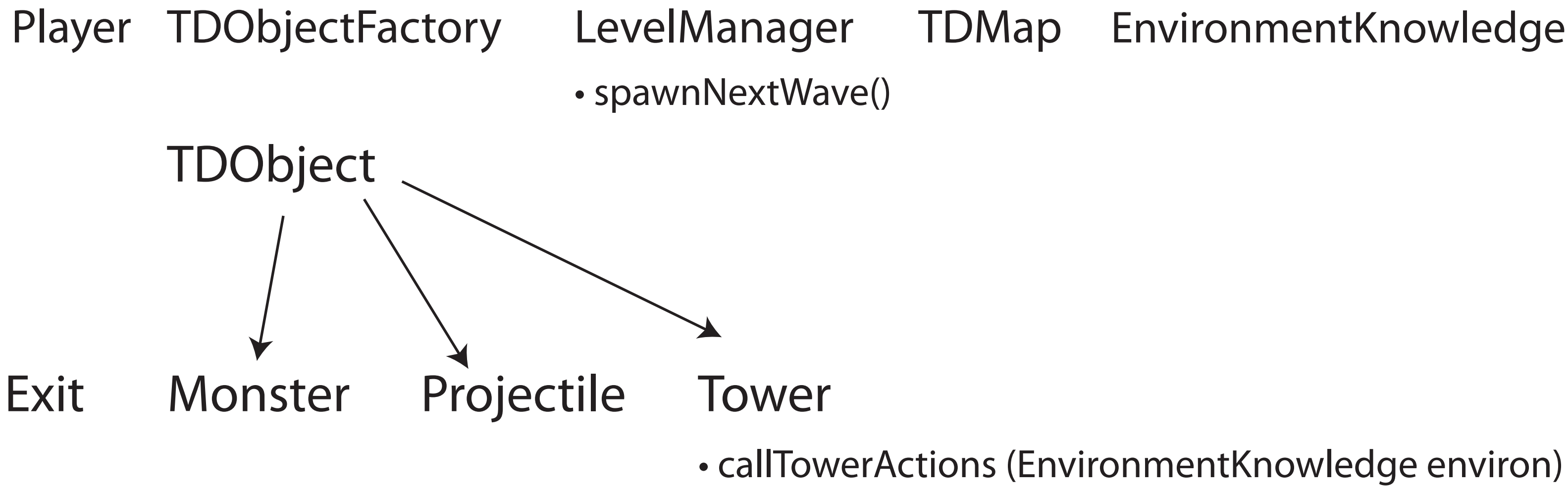
Save GameBlueprint/GameState

Get state to display

Modify state based on user interaction

Game Player

Model



Player

TDObjectFactory

LevelManager

- spawnNextWave()

TDMap

EnvironmentKnowledge

TDObject

Exit

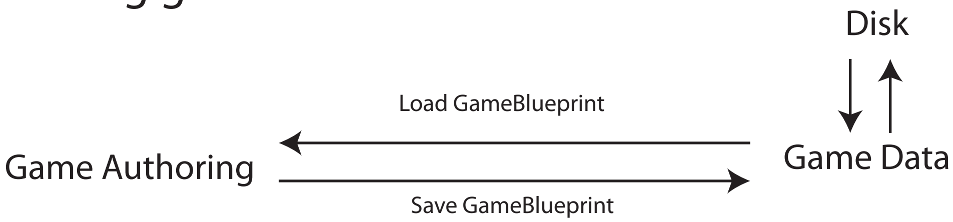
Monster

Projectile

Tower

- callTowerActions (EnvironmentKnowledge environ)

Authoring game



Disk

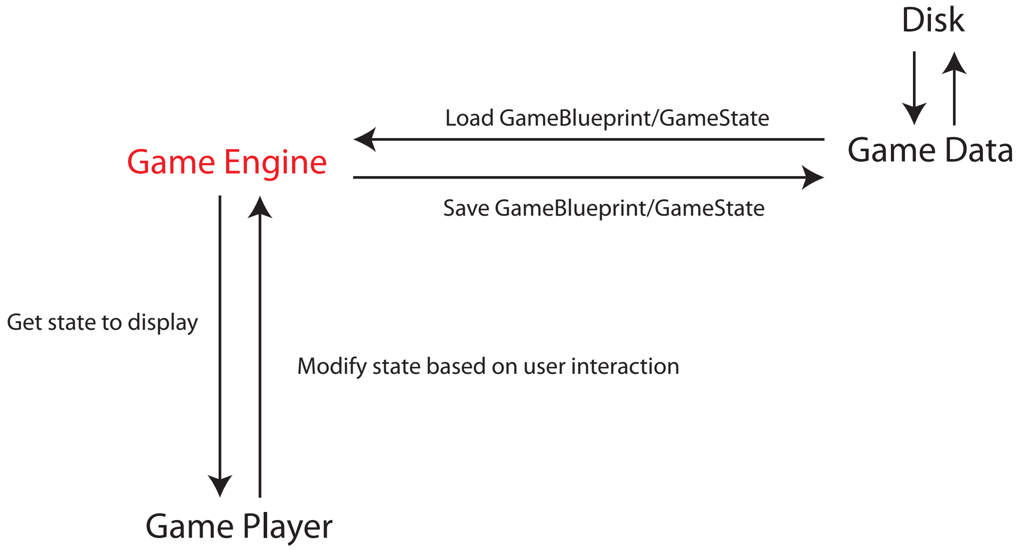
Load GameBlueprint

Game Authoring

Game Data

Save GameBlueprint

Playing game



Disk

Load GameBlueprint/GameState

Game Engine

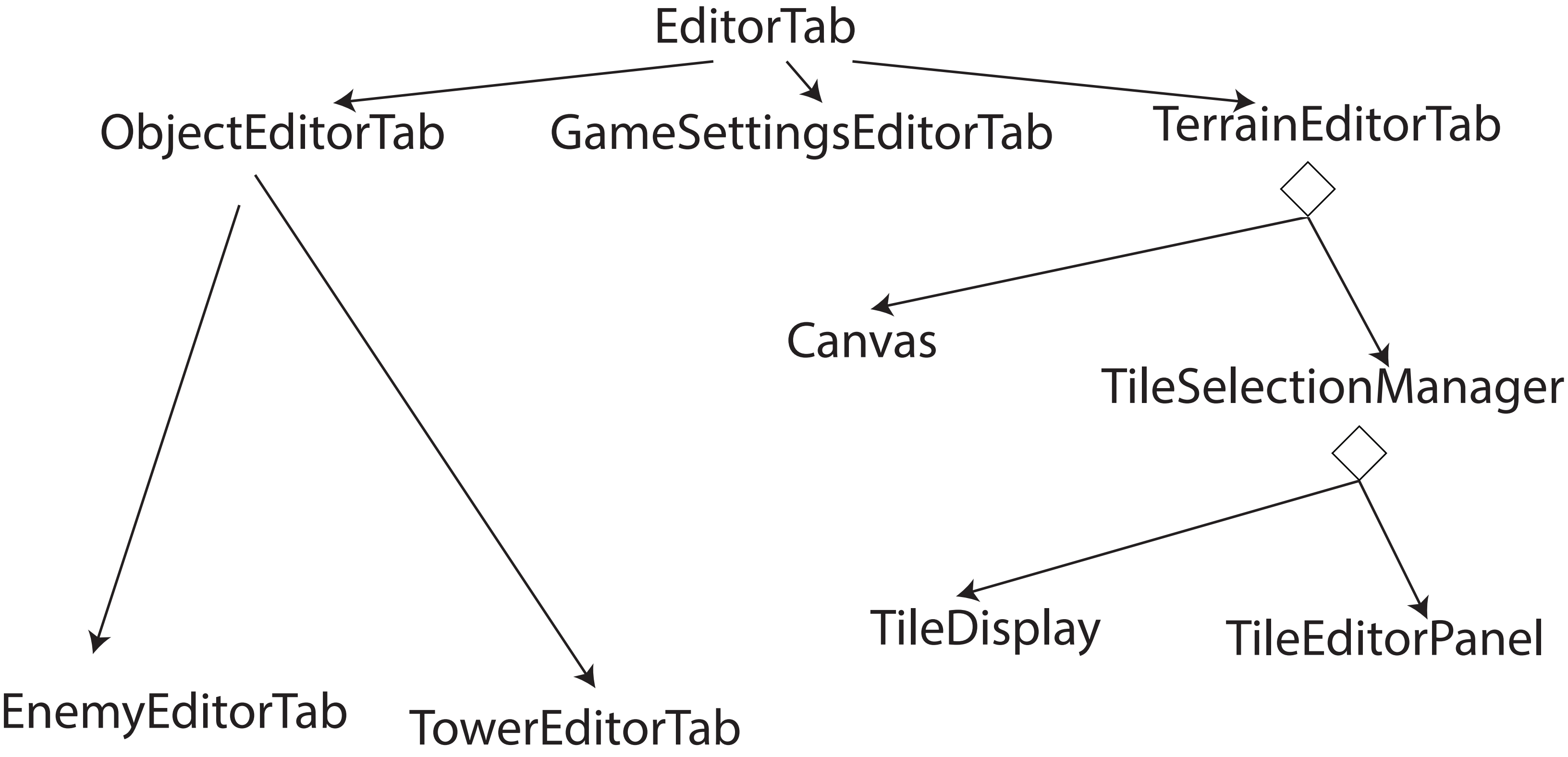
Game Data

Save GameBlueprint/GameState

Get state to display

Modify state based on user interaction

Game Player



EditorTab

ObjectEditorTab

GameSettingsEditorTab

TerrainEditorTab

Canvas

TileSelectionManager

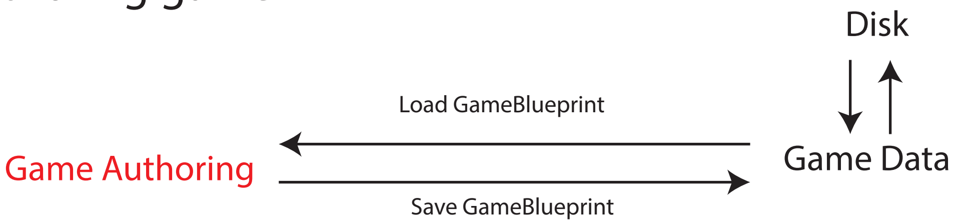
TileDisplay

TileEditorPanel

EnemyEditorTab

TowerEditorTab

Authoring game



Disk

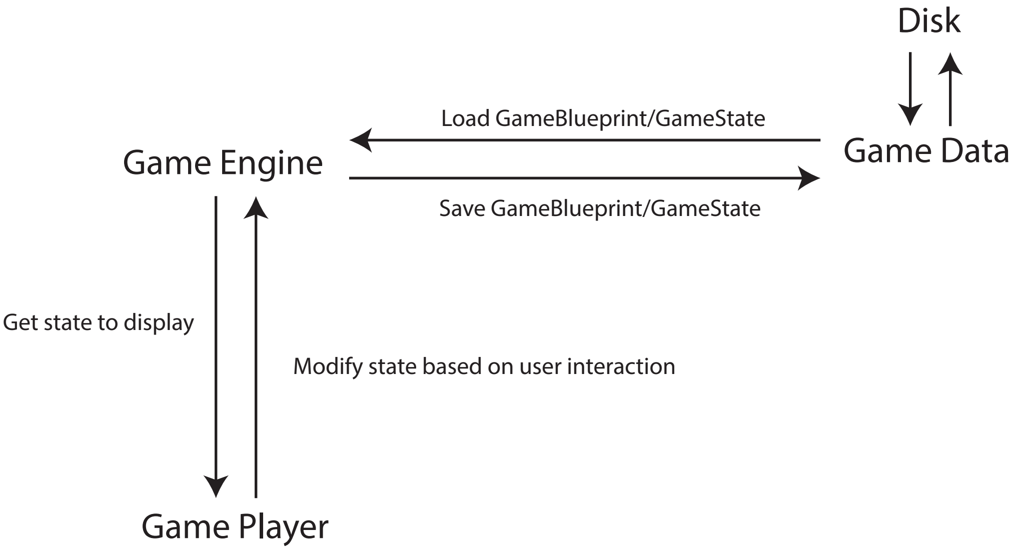
Load GameBlueprint

Game Authoring

Game Data

Save GameBlueprint

Playing game



Disk

Load GameBlueprint/GameState

Game Engine

Game Data

Save GameBlueprint/GameState

Get state to display

Modify state based on user interaction

Game Player

Player (JFrame)

GameInfoPanel (Observing)

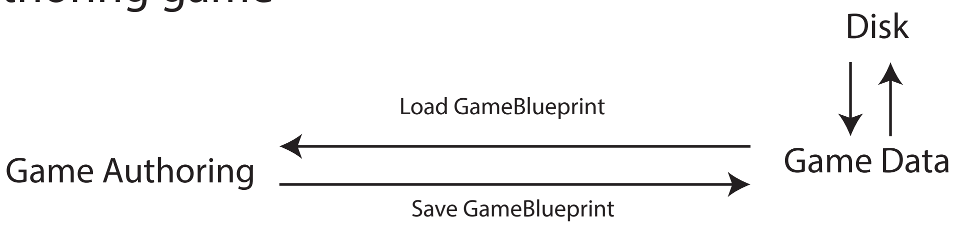
RepositoryViewer

UnitInfoPanel (Observing)

TDPlayerEngine (JGEngine, Subject)

We use the Observer design pattern

Authoring game



Disk

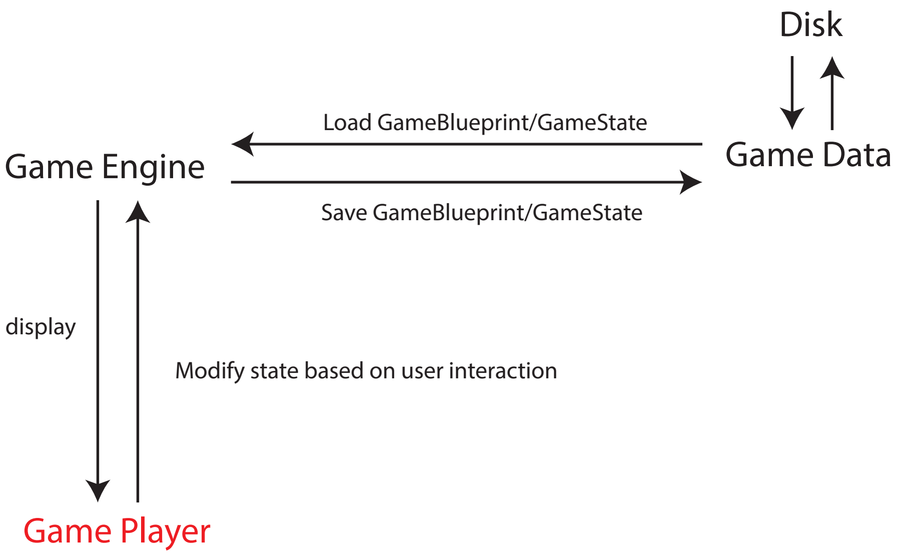
Load GameBlueprint

Game Authoring

Game Data

Save GameBlueprint

Playing game



Disk

Load GameBlueprint/GameState

Game Engine

Game Data

Save GameBlueprint/GameState

Get state to display

Modify state based on user interaction

Game Player

```java
//Engine API
public boolean placeTower(double x, double y)

public boolean isTowerPresent(double x, double y)

public void checkAndRemoveTower(int x, int y)//should be changed to double?

public void loadGameBlueprint(String filePath)//

public doSpawnActivity()

public void updateGame()//Should be called by doFrame() in TDPLayerEngine

public boolean upgradeTower(double x, double y)

public void checkCollisions

public double getScore()

public boolean isGameLost()

public double getGameClock()

public int getPlayerLives()

public int getMoney()

//Game data API
public boolean saveState(GameState currentGameState, String filePath) throws IOException //Returns whether the object was successfully saved

public GameState loadState(String filePath) throws ClassNotFoundException, IOException

public boolean saveBlueprint(GameBlueprint blueprint, String filePath) //Returns whether the object was successfully saved

public GameBlueprint loadBlueprint(String filePath) throws ClassNotFoundException, IOException
```