GP2Engine - Level Editor Guide
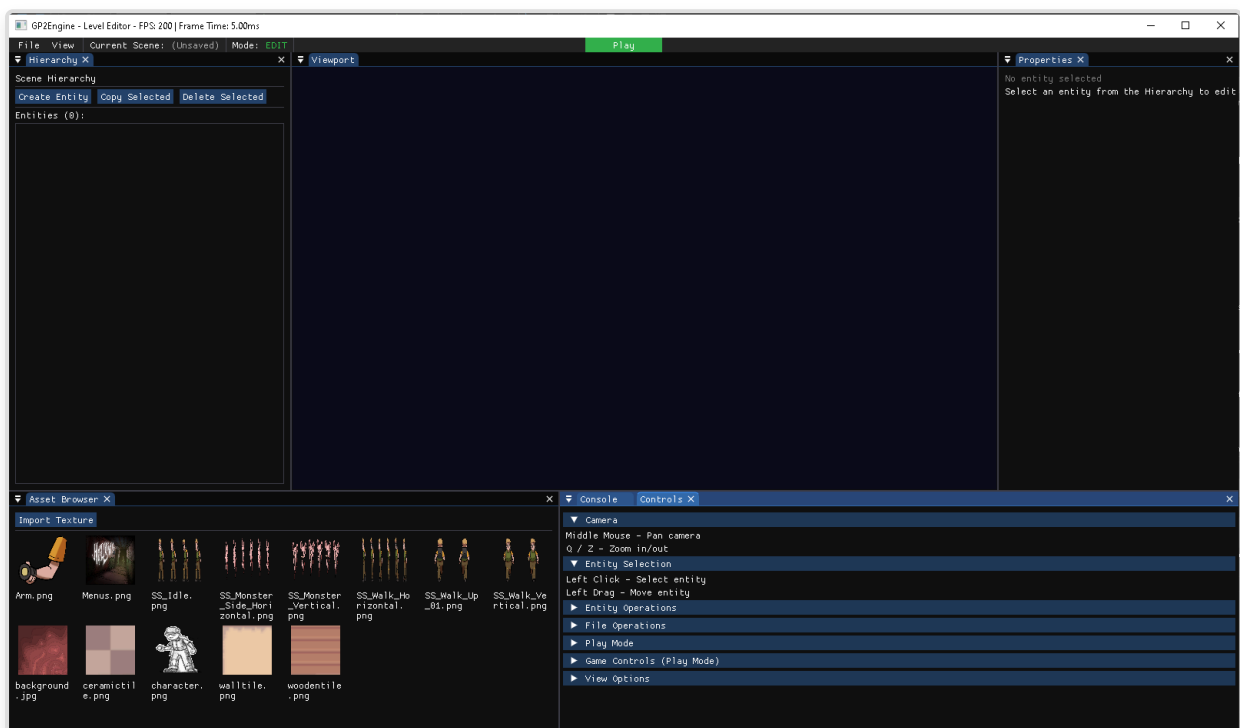
1. Editor Overview

Figure 1: The GP2Engine Level Editor main interface

The editor is divided into six main panels:

Hierarchy (Left)

Lists all entities in the scene with Create, Copy, and Delete buttons.

Viewport (Center)

Visual scene canvas for entity manipulation.

Properties (Right)

Component editor for selected entities.

Asset Browser (Bottom-Left)

Texture and font browser with thumbnails.

Console (Bottom-Right)

Controls (Bottom-Right)

| Displays log messages and errors. | Keyboard and mouse shortcuts reference. |

The menu bar shows the current scene name, editor mode (EDIT/PLAY), and the Play/Stop button.
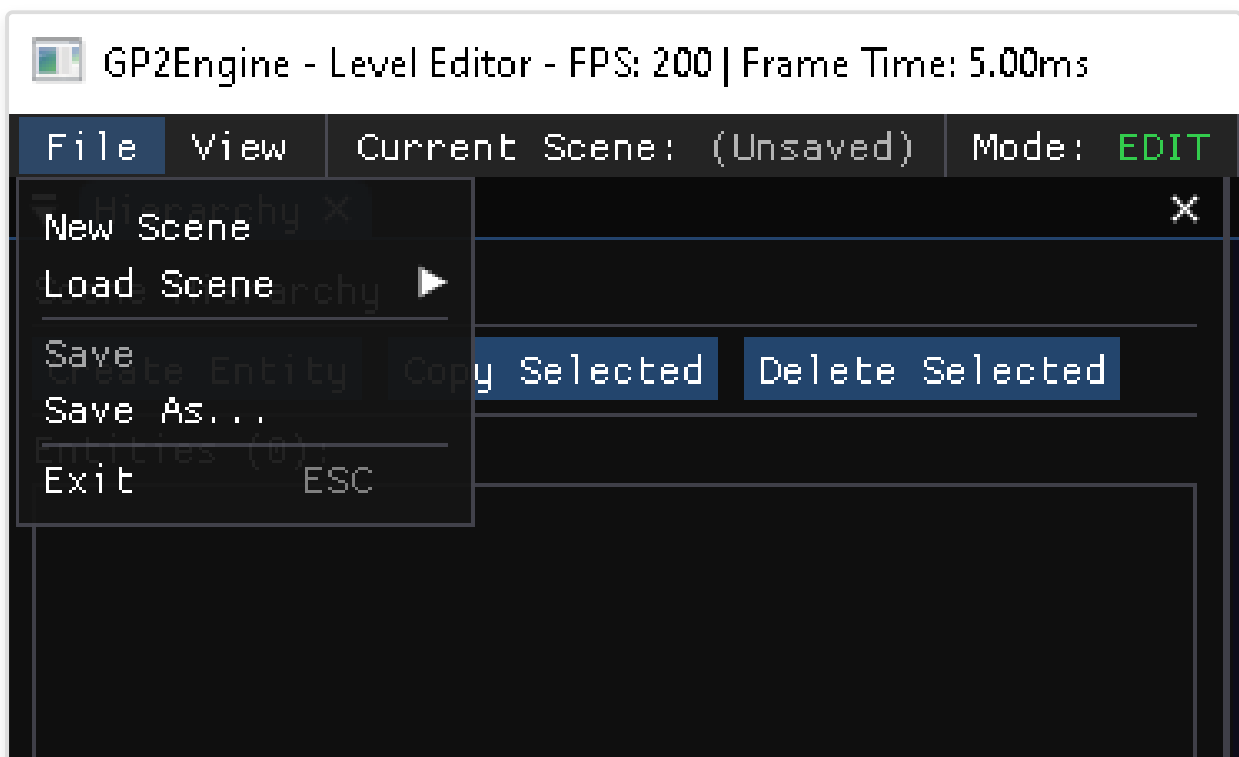
2. File Menu

Figure 2: File menu for scene management

- **New Scene:** Creates empty scene. Prompts to save if unsaved changes exist.

- **Load Scene:** Opens submenu listing all scenes in `assets/scenes/` directory.

- **Save:** Saves current scene to existing file.

- **Save As...:** Opens dialog to save scene with new filename.

- **Exit (ESC):** Closes editor with unsaved changes prompt.

Note: All scenes are saved as JSON files in
`../../Sandbox/assets/scenes/`. The `.json` extension is added
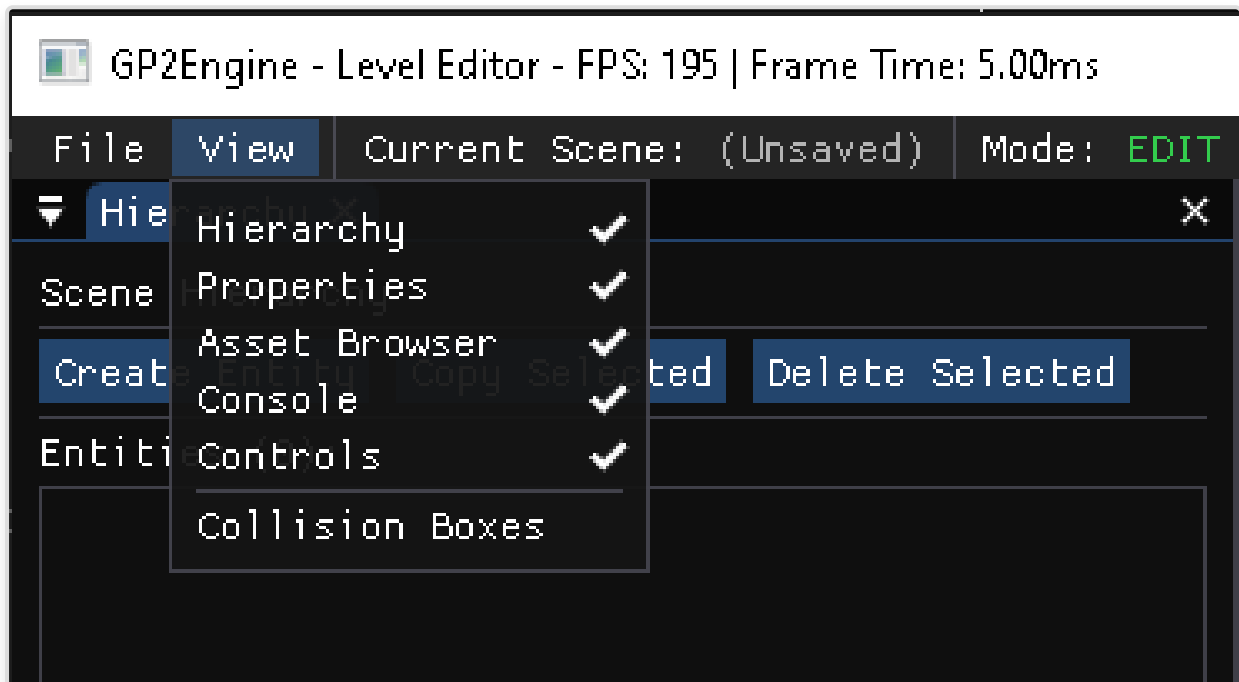
automatically.

3. View Menu

Figure 3: View menu for toggling panel visibility

Toggle panel visibility to customize your workspace:

- **Hierarchy ✓:** Entity list panel

- **Properties ✓:** Component editor panel

- **Asset Browser ✓:** Texture and font browser

- **Console ✓:** Log message panel

- **Controls ✓:** Keyboard shortcuts reference

- **Collision Boxes:** Debug visualization of entity bounding boxes

> **Tip:** Checkmarks (✓) indicate currently visible panels. Hide panels to maximize viewport space.
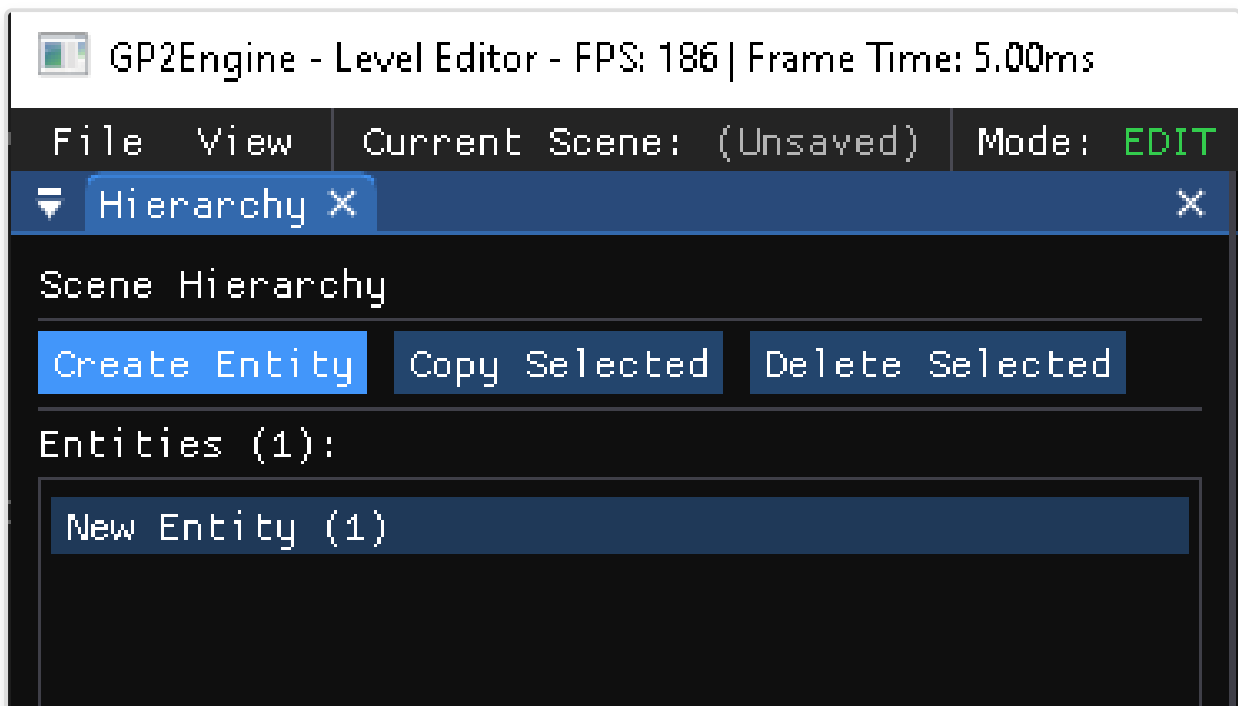
4. Creating Entities

Figure 4: New entity created in Hierarchy panel

To create a new entity:

1. Click **Create Entity** button in Hierarchy panel

2. A new entity named "New Entity (1)" appears in the list

3. The entity is automatically selected

4. Properties panel shows the entity's components

Every new entity includes Transform2D and Tag components by default.
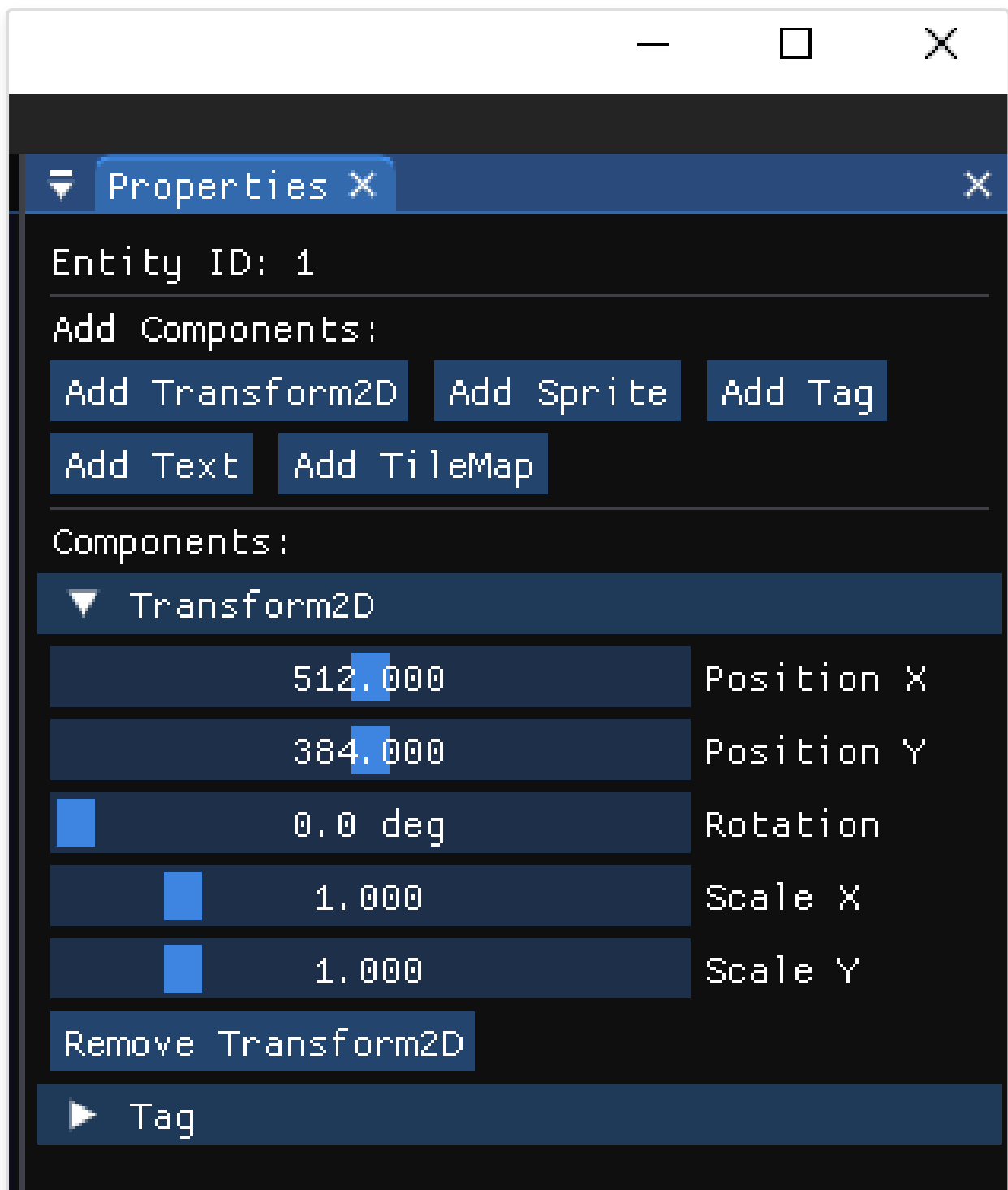
5. Transform2D Component

Figure 5: Transform2D component in Properties panel

The Transform2D component controls entity position, rotation, and scale:

- **Position X/Y:** Location in scene space (default: 512, 384)
- **Rotation:** Rotation angle in degrees (default: 0.0)

- **Scale X/Y:** Size multiplier (default: 1.0 = normal size)

Click slider values to type exact numbers, or drag sliders for visual adjustment.
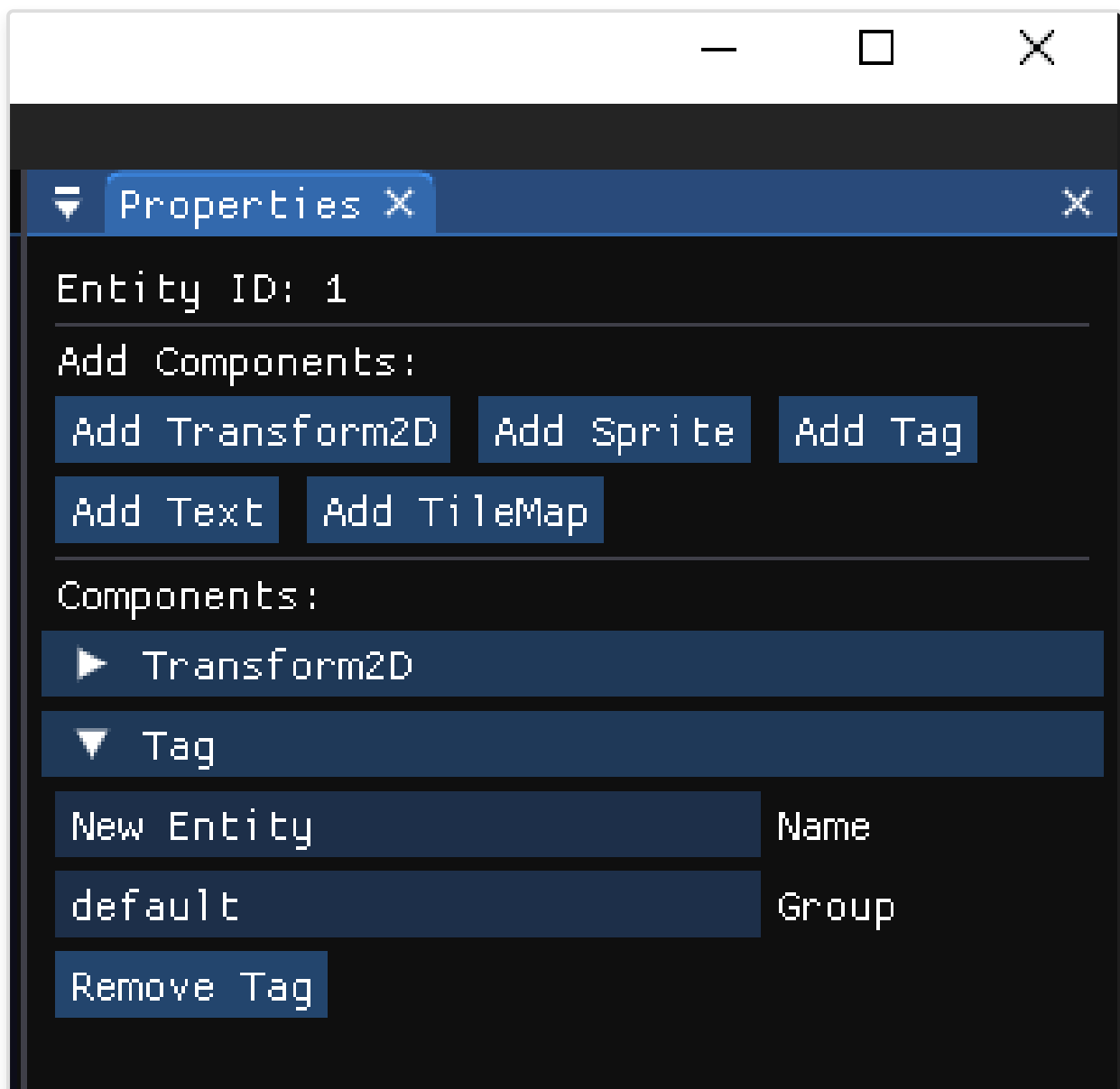
6. Tag Component

Figure 6: Tag component for entity identification

The Tag component provides entity identification:

- **Name:** Entity identifier (e.g., "Player", "Enemy_1", "Wall_Top")

- **Group:** Category for organizing entities (e.g., "enemies", "obstacles", "ui")
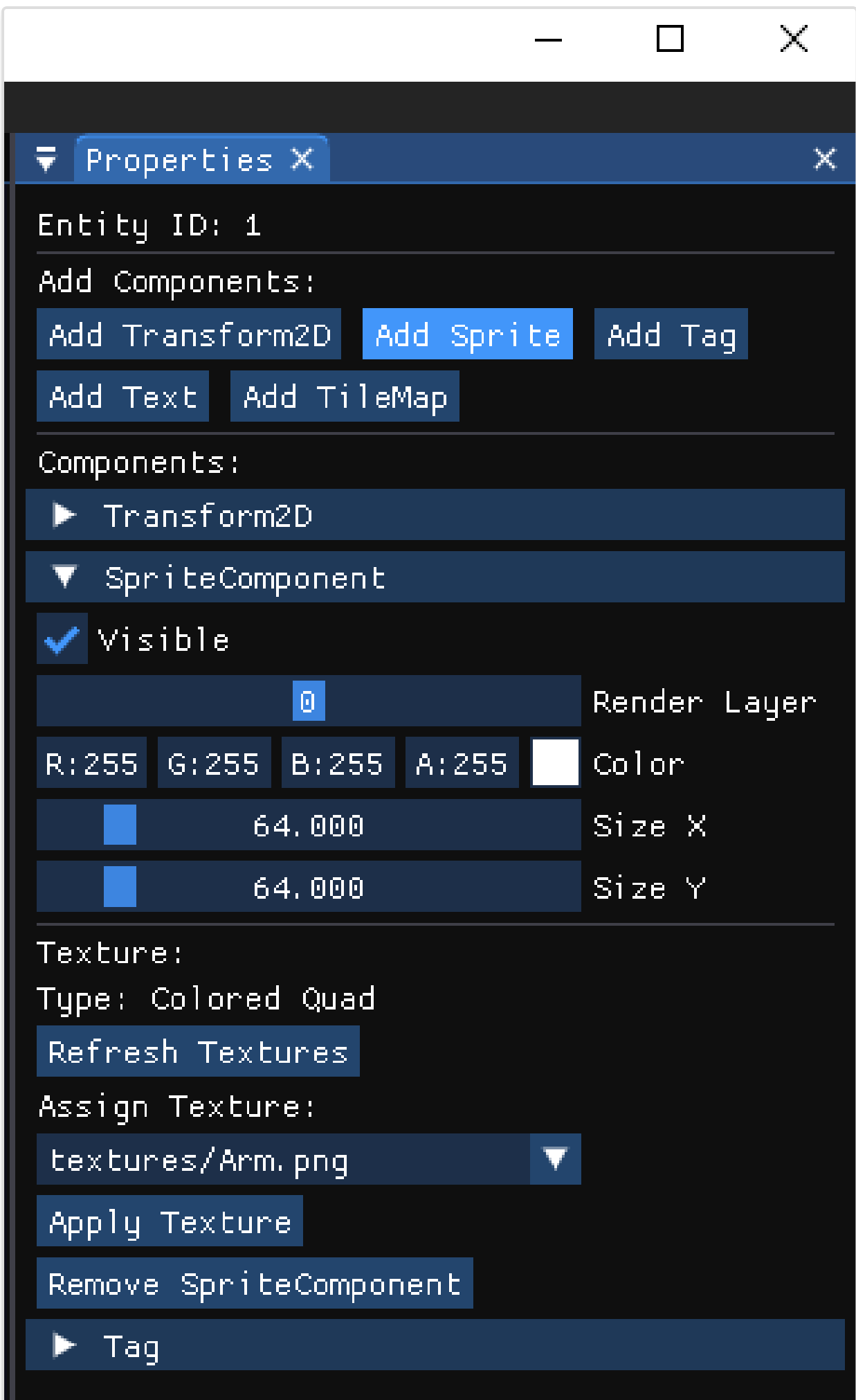
7. Sprite Component

— □ ✕

▼ Properties ✕   ✕

Entity ID: 1

Add Components:

Add Transform2D   Add Sprite   Add Tag

Add Text   Add TileMap

Components:

▶ Transform2D

▼ SpriteComponent

✔ Visible

        0   Render Layer

R:255   G:255   B:255   A:255   ⬜ Color

   ▮   64.000   Size X

   ▮   64.000   Size Y

Texture:
Type: Colored Quad
Refresh Textures
Assign Texture:

textures/Arm.png   ▼

Apply Texture

Remove SpriteComponent

▶ Tag

Add visual representation to entities with the Sprite component:

- **Visible:** Toggle sprite rendering on/off

- **Render Layer:** Drawing order (higher values render on top)

- **Color:** RGBA tint (default: white 255,255,255,255)

- **Size X/Y:** Sprite dimensions in pixels

- **Texture:** Displays current texture type (Colored Quad by default)

- **Assign Texture:** Dropdown to select texture from Asset Browser

> **Tip:** Click **Apply Texture** after selecting a texture from the dropdown to assign it to the sprite.
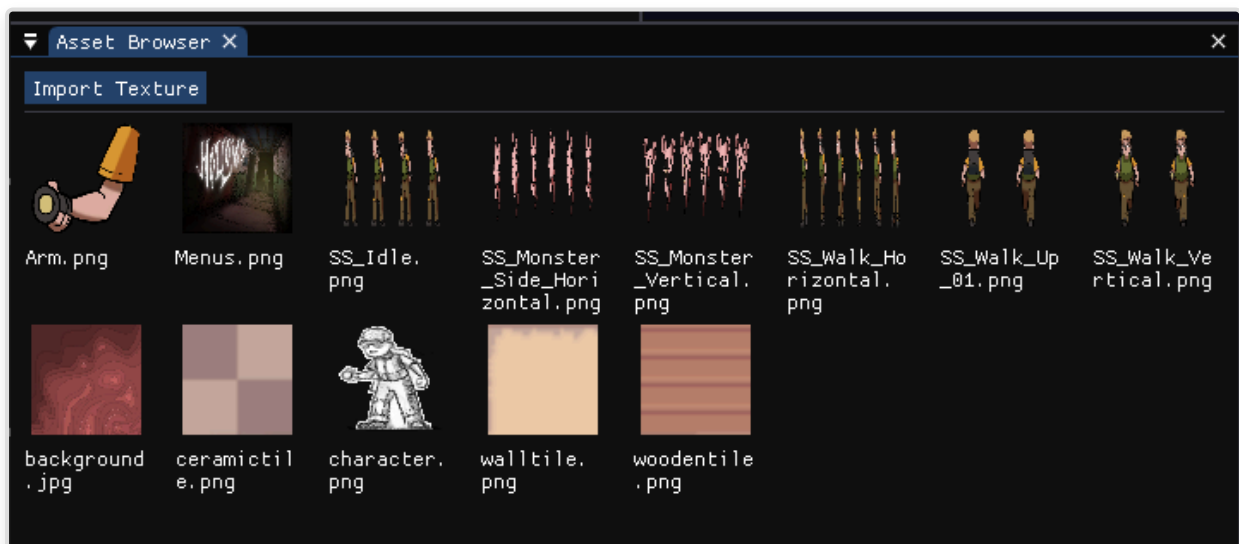
8. Asset Browser

Figure 8: Asset Browser showing available textures

The Asset Browser displays all available textures and fonts:

- **Import Texture:** Add new textures to the project

- **Texture Grid:** Thumbnail previews with filenames
- **Drag-and-Drop:** Drag textures into viewport to create sprite entities

Available textures automatically load from `assets/textures/` directory.
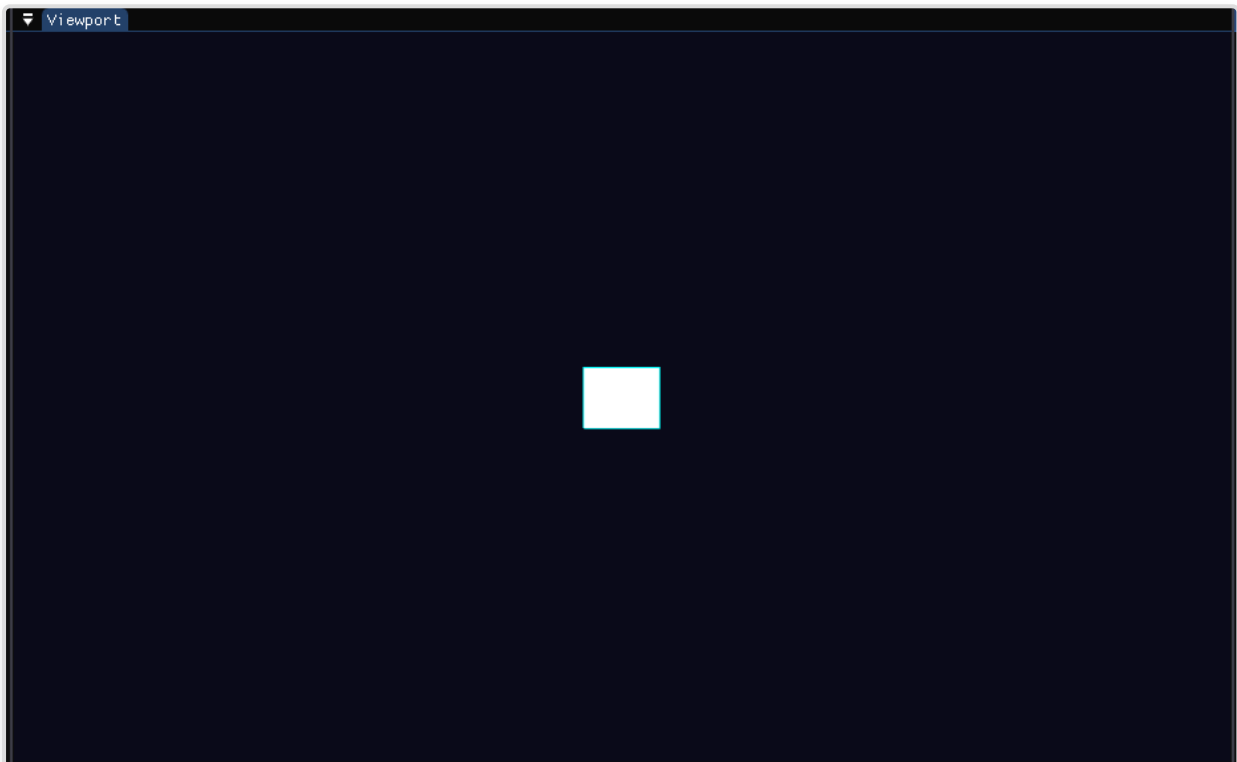
9. Viewport Interaction

Figure 9: Viewport showing sprite entity with selection outline

Interact with entities directly in the viewport:

- **Left Click:** Select entity (shows cyan selection outline)
- **Left Drag:** Move selected entity
- **Middle Mouse + Drag:** Pan camera
- **Q Key:** Zoom in
- **Z Key:** Zoom out

10. Console Panel

Figure 10: Console displaying log messages

The Console displays engine log messages:

- **INFO:** Normal operations (resource loading, entity creation)

- **WARNING:** Non-critical issues

- **ERROR:** Critical problems

Click **Clear** button to remove all messages from the console.
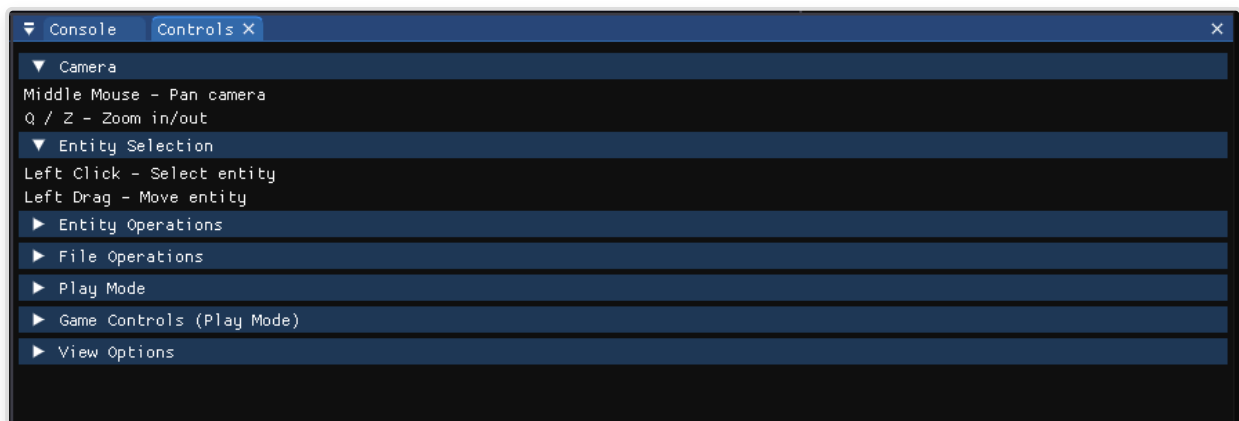
11. Controls Panel

The Controls panel provides quick reference for all editor shortcuts:

- **Camera:** Middle Mouse (pan), Q/Z (zoom)

- **Entity Selection:** Left Click (select), Left Drag (move)

- **Entity Operations:** Create, copy, delete entities

- **File Operations:** New, load, save scenes

- **Play Mode:** Test scene in-editor

- **View Options:** Toggle panel visibility
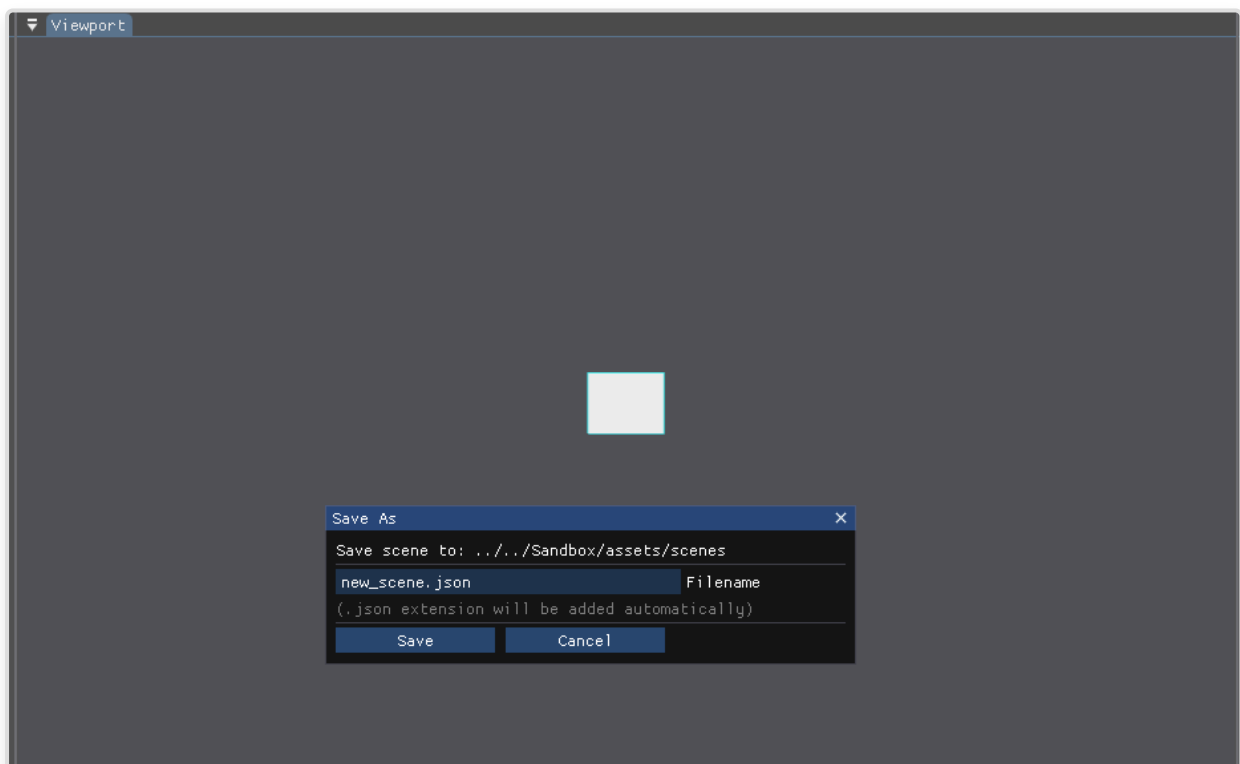
12. Saving Scenes

Figure 12: Save As dialog for scene persistence

Save your work using File > Save As:

1. Click **File > Save As**

2. Enter filename in the dialog (without extension)

3. Click **Save** button

Scenes are saved to `../../Sandbox/assets/scenes/`. The `.json` extension is added automatically if omitted.

> **Note:** Unsaved changes are tracked. The editor will prompt you to save before creating a new scene, loading another scene, or exiting.

13. Play Mode

GP2Engine - Level Editor - FPS: 200 | Frame Time: 5.00ms

File   View   Current Scene: (Unsaved) | Mode: EDIT   Play

Figure 13: Menu bar in Play Mode showing Stop button

Test your scene without leaving the editor:

- Click the green **Play** button to enter play mode

- Menu bar changes to show **Mode: PLAY**

- Play button becomes red **Stop** button

- Camera controls and entity editing are disabled during play mode

- Click **Stop** to return to edit mode

Summary

The GP2Engine Level Editor provides a complete workflow for scene creation:

- Panel-based interface with customizable layout

- Entity-Component architecture for flexible object design

- Real-time visual editing with immediate feedback

- Drag-and-drop asset workflow

- In-editor play mode testing

- JSON-based scene persistence

For help: Refer to the Controls panel (View > Controls) for all keyboard shortcuts and mouse controls.