Table of Contents

SODD Unity Framework

Welcome to the SODD Unity Framework, a robust tool for architecting your games using ScriptableObjects in Unity.

Introduction

Overview of the SODD Framework

The ScriptableObject Driven Development (SODD) Framework is an innovative and efficient approach to game development in Unity, designed to enhance modularity, maintainability, and scalability. Building on the foundational concepts introduced by Ryan Hipple at the Unite Austin conference in 2017, the SODD Framework leverages the power of ScriptableObjects to streamline the development process and overcome common challenges associated with traditional MonoBehaviour-centric architectures.

ScriptableObjects in Unity provide a robust way to manage data independently of the GameObjects in the scene, promoting a more organized and decoupled system. By encapsulating game events, variables, and collections within ScriptableObjects, the SODD Framework enables developers to create self-contained, reusable components that can be easily managed and modified without directly affecting other parts of the game.

Benefits and Applications

The SODD Framework offers several key benefits that make it a valuable tool for both independent developers and large development teams:

- **Modularity**: The framework allows for the creation of independent, self-contained game components. This reduces interdependencies and makes it easier to manage and update individual parts of the game without impacting the entire system. Modular components can be reused across different projects, saving development time and effort.
- **Editability**: With ScriptableObjects, game data can be easily modified through the Unity Inspector, enabling rapid prototyping and iterative development. Designers and developers can adjust game parameters on the fly, without needing to dive into the codebase, thus enhancing flexibility and responsiveness during the development process.
- **Debuggability**: The framework includes features for logging and tracking events and data changes, which simplifies debugging and testing processes. By maintaining clear and detailed logs of event invocations and variable updates, developers can quickly identify and resolve issues, ensuring a smoother development experience.

- **Scalability**: The decoupled nature of ScriptableObjects allows for scalable game architectures. As the game grows in complexity, the framework ensures that new features and systems can be added without causing significant disruptions to the existing structure.

Purpose and Goals

The primary goal of the SODD Framework is to provide a comprehensive toolset that enhances productivity and collaboration in Unity's development environment. By offering a structured approach to managing game data and events, the framework aims to:

1. **Simplify Dependency Management**: Reduce the complexity of interactions between different game systems, making it easier to maintain and expand the game over time.
2. **Enhance Collaboration**: Enable designers, developers, and other team members to work more effectively by providing clear interfaces and tools for modifying game behavior and data.
3. **Improve Code Quality**: Promote best practices in game engineering, such as modularity, data-driven design, and event-driven architecture, to create cleaner, more maintainable codebases.

Target Audience

The SODD Framework is designed for a wide range of Unity developers, from beginners to seasoned professionals. Whether you are an indie developer working on a small project or part of a large team developing a complex game, the framework provides the tools and methodologies to improve your development process. Familiarity with Unity and C# programming is recommended to fully leverage the capabilities of the SODD Framework.

This is the documentation for the Scripting APIs of this package.

3.5.1 (2024-06-03)

Bug Fixes

- null safety in InputActionIconProvider class (237ed9d)
- reference field of VoidInputActionHandler is now displayed in Inspector (b5bf4ef)

3.5.0 (2024-05-30)

Bug Fixes

- **input:** icon provider (6deef32)

Features

- add passive scriptable objects (9e5b703)
- **icons:** add CollisionDetector icon (d2e82e0)
- **icons:** add CollisionDetector2D icon (3053644)
- **icons:** add InputActionIconProvider icon (dd43290)
- **icons:** add OptionSelector icon (e32b1a4)
- **repositories:** add persistent scriptable object functionality (09db724)

3.4.0 (2024-05-30)

Features

- **ai:** add state machine method (6087bfd)
- **observers:** added multiplier to float observer (2efcdfb)

3.3.0 (2024-05-26)

Features

- add reload resources button in SODD Menu (34cf76c)
- **input:** add input action icon provider (7791379)
- **input:** add input control scheme handler (8bc55b3)
- **input:** add input icon repository (5f67f4a)

3.2.0 (2024-05-25)

Bug Fixes

- **handlers:** bool action handler now registers value to variable (bad2ea7)

Features

- **extensions:** add `isEmpty()` extension method to IEnumerable (1941f47)
- **repositories:** add variable repository debugging (5d42ecd)

3.1.0 (2024-05-25)

Bug Fixes

- **data:** serializable dictionary entry height adapts to content (b90b5b3)

Features

- **handlers:** action handlers now reference variables too (67c38c7)

3.0.0 (2024-05-23)

Features

- **ai:** add 2D collision detector (5d7b2f1)
- **ai:** add 3D collision detector (788899a)
- **ai:** add state machine (b926c21)
- **data:** add serializable dictionary data type (4d5acec)
- **data:** add serializable range data type (cc635fc)
- **ui:** add option selector ui component (3048af5)

BREAKING CHANGES

- **ui:** `Unity.TextMeshPro` is now a package dependency

Signed-off-by: Alex Ruiz a.ruiz.rabasseda@gmail.com

2.0.1 (2024-05-19)

Bug Fixes

- **documentation:** documentation organization (ba196db)
- **documentation:** table of contents (b48d2e9)

2.0.0 (2024-05-15)

Features

- add Comparison enum for comparison operations (b0fc547)
- **attributes:** add Collapsible attribute (6362242)
- **attributes:** add Disabled attribute (b5bbad8)
- **attributes:** add DisableIf attribute (0573352)

- **attributes:** add DisableIfMatch attribute (d93a292)
- **attributes:** add HideIf attribute (83252af)
- **attributes:** add HideIfMatch attribute (d0601c8)
- **attributes:** add ShowIf attribute (2a025a7)
- **attributes:** add ShowIfMatch attribute (67aef74)
- **observers:** add abstract VariableObserver class (5df3ab7)
- **observers:** add bool variable observer (fa5cbb8)
- **observers:** add float variable observer (813a326)
- **observers:** add int variable observer (f699f82)
- **observers:** add string variable observer (7cfb246)
- **observers:** add vector2 variable observer (ae0b580)
- **observers:** add vector3 variable observer (4a79681)
- **repositories:** add binary file repository (d233eed)
- **repositories:** add file repository abstract class (7868fe3)
- **repositories:** add IRepository interface (60c277e)
- **repositories:** add json file repository (14c8e15)
- **repositories:** add variable repository (ea8d3c6)

BREAKING CHANGES

- **repositories:** changes IVariable and Variable signature

Signed-off-by: Alex Ruiz a.ruiz.rabasseda@gmail.com

1.2.0 (2024-05-07)

Features

- add coroutine builder (647f3b3)
- **extensions:** add AudioMixer extensions (481c814)
- **extensions:** add GameObject extension methods for retrieving components (2c1c720)
- **extensions:** add GameObject extension methods for sending messages (4ac7eac)
- **extensions:** add GameObject extensions for vector operations (27fc44f)
- **extensions:** add IEnumerable extensions (a060d60)
- **extensions:** add IsInLayerMask extension method for GameObject type (88d98e0)
- **extensions:** add NavMeshAgent extensions (5abe4a0)
- **extensions:** add primitive type extensions (6429033)
- **extensions:** add Transform extensions for vector operations (9c647c5)
- **extensions:** add Vector2 extensions for vector operations (5d135f3)
- **extensions:** add Vector3 extensions for vector operations (4585ba3)
- **variables:** added value reference (034ada8)

1.1.0 (2024-04-15)

Features

- **collections:** collections now implement IList interface and have a method to get random items (48784a1)

1.0.0 (2024-03-02)

Features

- add EditorHelper and Framework classes (d494a8c)
- add icons for events, listeners, variables, collections and action handlers (f3a67f5)
- **attributes:** add OnValueChanged attribute (5d19606)
- **collections:** add abstract Collection class for scriptable collections (adc4fd6)
- **collections:** add AudioClipCollection (42ceb3c)
- **collections:** add ComponentCollection (dc22ba6)
- **collections:** add GameObjectCollection (a4fb546)
- **collections:** add ObjectCollection (1a6e6bc)
- **collections:** add ScriptableObjectCollection (6c5abc2)
- **editor:** update event attributes and add EventEditor (e0aa829)
- **events:** add abstract Event class for scriptable events (dbe9d5a)
- **events:** add BoolEvent (11a1cea)
- **events:** add EventListener abstract class (b5d6ba7)
- **events:** add FloatEvent (466d076)
- **events:** add GameObjectEvent (43c52fc)
- **events:** add IEvent interface (fe6d844)
- **events:** add IEventListener interface (4a227ce)
- **events:** add IntEvent (e8cea71)
- **events:** add StringEvent (87f09f6)
- **events:** add StringEventListener (28fd932)
- **events:** add Vector2Event (0c73d3f)
- **events:** add Vector3Event (4f26d89)
- **events:** add VoidEvent (49ff349)
- **input:** add abstract InputActionHandler class for scriptable action handlers (cb7a9a0)
- **input:** add BoolInputActionHandler (d5f8d72)
- **input:** add FloatInputActionHandler (3ccde51)
- **input:** add Vector2InputActionHandler (4502560)
- **input:** add Vector3InputActionHandler (f6ce12a)
- **input:** add VoidInputActionHandler (045ea1b)
- **listeners:** add BoolEventListener (2406b04)
- **listeners:** add FloatEventListener (c967cfe)

- **listeners:** add GameObjectEventListener (115aeeb)
- **listeners:** add IntEventListener (e80bf48)
- **listeners:** add Vector2EventListener (d077d86)
- **listeners:** add Vector3EventListener (efbd266)
- **listeners:** add VoidEventListener (2075494)
- **variables:** add abstract Variable class for scriptable variables (fbb8385)
- **variables:** add BoolVariable (4bb16dd)
- **variables:** add FloatVariable (52cae6a)
- **variables:** add GameObjectVariable (2eec78a)
- **variables:** add generic variable interfaces (f40e4ac)
- **variables:** add IntVariable (6e7b453)
- **variables:** add LayerMaskVariable (5cf5f30)
- **variables:** add StringVariable (ab24b07)
- **variables:** add Vector2Variable (2fa92a3)
- **variables:** add Vector3Variable (c5e3094)

BREAKING CHANGES

- **input:** `Unity.InputSystem` is now a package dependency

Signed-off-by: Alex Ruiz a.ruiz.rabasseda@gmail.com

MIT License