Ariadne
3D Dungeon Maker

Manual

Ver 1.0.1

Chapter 1

Overview

Ariadne 3D Dungeon Maker is a powerful asset to create 3D dungeons. In the extension, you can make grid-based map data in MapEditor.

And set it to the Game Controller prefab, Ariadne 3D Dungeon Maker produces dungeons according to your map data at runtime.

This asset includes the MoveController that enables movement in the dungeon.

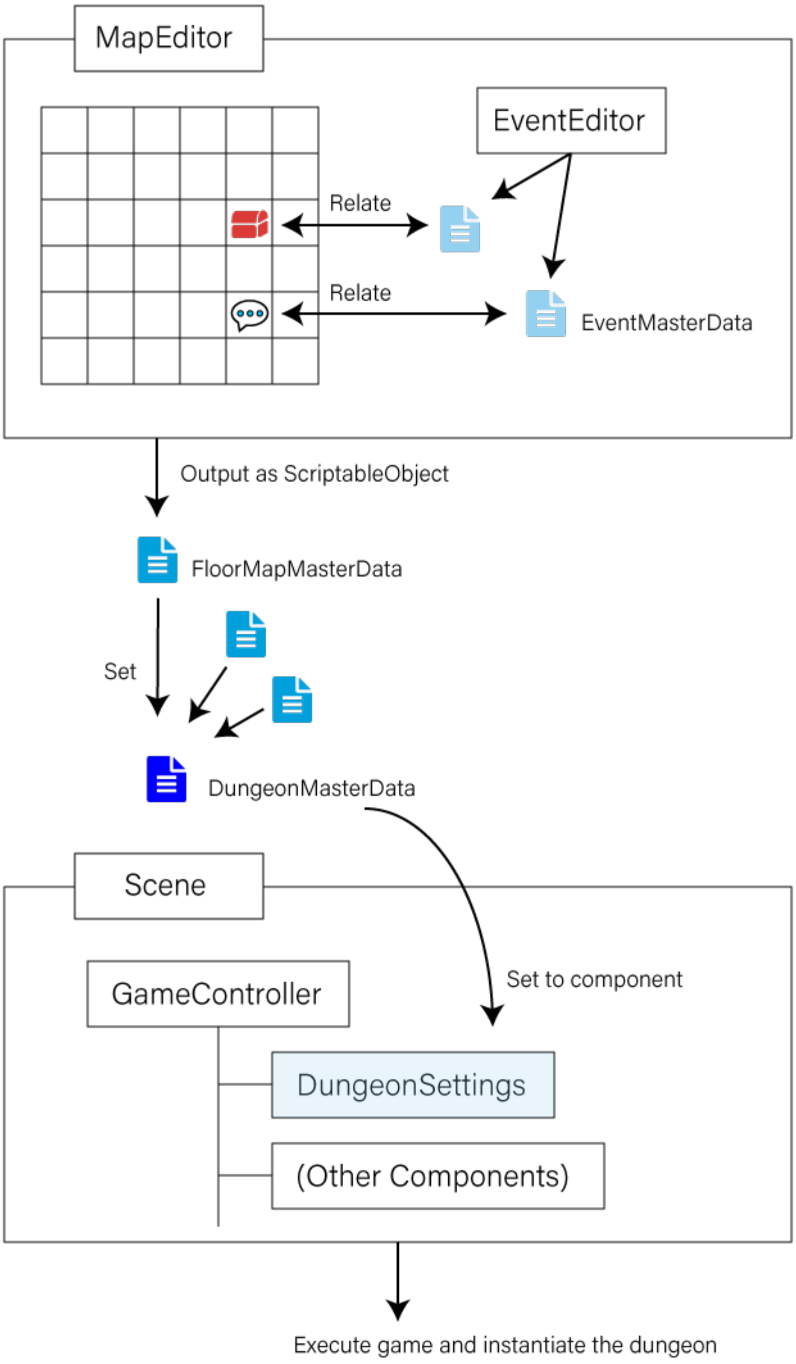
The controller enables processing events, too.

Chapter 2

Getting Started

Data flow

Here is the data flow of Ariadne 3D Dungeon Maker.



Workflow to Generate Dungeon

Create a map data by using MapEditor

First, create a map data by using MapEditor. You can open the MapEditor from [**Window/MapEditor**] in the menu bar.

On the MapEditor, you can set attributes of the map by using draw tools. To add events, select a position using select tool and press [**Open Event Editor**] button. On the EventEditor, you can define contents of events and starting conditions.

Set the map data to Dungeon Data

Second, after saving the map data, create a dungeon data from [**Asset/Create/Ariadne/DungeonData**] in the menu bar.

The dungeon data is a holder of map data. Set map data that you created to dungeon data.

Ready objects in the scene

Next, set the dungeon data to GameController object in the Scene. The GameController object has a component named DungeonSettings, so set the dungeon data to this component.

Required objects are placed in [**Ariadne/Resources/Prefabs/SceneObjects**] folder and [**Ariadne/Resources/Prefabs/SceneObjects/CanvasParts**] folder as prefabs.

When you intend to create a new scene, it is useful to duplicate demo scene and customize it.

Execute the game

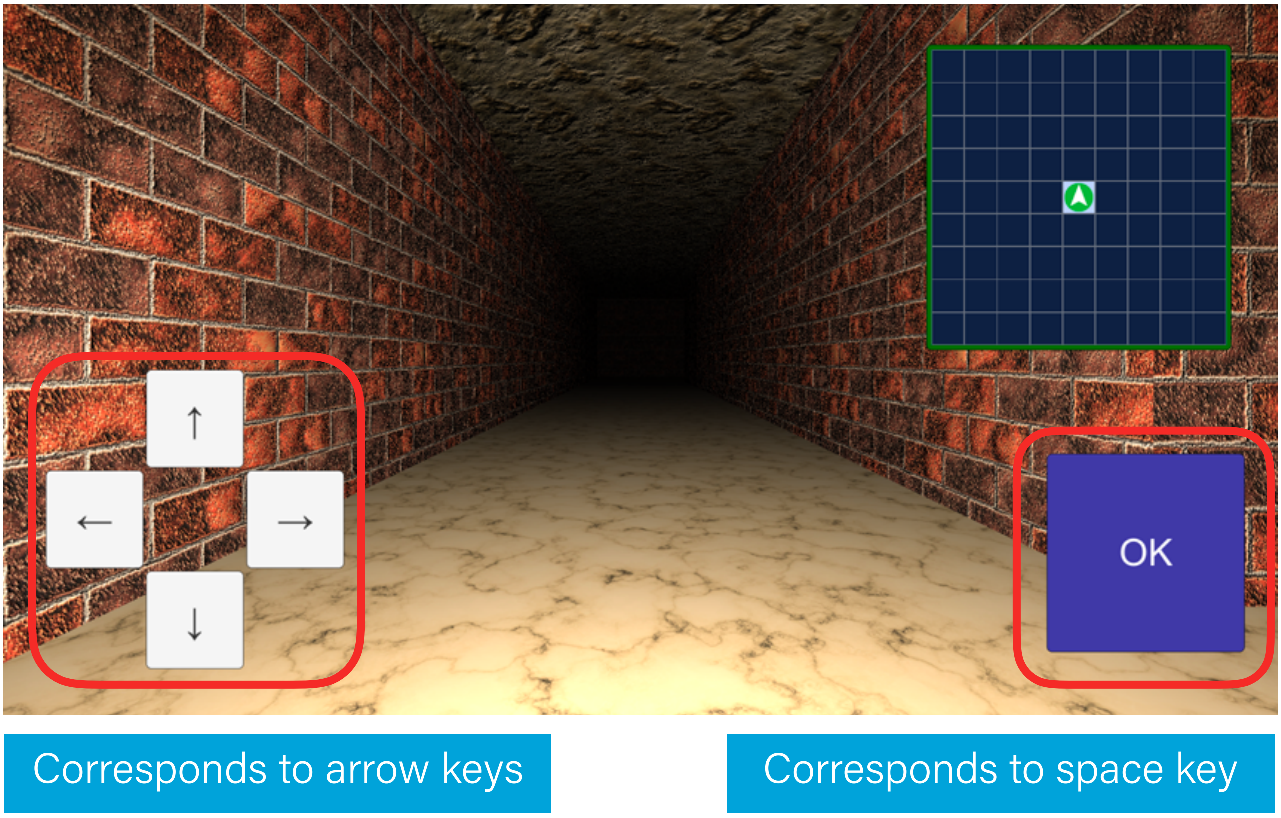
Finally, ready to explore your dungeon. Execute the game!

Operation in the Dungeon

To move in the dungeon, you can use arrow keys and space key.

Key	Action
Up Arrow	Moves the player to forward.
Left Arrow	Turns the player in a counterclockwise direction.
Right Arrow	Turns the player in a clockwise direction.
Down Arrow	Turns the player around to back.
Space	Start the event and decide.

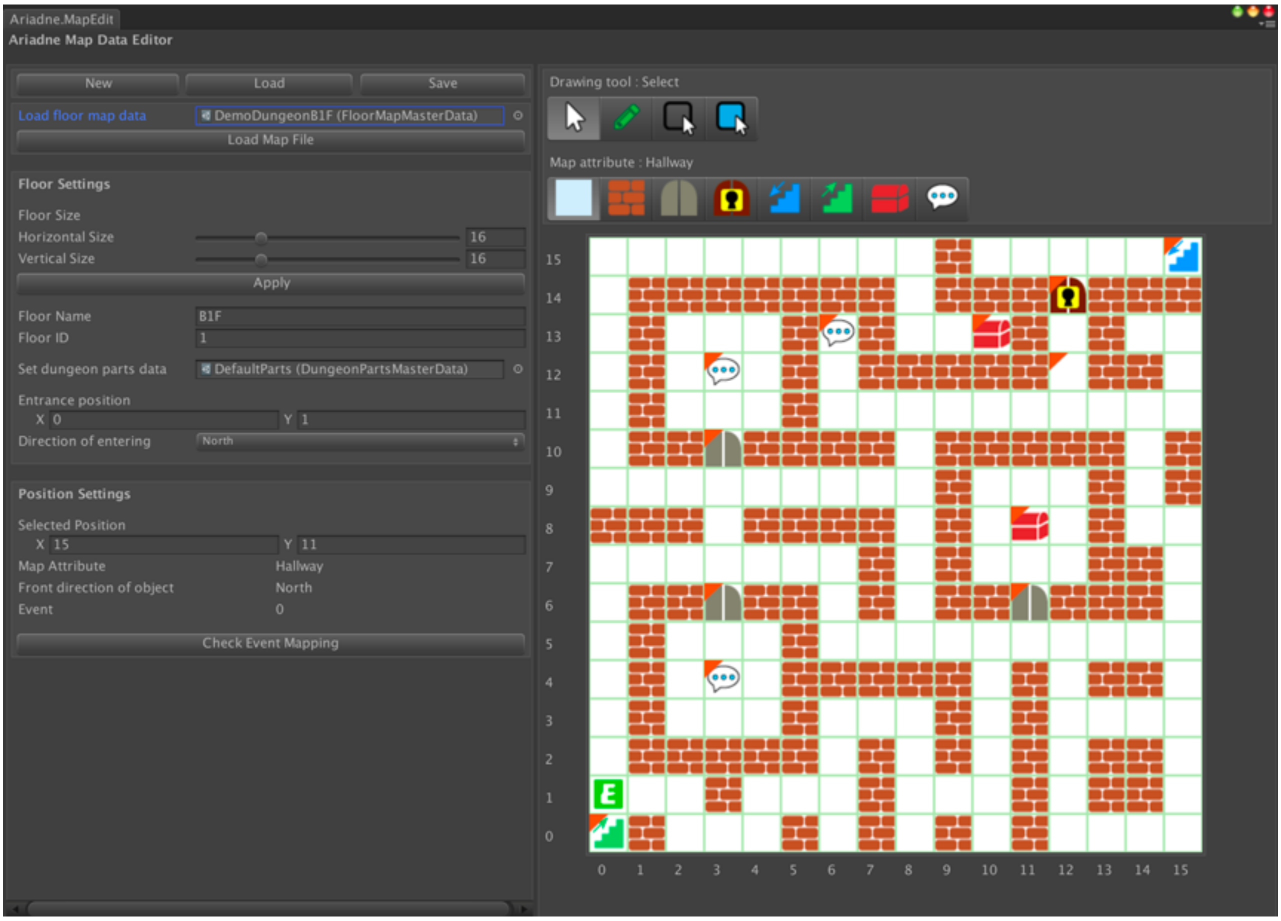
And you can also use uGUI button on the screen. Those buttons can be set invisible in MoveController component.



Chapter 3

Editor Extensions

MapEditor



Ariadne.MapEdit
Ariadne Map Data Editor

Load floor map data: DemoDungeonB1F (FloorMapMasterData)

Floor Settings
Floor Size
Horizontal Size: 16
Vertical Size: 16
Floor Name: B1F
Floor ID: 1
Set dungeon parts data: DefaultParts (DungeonPartsMasterData)
Entrance position: X 0, Y 1
Direction of entering: North

Position Settings
Selected Position: X 15, Y 11
Map Attribute: Hallway
Front direction of object: North
Event: 0

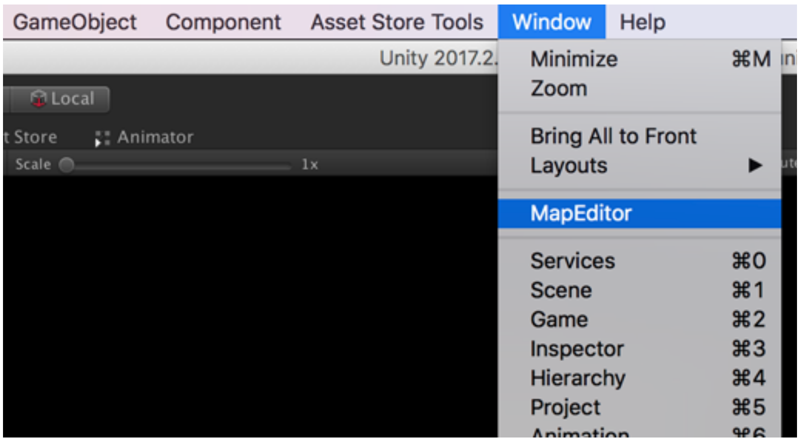
Drawing tool : Select
Map attribute : Hallway

MapEditor is a core function of Ariadne 3D Dungeon Maker. MapEditor edits and outputs **FloorMapMasterData** which defines the size of the floor, map ID, map name and detailed information about each position in the map.

See also the description of FloorMapMasterData.

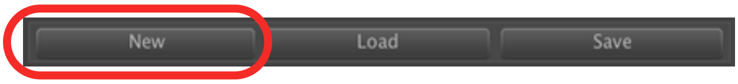
How to open MapEditor

Open MapEditor from [**Window/MapEditor**] on the menu bar.

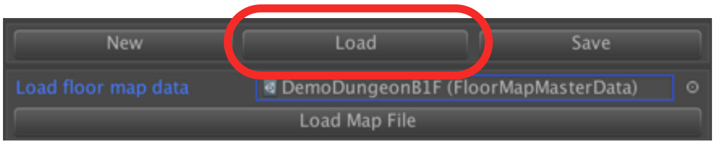


Settings pane

File operation pane

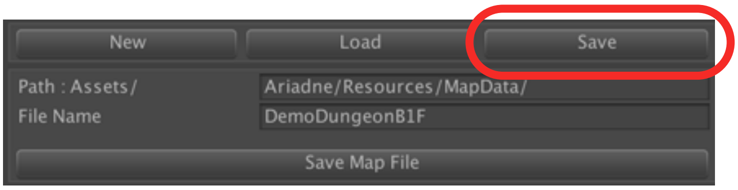


To create new FloorMapMasterData, press [**New**] button.



Load floor map data DemoDungeonB1F (FloorMapMasterData)

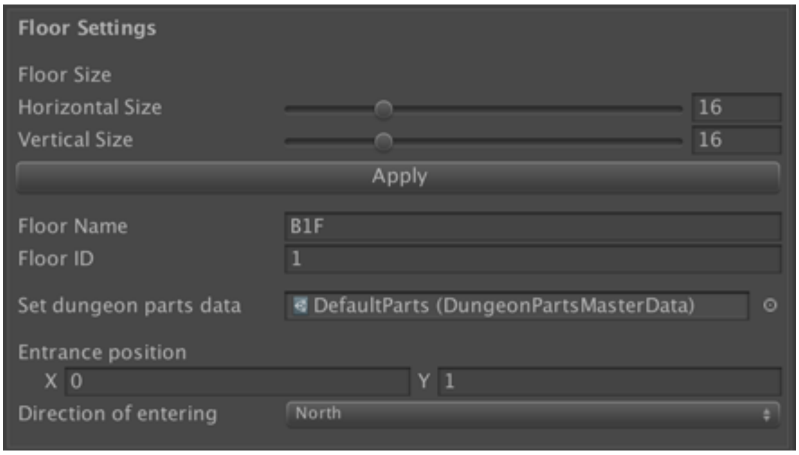
Press [**Load**] button, then load pane will be shown. On the load pane, select FloorMapMasterData to load. And press [**Load Map File**], MapEditor loads the specified file.



Path : Assets/ Ariadne/Resources/MapData/
File Name DemoDungeonB1F

Press [**Save**] button, then Save pane will be shown. On the Save pane, specify the path and file name to save. And press [**Save Map File**], MapEditor saves the file as asset file.

Floor Settings pane



Horizontal Size 16
Vertical Size 16
Floor Name B1F
Floor ID 1
Set dungeon parts data DefaultParts (DungeonPartsMasterData)
Entrance position X 0 Y 1
Direction of entering North

Define “**Horizontal Size**” and "**Vertical Size**” of the floor. To apply sizes to map, press [**Apply**] button. If the value of the slider is smaller than previous one, a part of map attribute will be discarded.

Set “**Floor Name**” and “**Floor ID**”. Floor ID is used in the DungeonMasterData to specify the start floor of the dungeon. If there are some FloorMapMasterData which have the same floor ID, the selector in DungeonMasterData shows only one of them. So you have to specify different IDs for those floors.

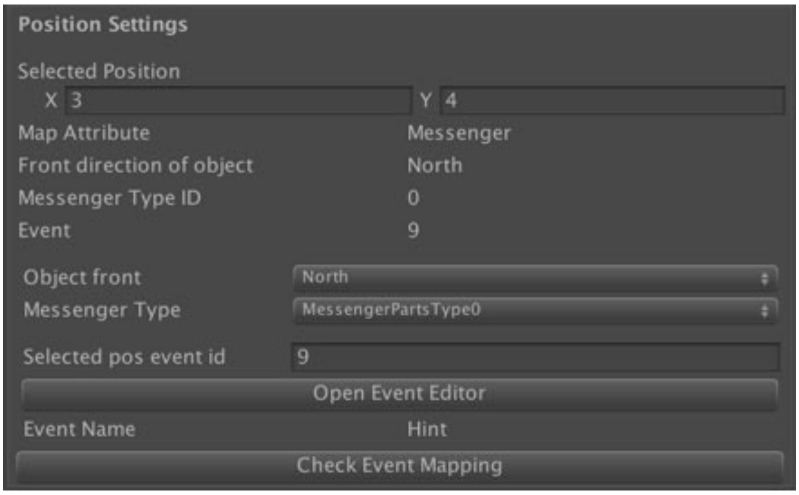
DungeonPartsMasterData is the set of dungeon parts, such as a wall prefab. See also DungeonPartsMasterData.

When this field is null, MapEditor sets the default parts automatically.

"**Entrance Position**" is the first position when the player enters this dungeon. The setting of the first floor of the dungeon is in DungeonMasterData. See also DungeonMasterData.

You can specify the direction of entering, too.

Position Settings pane



Position Settings

Selected Position	
X	3
Y	4
Map Attribute	Messenger
Front direction of object	North
Messenger Type ID	0
Event	9
Object front	North
Messenger Type	MessengerPartsType0
Selected pos event id	9
Event Name	Hint

"Position Settings pane" shows information about the selected position.

"**Selected position**" indicates the position on the grid where the mouse cursor is. When the "**Select**" tool is selected, this shows the position where you selected.

Label fields show information about the position.

"**Object front**" selector defines the direction of dungeon parts.

"**Messenger Type**" defines the parts index of messenger parts. This selector is visible when the map attribute is Messenger.

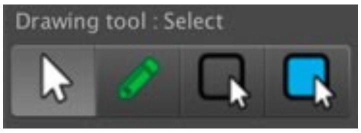
"**Selected pos event id**" is the event ID of the position. When you input the value larger than 0, you can select [**Open Event Editor**] and edit the event. See also EventEditor for more detail.

Press [**Check Event Mapping**] button shows EventMappingViewer. This viewer shows mappings between FloorMapMasterData and EventMasterData. See also EventMappingViewer.

Drawing map pane

"Drawing map pane" is editing place for attributes of each position.

Drawing tool



"**Select**" tool can pick up information about the position. You can see them in "Position Settings pane".

"**Draw**" tool can set an map attribute value. When you drag mouse cursor, this tool sets map attributes to positions where you drag.

"**Draw rect**" tool can set map attributes as a rectangle.

"**Draw filled rect**" tool is similar to "Draw rect" tool. This tool fills inside of the rectangle.

Map attribute

By default, Ariadne defines follow map attributes.

ID	Icon	Attribute	Can a player walk?
0	⬜	Hallway	Yes
1	🧱	Wall	No
2	🚪	Door	No
3	🔒	Locked Door	No
4	↙	Downstairs	No
5	↗	Upstairs	No
6	🧰	Treasure	No
7	💬	Messenger	No
8	🏛	Pillar	No
9	💡	WallWithTorch	No

Grid-based map

Ariadne 3D Dungeon Maker uses grid-based map for instantiating dungeons and holding the player position.

The origin is left-bottom.

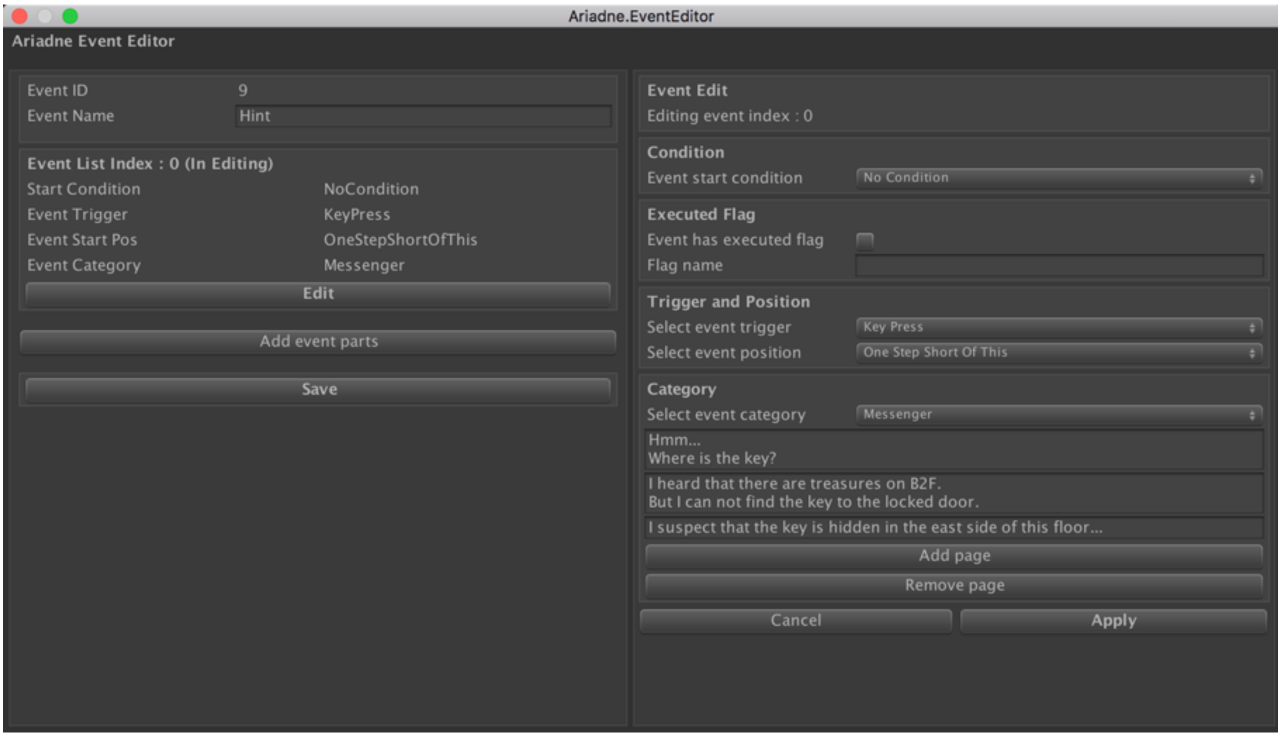
How to close the window

To close this window, click the close button on the window.

When this window lost focus, editing data will be saved as temporary data in [**Assets/Ariadne/Resources/MapData/MapEditor**] folder.

Attention: Don't set temporary data to DungeonMasterData to prevent unexpected map changing. When you use FloorMapMasterData, save the file by using [**Save Map File**] button.

EventEditor



Ariadne.EventEditor

Ariadne Event Editor

Event ID 9

Event Name Hint

Event List Index : 0 (In Editing)

Start Condition NoCondition

Event Trigger KeyPress

Event Start Pos OneStepShortOfThis

Event Category Messenger

Event Edit

Editing event index : 0

Condition

Event start condition: No Condition

Executed Flag

Event has executed flag □

Flag name

Trigger and Position

Select event trigger: Key Press

Select event position: One Step Short Of This

Category

Select event category: Messenger

Hmm...
Where is the key?

I heard that there are treasures on B2F.
But I can not find the key to the locked door.

I suspect that the key is hidden in the east side of this floor...

EventEditor is the tool for editing events. The event is an action on the dungeon, such as opening the door. EventEditor saves events as **EventMasterData** that inherit ScriptableObject.

How to open EventEditor

Open EventEditor by pressing [**Open Event Editor**] button on MapEditor window.

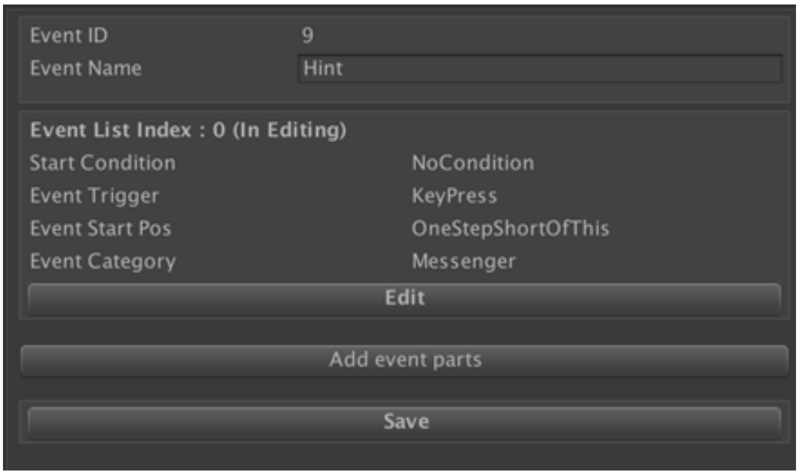
About EventParts

EventMasterData have parts group called EventParts. EventParts defines an action in the dungeon, such as opening the door.

Each EventParts has a condition for starting event. By checking the condition, Ariadne 3D Dungeon Maker decides the target of the processing event. You can set "executed flag" to them, and you can use the flag to check condition.

Ariadne 3D Dungeon Maker checks conditions by descending order from the end of EventParts list.

EventMasterData Setting



Event ID 9

Event Name Hint

Event List Index : 0 (In Editing)

Start Condition NoCondition

Event Trigger KeyPress

Event Start Pos OneStepShortOfThis

Event Category Messenger

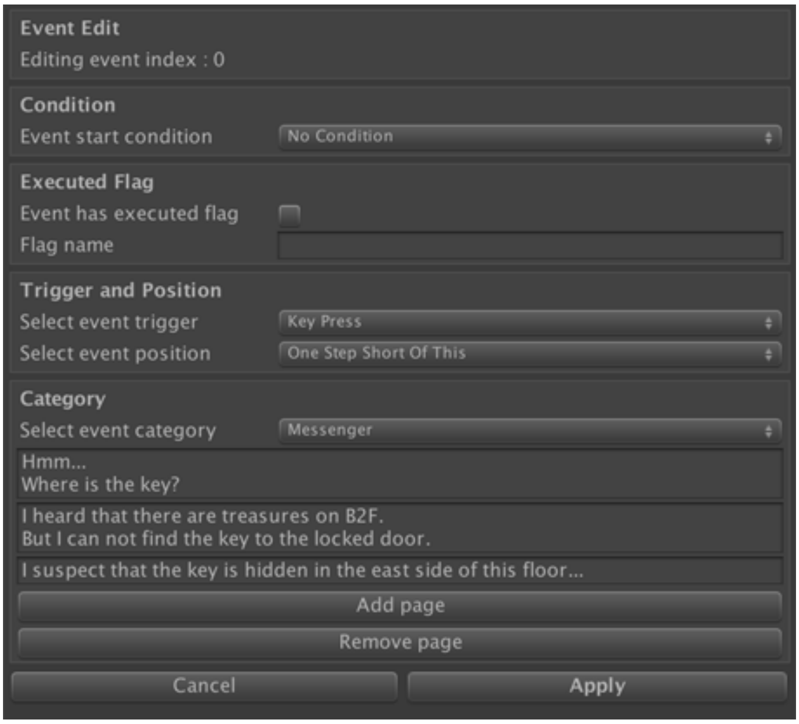
"**Event Name**" field defines the name of this event. This event name is shown in MapEditor window.

To edit an EventParts, press [**Edit**] button. EventEditor shows settings of selected EventParts on the right pane.

If you want to add a new EventParts, press [**Add event parts**] button. This action adds a new element to EventParts list. When there is more than one EventParts, [**Remove**] button will be visible on the left of [**Edit**] button. To remove the target EventParts, press [**Remove**] button.

To save EventMasterData, press [**Save**] button. EventEditor outputs EventMasterData file to [**Assets/Ariadne/Resources/EventData**] folder.

EventParts Setting



Event Edit

Editing event index : 0

Condition

Event start condition: No Condition

Executed Flag

Event has executed flag □

Flag name

Trigger and Position

Select event trigger: Key Press

Select event position: One Step Short Of This

Category

Select event category: Messenger

Hmm...
Where is the key?

I heard that there are treasures on B2F.
But I can not find the key to the locked door.

I suspect that the key is hidden in the east side of this floor...

After changing the EventParts settings, press [**Apply**] button to apply changes for the EventParts. If you want to revert changes, use [**Cancel**] button.

Condition

Select a condition for starting the EventParts.

Condition	Description
No Condition	The event starts without a condition.
Flag	The event starts when the selected flag is true.
Item	The event starts when the player has specified item. You can specify a comparison operator for comparing the value of the item.
Money	The event starts when the player has specified amount of money. You can specify a comparison operator for comparing the amount of money.

Executed flag

You can set an executed flag to EventParts. When you use this flag, check [**Event has executed flag**] and input the flag name. EventEditor uses this name in the selector of the condition setting.

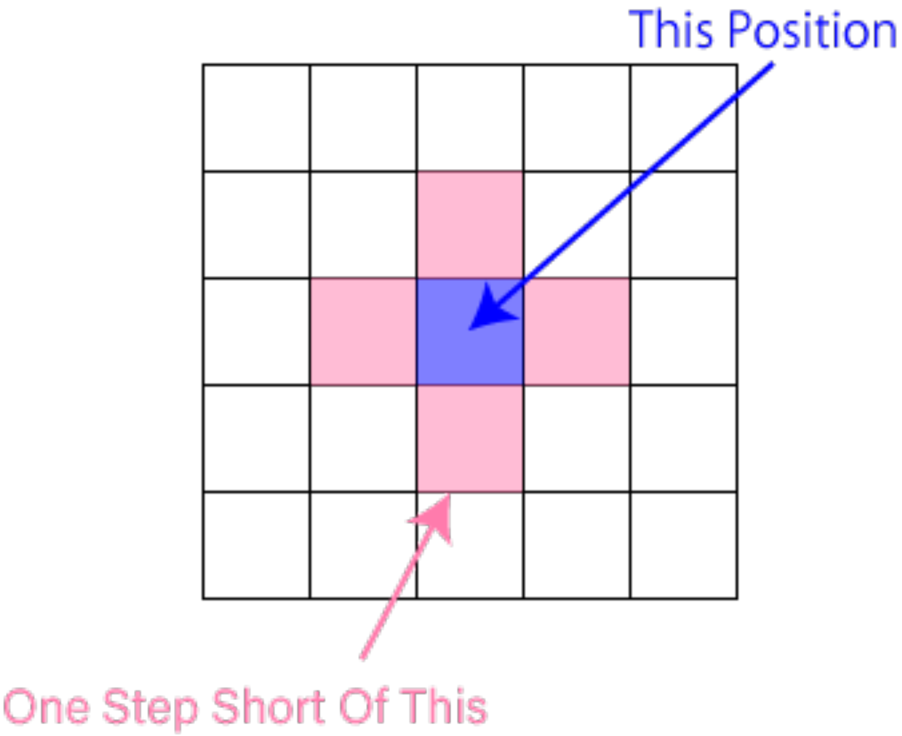
Trigger and Position

The event trigger defines how to start the event.

Trigger	Description
KeyPress	The event starts by pressing the space key.
Auto	The event starts after player moving automatically.

The event position is the place where the event starts.

Position	Description
This Position	The event starts at the position where this event is defined.
One Step Short Of This	The event starts at the position next to this event.



Event category

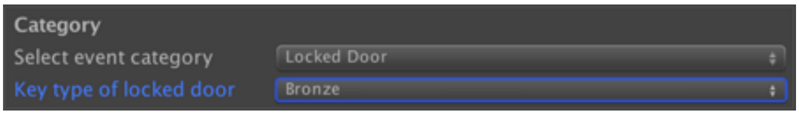
None

This event does nothing.

Door

This event calls the process of opening the door in the dungeon.

Locked Door

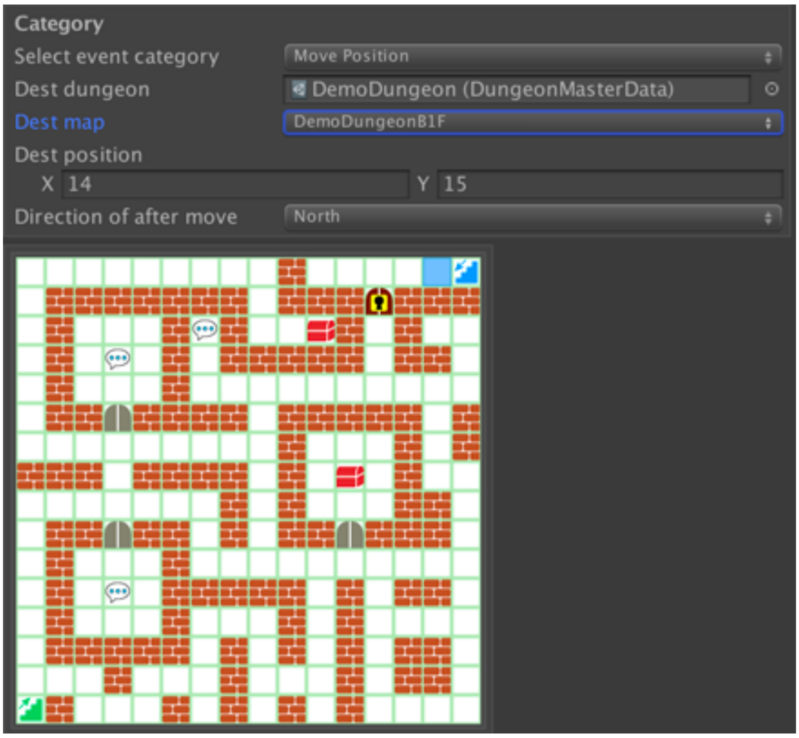


Category
Select event category: Locked Door
Key type of locked door: Bronze

Select the key type of the door. When the key type is matched, this event calls the process of opening the door. If you specified "None" to the key type, this event works the same as Door event.

The key type is defined in DoorKeyType enum in **AriadneEnumerations.cs**.

Move Position



Category

Select event category: Move Position

Dest dungeon: DemoDungeon (DungeonMasterData)

Dest map: DemoDungeonB1F

Dest position

X 14 Y 15

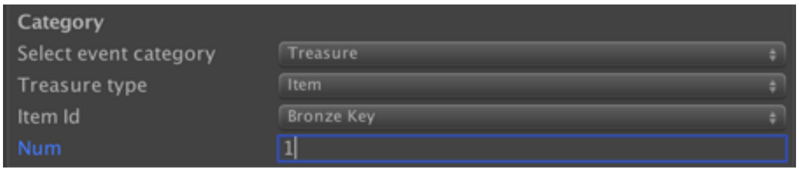
Direction of after move: North

Set the position where the player moves to.

You have to set destination dungeon, destination floor, position, and direction.

A moving event such as downstairs uses this event category.

Treasure



Category

Select event category: Treasure

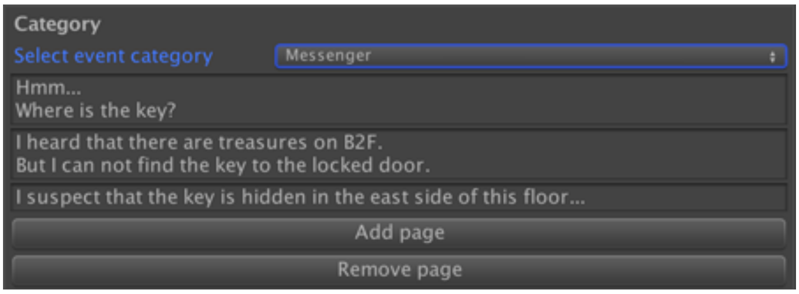
Treasure type: Item

Item Id: Bronze Key

Num: 1

Select treasure type and set the item ID or amount of money that the player gets.

Messenger



Category

Select event category: Messenger

Hmm...
Where is the key?

I heard that there are treasures on B2F.
But I can not find the key to the locked door.

I suspect that the key is hidden in the east side of this floor...

Set messages which you want to show on the screen.

By pressing [**Add page**] button, EventEditor adds a new page. And [**Remove page**] is pressed, EventEditor removes the last page.

Exit Position

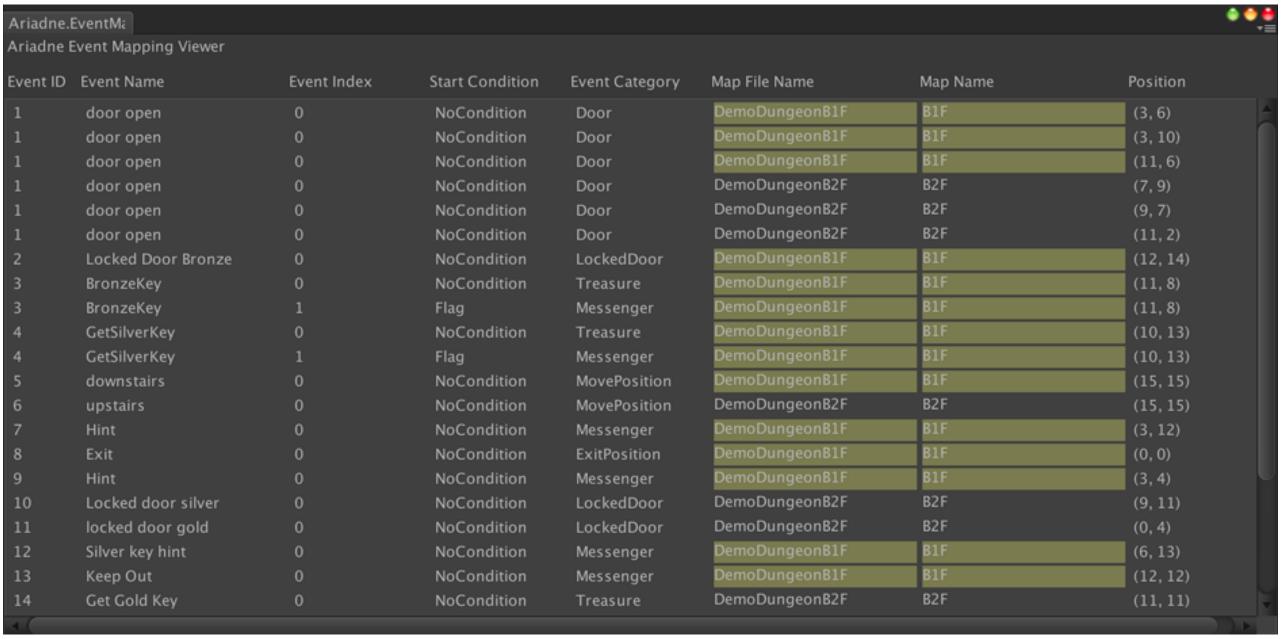
Set this event as exit position of the dungeon.

How to close the window

To close this window, click the close button on the window.

If you close this window without saving, changes to this event are discarded.

EventMappingViewer



Event ID	Event Name	Event Index	Start Condition	Event Category	Map File Name	Map Name	Position
1	door open	0	NoCondition	Door	DemoDungeonB1F	B1F	(3, 6)
1	door open	0	NoCondition	Door	DemoDungeonB1F	B1F	(3, 10)
1	door open	0	NoCondition	Door	DemoDungeonB1F	B1F	(11, 6)
1	door open	0	NoCondition	Door	DemoDungeonB2F	B2F	(7, 9)
1	door open	0	NoCondition	Door	DemoDungeonB2F	B2F	(9, 7)
1	door open	0	NoCondition	Door	DemoDungeonB2F	B2F	(11, 2)
2	Locked Door Bronze	0	NoCondition	LockedDoor	DemoDungeonB1F	B1F	(12, 14)
3	BronzeKey	0	NoCondition	Treasure	DemoDungeonB1F	B1F	(11, 8)
3	BronzeKey	1	Flag	Messenger	DemoDungeonB1F	B1F	(11, 8)
4	GetSilverKey	0	NoCondition	Treasure	DemoDungeonB1F	B1F	(10, 13)
4	GetSilverKey	1	Flag	Messenger	DemoDungeonB1F	B1F	(10, 13)
5	downstairs	0	NoCondition	MovePosition	DemoDungeonB1F	B1F	(15, 15)
6	upstairs	0	NoCondition	MovePosition	DemoDungeonB2F	B2F	(15, 15)
7	Hint	0	NoCondition	Messenger	DemoDungeonB1F	B1F	(3, 12)
8	Exit	0	NoCondition	ExitPosition	DemoDungeonB1F	B1F	(0, 0)
9	Hint	0	NoCondition	Messenger	DemoDungeonB1F	B1F	(3, 4)
10	Locked door silver	0	NoCondition	LockedDoor	DemoDungeonB2F	B2F	(9, 11)
11	locked door gold	0	NoCondition	LockedDoor	DemoDungeonB2F	B2F	(0, 4)
12	Silver key hint	0	NoCondition	Messenger	DemoDungeonB1F	B1F	(6, 13)
13	Keep Out	0	NoCondition	Messenger	DemoDungeonB1F	B1F	(12, 12)
14	Get Gold Key	0	NoCondition	Treasure	DemoDungeonB2F	B2F	(11, 11)

EventMappingViewer shows relations between FloorMapMasterData and EventMasterData. EventMasterData is related to some positions on the map, so this view shows them as a list.

How to open EventMappingViewer

Open EventMappingViewer by pressing [**Check Event Mapping**] button on MapEditor window.

Viewer description

EventMappingViewer highlights events that related to the map editing in MapEditor.

Field	Description
Event ID	ID of events.
Event Name	Name of the event.
Event Index	Index of EventParts.
Start Condition	Condition to start the event.
Event Category	Category of the event.
Map File Name	Name of FloorMapMasterData file.
Map Name	Name of the floor.
Position	Position of the event.

How to close the window

To close this window, click the close button on the window. Or by clicking other windows, this window closes too.

Chapter 4

Master Data

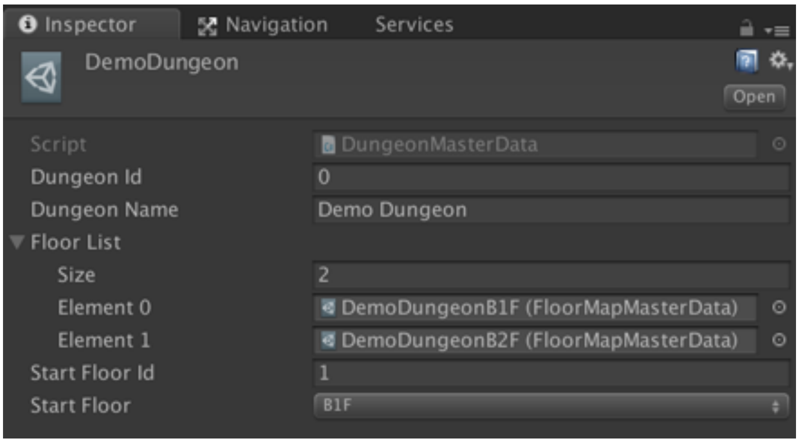
DungeonMasterData

DungeonMasterData is data that holds some FloorMapMasterData and ID of start floor in the dungeon. After creating some FloorMapMasterData, you should create DungeonMasterData and set this to GameController.

How to create

Create DungeonMasterData from [**Assets/Create/Ariadne/DungeonData**] on the menu bar.

Set FloorMapMasterData



Script	DungeonMasterData
Dungeon Id	0
Dungeon Name	Demo Dungeon
Floor List	
Size	2
Element 0	DemoDungeonB1F (FloorMapMasterData)
Element 1	DemoDungeonB2F (FloorMapMasterData)
Start Floor Id	1
Start Floor	B1F

Set FloorMapMasterData to DungeonMasterData via inspector window.

You can specify the "**Start Floor Id**" by using the selector.

Attention: If there are some FloorMapMasterData which have the same floor ID, the selector shows only one of them.

So you have to specify different IDs for those floors.

DungeonPartsMasterData

DungeonPartsMasterData is a set of dungeon parts prefabs. Scripts that draw dungeons instantiate them at runtime.

How to create

Create DungeonPartsMasterData from [**Assets/Create/Ariadne/DungeonPartsData**] on the menu bar.

Set dungeon parts prefabs

Default prefabs are placed at [**Assets/Ariadne/Resources/Prefabs**] folder.

Dungeon drawer scripts calculate the unit size of the dungeon based on the size of wall prefab.

When you change the size of wall prefab, you should check the value of transform of the Player object. See also Player object.

Field	Description
Wall Obj	Set a wall parts prefab.
Ground Obj	Set a ground parts prefab.
Ceiling Obj	Set a ceiling parts prefab.
Door Obj	Set a door parts prefab.
Locked Door Obj	Set a locked door parts prefab.
Upstairs Obj	Set a upstairs parts prefab.
Downstairs Obj	Set a downstairs parts prefab.
Treasure Obj	Set a treasure box parts prefab.
Messenger Object	Set messenger parts prefabs. You can set some messenger parts to instantiate. Specify which parts to instantiate in MapEditor.
Pillar Obj	Set a pillar parts prefab.
Wall With Torch Obj	Set a wall with torch parts prefab.

FloorMapMasterData

FloorMapMasterData holds definitions of each map. Dungeon parts are instantiated based on this data.

How to create

Create FloorMapMasterData from **MapEditor**. To avoid misconfiguration, editing this data directly via Inspector window is not recommended.

Data structure

FloorMapMasterData

Field	Description
Floor ID	ID of the floor. Don't duplicate with other floors in the dungeon.
Floor Name	Name of the floor.
Dungeon Parts	Parts set o the floor.
Floor Size Horizontal	The horizontal size of the floor.
Floor Size Vertical	The vertical size of the floor.
Entrance Pos	Entrance position of the floor. Ariadne uses this value when this floor is the first floor of the dungeon.
Entering Dir	Entering direction of the floor. Ariadne uses this value when this floor is the first floor of the dungeon.
Map Info	Settings of each position in the floor.

MapInfo

Field	Description
Event ID	ID of the event.
Map Attr	Map attribute of the position.
Object Front	Front direction of the object that corresponds to the map attribute.
Messenger Type	Messenger object to instantiate on the position.

EventMasterData

EventMasterData holds information about events.

How to create

Create EventMasterData from **EventEditor**. To avoid misconfiguration, editing this data directly via Inspector window is not recommended.

Data structure

EventMasterData

Field	Description
Event ID	ID of the event.
Event Name	Name of the event.
Event Parts	List of EventParts.

About EventParts information, see EventParts Setting.

ItemMasterData

ItemMasterData stores an ID of the item and a key type of the door. Ariadne doesn't treat other parameters such as a value of healing like a portion.

How to create

Create ItemMasterData from [**Assets/Create/Ariadne/ItemData**] on the menu bar.

Data structure

ItemMasterData

Field	Description
Item Type	Indicates this item is key or not.
Item Id	ID of the item.
Item Name	Name of the item.
Door Key Type	Corresponding key type of the door.

Chapter 5

Data Management

PlayerPosition

PlayerPosition is a static class to store information about the player position.

Managing data

Field	Description
playerPos	Current position of the player.
playerPosPre	Previous position of the player.
direction	Current direction of the player.
currentDungeonId	Dungeon ID that player is exploring.
currentFloorId	Floor ID that player is exploring.

ItemManager

ItemManager is a static class to management ItemData. ItemManager stores the dictionary of holding items. And ItemManager has methods for checking event conditions for items and money.

Managing data

Field	Description
holdItemDict	The dictionary for item IDs and values.
money	Amount of money the player has.

FlagManager

FlagManager is a static class to management event flags. This class stores the dictionary of executed flags in EventParts. And FlagManager has methods for checking conditions of event flags.

Managing data

Field	Description
eventFlagDict	The dictionary for event executed flags and these states.

TraverseManager

TraverseManager is a static class to management traverse data. The traverse data stores whether the player traversed the position or not. Ariadne identifies the traverse data by dungeon id, floor id and the position on the floor.

MapHall object in the scene uses traverse data to draw a map. See also MapHall.

Managing data

Field	Description
traverseList	The list of TraverseData.

TraverseData

Field	Description
dungeonId	ID of the dungeon.
traverseDict	The dictionary of traverse data. The key string is generated by floor ID and the position.

Chapter 6

Scene Objects

Here are GameObjects that the scene requires. Required objects are placed in [**Ariadne/Resources/Prefabs/SceneObjects**] folder and [**Ariadne/Resources/Prefabs/SceneObjects/CanvasParts**] folder as prefabs.

GameController

This object has components of settings and controllers.

Components

DungeonSettings

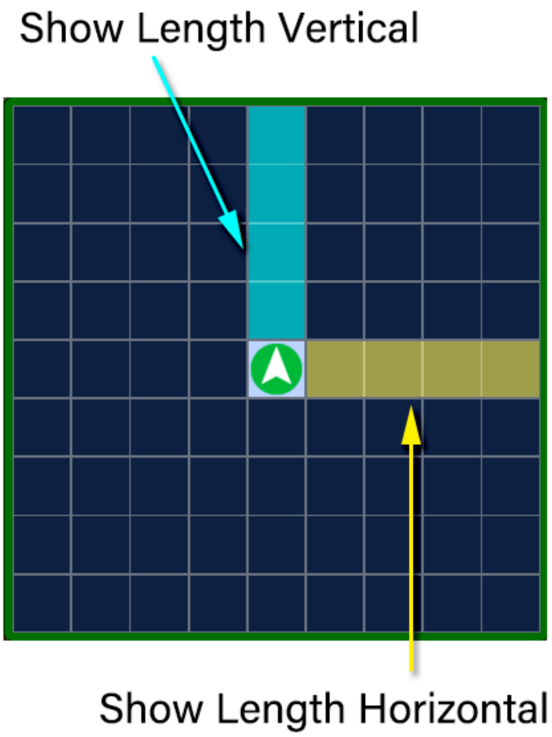
This component stores setting of the dungeon.

Field	Description
Dungeon Data	Set the dungeon data that the scene uses.
Is Draw Outside Wall	The setting whether draw outside walls or not.
Outside Wall Size	Ariadne instantiates walls that are outside of the floor. The width of walls is according to this value.

MapShowingSettings

This component stores setting of the uGUI map.

Field	Description
Show Length Horizontal	The number of horizontal grid from the player's position to the end of map.
Show Length Vertical	The number of vertical grid from the player's position to the end of map.
Smoothness	How many frames Ariadne spend when updates map uGUIs. As this value lower, updating become more smooth. But there is a possibility that it will be a performance cost.
Grid Line Width	The width of grid lines.



In this example, both of "Show Length Horizontal" and "Show Length Vertical" are 4.

DrawManager

Manager script for drawing dungeons and the uGUI map. This script sends messages to scripts that instantiate dungeon parts.

Field	Description
Is Draw Map	The setting whether draw uGUI map on the screen or not.

MoveController

Manager script of the movement of the player. This script checks the existence of an event on the position too.

Field	Description
Move Wait	Waiting time for the movement of the player.
Is UGUI Button Used	The setting whether show the uGUI controller on the screen or not.
Ok Button Parent	The reference to the OK button.
Arrow Button Parent	The reference to the Arrow button.

FadeManager

Manager script of the fading on the screen.

Field	Description
Fade Mask Panel	The reference to the FadeMaskPanel prefab.
Map Mask Panel	The reference to the MapMaskPanel prefab.
Fade Time	Fading time of the screen.
Key Wait Window	The reference to the KeyWaitWindow prefab.
Key Wait Fade Time	Fading time of the key wait window.
Msg Window	The reference to the MapMaskPanel prefab.

EnterDungeonManager

Processes of entering the dungeon.

Field	Description
Enter Button Obj	The reference to the EnterDungeonButton prefab.

EventProcessor

Manager class of processing events.

AriadneEventStrategyFactory

Factory class of each event. This class appends the event script to GameController.

CanvasParts

MapParent

Parent parts of uGUI map parts.

MapBackground

The background parts of the map. You can change the color of the background in Image component.

MapBase

The base parts of the map. You can change the color of the base in Image component.

MapHall

The hallway parts of the map. Ariadne draws the color of the hallway on this object. Specify the color in MapTraversedColor material.

Field	Description
Is Auto Mapping	Set whether draw hallways based on TraverseData or not. If this setting is false, all hallways are visible regardless of TraverseData.

MapIcon

Showing icon parts of the map. Ariadne draws icons on this object. Specify icons that correspond to map attributes. Default icons are placed on [**Assets/Ariadne/Resources/MapData/icon**] folder.

MapGrid

Grid lines parts of the map. Specify the grid color in MapGridColor material.

MapShowingSettings defines the width of the grid line.

PlayerPointer

The pointer icon of the player. You can change this icon at Image component.

MapMaskPanel

The mask panel of the map on uGUI. In this demo scene, MapMaskPanel hides the map when the player is out of the dungeon.

KeyWaitWindow

Information window of waiting for the keypress. You can change the message of this window in the Text component of **KeyWaitMsg**.

MessageWindow

The message window for events. EventProcessor uses this window for showing messages.

OKButtonParent

The parent parts of the OK button. To change the label of this button, edit the Text component of **OKButtonLabel**.

ArrowButtonParent

The parent parts of arrow buttons. To change labels of these buttons, edit the Text component of each child object.

EnterDungeonButton

The button to enter the dungeon. This button is **not necessarily required** for your scene because processes after exiting the dungeon may be different from this demo scene. To integrate entering and exiting dungeon process to your scene, see Enter & Exit the dungeon.

FadeMaskPanel

The mask panel of the screen.

Player

The player objects that include a camera and lights. MoveController script controls this object by the input.

This object defines the height of the camera. When you change sizes of prefabs, you should change **y-value** of transform in the Player object.

The Player object has light objects for lighting the dungeon.

DungeonParent

The parent parts of the dungeon parts.

GroundParent

The parent parts of the ground object of the dungeon.

WallParent

The parent parts of the wall objects of the dungeon.

CeilingParent

The parent parts of the ceiling object of the dungeon.

Field	Description
Is Draw Ceiling	The setting whether drawing the ceiling or not.

Chapter 7

Integration

Enter & Exit the dungeon

To integrate Ariadne into your game, some settings are required.

To load the dungeon scene from scripts, add the scene to "Scenes In Build" in BuildSettings.

If the player exits the dungeon, MoveController sends OnExitDungeon() message to the GameController via IExitDungeon interface. So you should add a script to the GameController to catches the message.

After adding your script, remove EnterDungeonManager script from the GameController to avoid showing the default button.

Encounter with enemies

To integrate an encounter system for 3D dungeon RPG, add your code that determines meeting with enemies to MoveController.

One of the good places to add the code is PostMove() method in MoveController.

MoveController.cs

```
// When no event has been executed, set true to the move flag.
if (!isExecuted && !isForwardEventExecuted){
    // *** Add your check encounter code ***
    isMoveOk = true;
    fadeManager.InitializeWaitFlags();
}
```

When you add codes, control "isMoveOk" flag too. MoveController checks whether the player can move or not by checking this flag.