



POWER BOOKS

User Guide



Content

Overview	2
Getting Started.....	3
The Power Book Inspector Window	4
Simple Book Creator	5
Custom Pages	6
Demo Scenes Content.....	7
Power Books Scripting API	8
Events	15
Events (Inspector).....	16
Book Covers / Titles.....	17

Overview

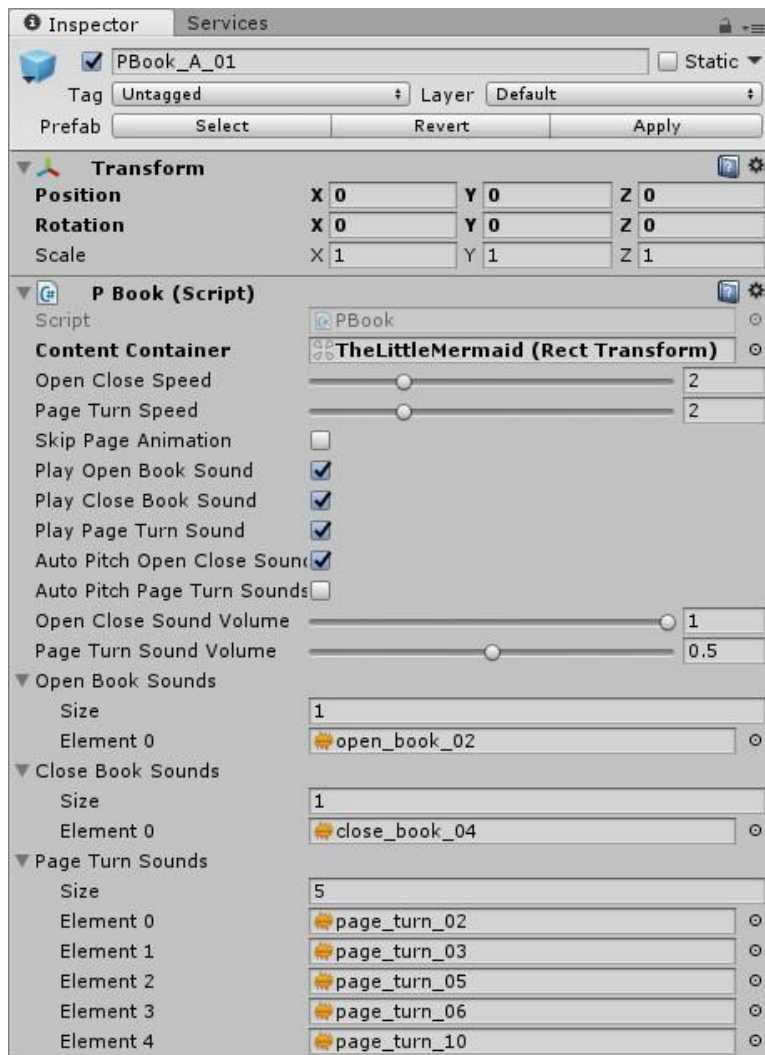
Thank you for purchasing and using Power Books! Power Books lets you easily create animated, readable books for your game. This documentation was designed to provide you with a basic overview of the features and functionality of the Power Books package.

Getting Started

Step by step guide to get your first readable power book into a scene:

- 1) Drag a **Power Book** prefab from the project folder *(Assets/PowerBooks/Prefabs/PowerBooks/)* into the scene.
- 2) Drag an example content prefab from the project folder *(Assets/PowerBooks/Prefabs/ExampleContent/)* into the scene.
- 3) Select the **Power Book** game object and assign the example content game object to the **Content Container** slot in the inspector.
- 4) Drag the **ExampleUIController** prefab from the project folder *(Assets/PowerBooks/DemoScene/Prefabs/BookController/)* into the scene.
- 5) Select the **ExampleUIController** game object and assign the **Power Book** game object to the **P Book** slot in the inspector.
- 6) Position the camera so that you can see the Power Book. Press play.

The Power Book Inspector Window



Most of the settings in the inspector are self-explanatory. The rest of this page discusses the settings that are not entirely self-explanatory:

Content Container: The game object that holds the book content (pages). Multiple Power Books can share the same content container, but only one of them can be opened at a time.

Sounds: If the arrays holds more than one sound, sounds will be played randomly.

Simple Book Creator

Step by step guide to create a simple book from a text asset:

- 1) Open the SimpleBookCreator scene (*Assets/PowerBooks/BookCreatorScene/*).
- 2) Select the **SimpleBookCreator** game object and assign an example text asset (*Assets/PowerBooks/BookCreatorScene/TextFiles/*) to the **Text File** slot in the inspector.
- 3) Type a prefab name into the **Prefab Name** field in the inspector.
- 4) Press play.
- 5) Press the **Save Content** button to save the generated book content. The book content will be saved into the following folder: *Assets/PowerBooks/BookCreatorScene/SavedBookContent/*.

Custom Pages

Step by step guide to create custom pages:

- 1) Drag the **CustomBookTemplate** prefab from the project folder (*Assets/PowerBooks/Prefabs/CustomBookTemplate/*) into the scene.
- 2) Click the **CustomBookTemplate** object's drop-down arrow to show its child-object's. The **LeftPageTemplate** and **RightPageTemplate** objects holds the editable page objects.
- 3) Edit the pages by adding Unity UI component to them (Text, Images, Buttons, etc.).
- 4) Drag all edited pages into the empty **ContentContainer** object and sort them as you want.
- 5) Rename the **ContentContainer** game object and save it as a new prefab.

Demo Scenes Content

DemoScene

Provides an overview of the standard functions of the Power Book package.

DemoSceneDragController

Drag the mouse to use a Power Book (Drag to: open book, close book, next page, previous page).

DemoSceneEvents

Demonstrates event handling. Any script can register to the following events:

DemoSceneFPSController

Aim / Look at a book and interact with it.

DemoSceneJournal

Aim / Look at a book and open it in "Journal Mode".

Power Books Scripting API

OpenBook

`public void OpenBook()`

Description

Opens the book.

CloseBook

`public void CloseBook()`

Description

Closes the book.

NextPage

`public void NextPage()`

Description

Turns one page forward.

(If the **skipPageAnimation** flag is set to **true**, page animation will be skipped.)

PrevPage

`public void PrevPage()`

Description

Turns one page backward.

(If the **skipPageAnimation** flag is set to **true**, page animation will be skipped.)

GotoPage

public void GotoPage(int pageNumber, float speed)

Parameter

pageNumber	The page number to turn to.
speed	Page turn animation speed.

Description

Turns to a page.

The parameter **speed** is clamped to the range [1, 50].

This function will do nothing if the current **BookState** is not set to **BookState.Open**

GotoFirstPage

public void GotoFirstPage(float speed)

Parameter

speed	Page turn animation speed.
-------	----------------------------

Description

Turns to the first page.

The parameter **speed** is clamped to the range [1, 50].

This function will do nothing if the current **BookState** is not set to **BookState.Open**

GotoLastPage

public void GotoLastPage(float speed)

Parameter

speed	Page turn animation speed.
-------	----------------------------

Description

Turns to the last page.

The parameter **speed** is clamped to the range [1, 50].

This function will do nothing if the current **BookState** is not set to **BookState.Open**

JumpToPage

public void JumpToPage(int pageNumber, bool playSound)

Parameter

pageNumber	The page number to turn to.
playSound	Set this parameter to false to disable page turn sound.

Description

Jumps to a page, without playing page turn animation.

JumpToFirstPage

public void JumpToFirstPage(bool playSound)

Parameter

playSound	Set this parameter to false to disable page turn sound.
-----------	---

Description

Jumps to the first page, without playing page turn animation.

JumpToLastPage

```
public void JumpToLastPage(bool playSound)
```

Parameter

playSound	Set this parameter to false to disable page turn sound.
------------------	---

Description

Jumps to the last page, without playing page turn animation.

SetOpenCloseSpeed

```
public void SetOpenCloseSpeed(float speed)
```

Parameter

speed	Speed for open/close animation.
--------------	---------------------------------

Description

Sets the speed for the open/close animation.

The parameter **speed** is clamped to the range [1, 5].

SetPageTurnSpeed

```
public void SetPageTurnSpeed (float speed)
```

Parameter

speed	Speed for page turn animation.
--------------	--------------------------------

Description

Sets the speed for the page turn animation.

The parameter **speed** is clamped to the range [1, 5].

SetSkipPageAnimation

public void SetSkipPageAnimation (bool skipAnimation)

Parameter

skipAnimation	Sets the skipPageAnimation flag.
---------------	----------------------------------

Description

Sets the skipPageAnimation flag.

SetOpenCloseSoundVolume

public void SetOpenCloseSoundVolume (float v)

Parameter

v	The volume for open/close sounds.
---	-----------------------------------

Description

Sets the volume for open/close sounds.

SetPageTurnSoundVolume

public void SetPageTurnSoundVolume (float v)

Parameter

v	The volume for page turn sounds.
---	----------------------------------

Description

Sets the volume for page turn sounds.

GetCurrentPageIndex

public int GetCurrentPageIndex ()

Returns

int The current page index (e.g. 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, ...)

Description

Returns the current page index.

GetPageCount

public int GetPageCount ()

Returns

int The page count of the book.

Description

Returns the page count of the book.

IsFirstPage

public bool IsFirstPage ()

Returns

bool true if the current page is the first page.

Description

Returns true if the current page is the first page.

IsLastPage

public bool IsLastPage ()

Returns

bool true if the current page is the last page.

Description

Returns true if the current page is the last page.

GetBookState

public BookState GetBookState ()

Returns

BookState The current state of the book.

Description

Returns the current state of the book.

(public enum BookState {CLOSED, OPENBOOK, OPEN, CLOSEBOOK, NEXTPAGE, PREVPAGE})

GetAudioSource

public AudioSource GetAudioSource ()

Returns

AudioSource The AudioSource component of the power book.

Description

Returns the AudioSource component of the power book.

GetBookAnimator

public Animator GetBookAnimator()

Returns

Animator The Animator component of the animated book game object of the power book.

Description

Returns the Animator component of the animated book game object of the power book.

GetPageAnimator

public Animator GetPageAnimator()

Returns

Animator The Animator component of the animated page game object of the power book.

Description

Returns the Animator component of the animated page game object of the power book.

Events

You can subscribe methods to the following events:

`OnBookOpened`

`OnBookWillOpen`

`OnBookClosed`

`OnBookWillClose`

`OnBookFirstPage`

`OnBookTurnToFirstPage`

`OnBookLastPage`

`OnBookTurnToLastPage`

The demo scene “DemoSceneEvents” contains an example of how to use events.

Events (Inspector)

You can use events in the editor/inspector too:



Book Covers / Titles

The package contains 8 designs (and 32 Prefabs) which are suitable for books with a cover/title.

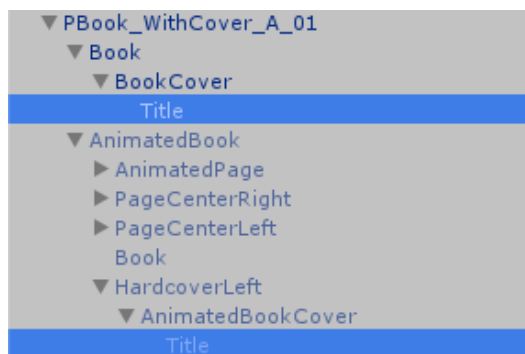
(Assets/PowerBooks/Prefabs/PowerBooks/WithCover/)

These prefabs contain two canvas objects with a text component assigned to them. To edit the title of a book, edit the canvas components under:

PrefabName->Book->BookCover->EditOrAddTextOrImageComponentsHere

and

PrefabName->AnimatedBook->HardcoverLeft->AnimatedBookCover->EditOrAddTextOrImageComponentsHere.



The BookCoverHelper.cs *(Assets/PowerBooks/Scripts/)* script can be used to copy text/image component settings onto the corresponding objects on the second canvas. You can also select both objects and edit both components at the same time.