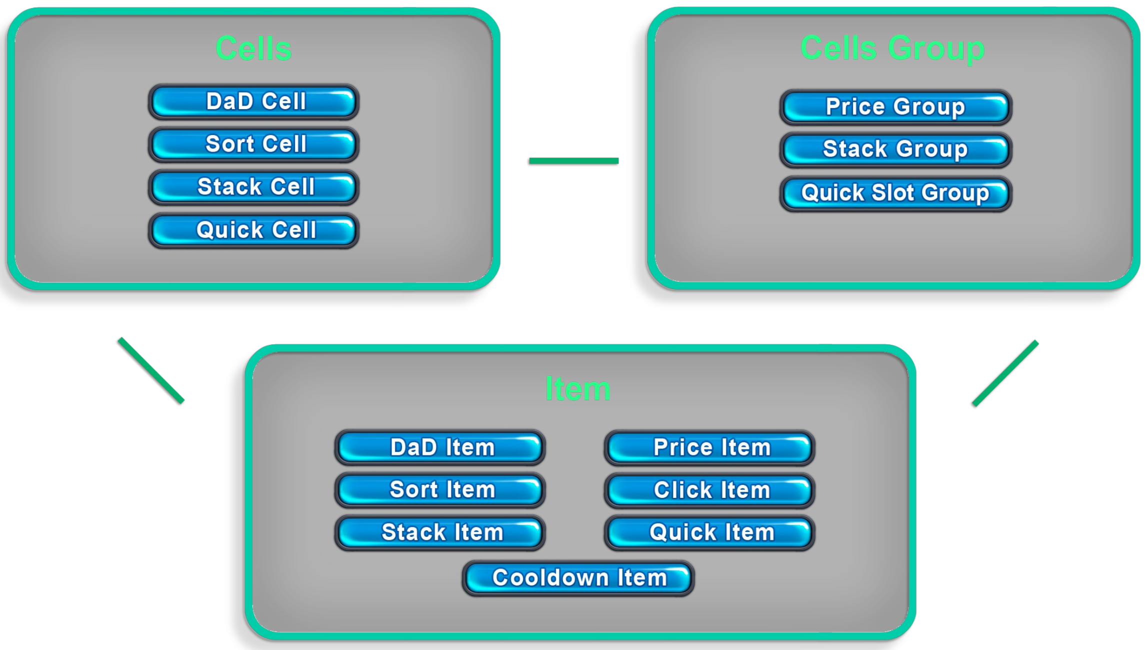
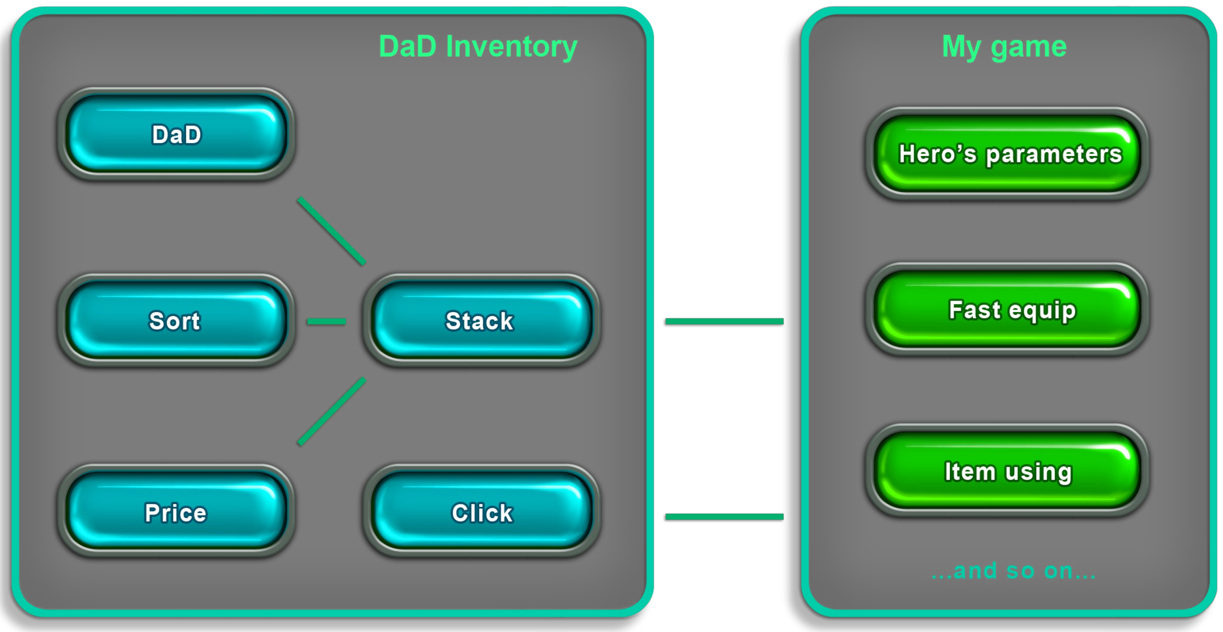
Demo

Run “Demo” scene for asset's features demonstration.

DaD Inventory internal logic



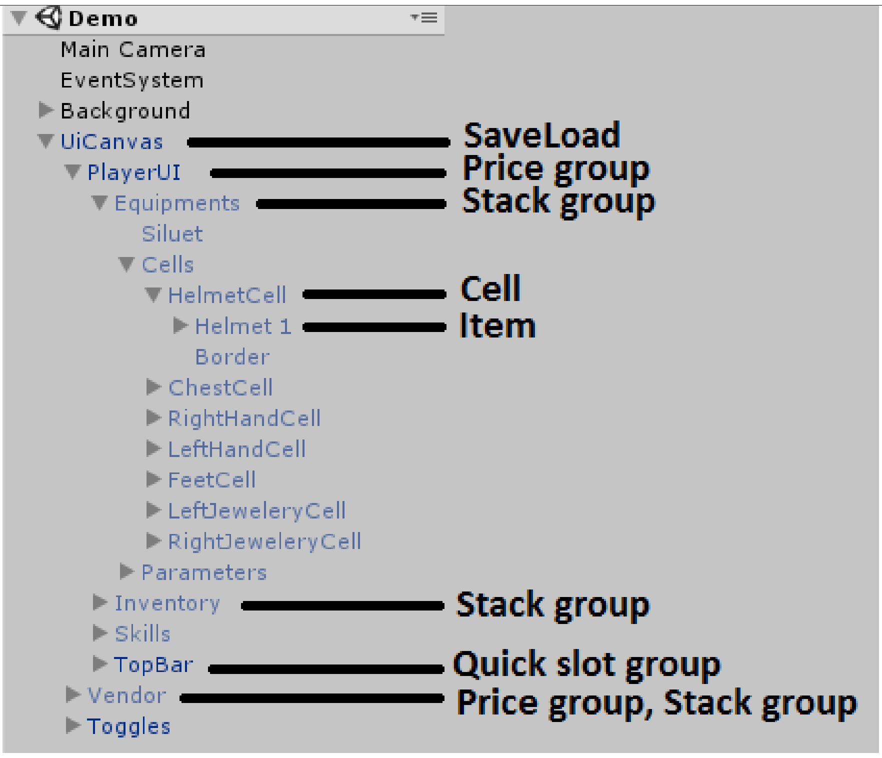
Interaction with the game logic



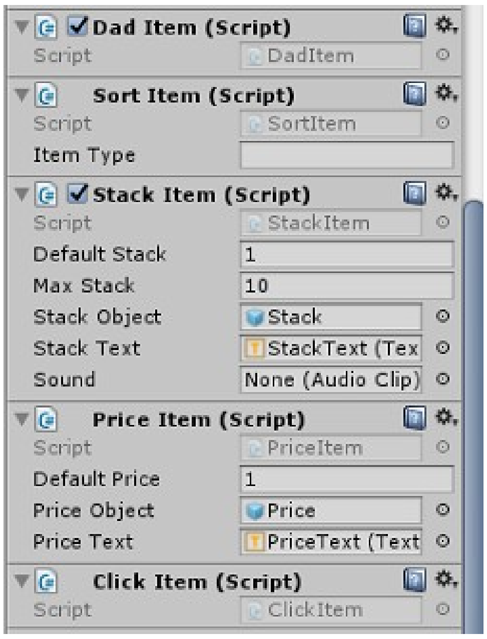
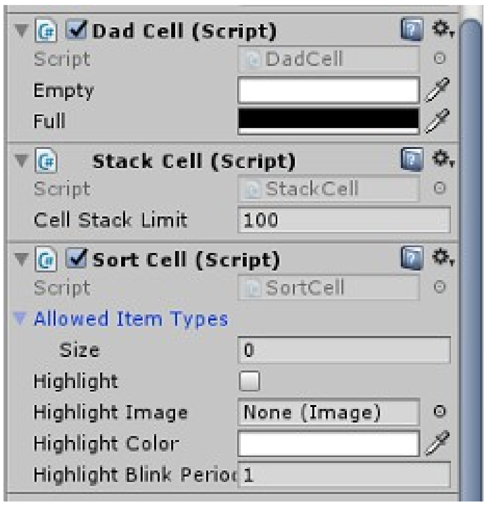
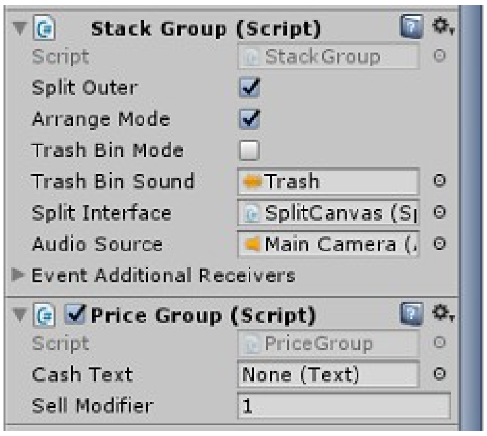
DaD Inventory interaction with the game logic includes:

- Stack items movement events → OnStackGroupEvent
- Double click event → OnItemClick
- Item adding/replacing/removing using StackGroup or StackCell methods

Components

Demo scene


```
▼ Demo
    Main Camera
    EventSystem
  ▶ Background
  ▼ UiCanvas                          SaveLoad
    ▼ PlayerUI                        Price group
      ▼ Equipments                    Stack group
          Siluet
        ▼ Cells
          ▼ HelmetCell                Cell
            ▶ Helmet 1                Item
              Border
          ▶ ChestCell
          ▶ RightHandCell
          ▶ LeftHandCell
          ▶ FeetCell
          ▶ LeftJeweleryCell
          ▶ RightJeweleryCell
        ▶ Parameters
      ▶ Inventory                     Stack group
      ▶ Skills
      ▶ TopBar                        Quick slot group
    ▶ Vendor                          Price group, Stack group
    ▶ Toggles
```

Item	Cell	Group
		

Item

```
▼ ☑ Dad Item (Script)
  Script              DadItem
▼ Sort Item (Script)
  Script              SortItem
  Item Type
▼ ☑ Stack Item (Script)
  Script              StackItem
  Default Stack       1
  Max Stack           10
  Stack Object        Stack
  Stack Text          StackText (Tex
  Sound               None (Audio Clip)
▼ Price Item (Script)
  Script              PriceItem
  Default Price       1
  Price Object        Price
  Price Text          PriceText (Text
▼ Click Item (Script)
  Script              ClickItem
```

Cell

```
▼ ☑ Dad Cell (Script)
  Script              DadCell
  Empty
  Full
▼ Stack Cell (Script)
  Script              StackCell
  Cell Stack Limit    100
▼ ☑ Sort Cell (Script)
  Script              SortCell
▼ Allowed Item Types
    Size              0
  Highlight           ☐
  Highlight Image     None (Image)
  Highlight Color
  Highlight Blink Perioc 1
```

Group

```
▼ Stack Group (Script)
  Script              StackGroup
  Split Outer         ☑
  Arrange Mode        ☑
  Trash Bin Mode      ☐
  Trash Bin Sound     Trash
  Split Interface     SplitCanvas (S|
  Audio Source        Main Camera (.
▶ Event Additional Receivers
▼ ☑ Price Group (Script)
  Script              PriceGroup
  Cash Text           None (Text)
  Sell Modifier       1
```

Items adding and removing

To add item from prefab use **StackGroup.AddItemFromPrefab** or **StackCell.AddItemFromPrefab** (to place item into needed cell, in this case old item will be deleted from this cell).
To place existing on scene item into needed group use **StackGroup.AddExistingItem**.
If item was placed on scene with any other way (including editor mode) you need to call **StackItem.Init** method. Otherwise behavior may not be expected if item was placed in inactive cell, because of Awake function will not be called in this case. Code example on scene start:
foreach (StackItem stackItem in AccessUtility.FindObjectsOfType<StackItem>()) {stackItem.Init();}
To remove item use **StackGroup.RemoveItem**.
To add new skills use **DadCell.AddItemFromPrefab** because skills have no StackItem component.

Stack events

To organize game logic there is **OnStackGroupEvent** called on every item's movement. That event called on source group and on destination group's game objects and on all their parent game objects. Also you may specify additional event receivers in inspector window of stack group.
OnStackGroupEvent has **StackGroupEventDescriptor** that include needed information about event:
desc.**sourceGroup** – stack group from which item was dragged;
desc.**sourceCell** – stack cell from which item was dragged;
desc.**destinationGroup** – stack group into which item was dropped;
desc.**destinationCells** – list with cells into which item was placed (distributed);

To get item from needed cell use **StackCell.GetStackItem**.

To receive stack events place this handler in your code:
public void OnStackGroupEvent(StackGroup.StackGroupEventDescriptor desc)
{ }

DaD Inventory modules description

DadCell, DadItem

This is the base of drag and drop interface, that operates two separate units:

- Cell
- Item, that may be dragged between cells

SortCell, SortItem

These scripts allow to specify item's type (sort). The cell also may be specified to contain only chosen item types. If dropped item's type does not match to cell's type, the item transaction will be forbidden.

PriceItem, PriceGroup

PriceItem allows to set the price for item. When items dropped from one PriceGroup to another, the price amount will be automatically calculated. If there is enough cash, it will be decreased on the price amount. Otherwise the transaction will be forbidden. If the game object that displays the amount of available cash is not setup, then the cash amount is considered infinite (example: trader).

StackItem, StackCell, StackGroup

StackItem and StackCell allows to put items with the same name together. The stack limit can be set for the specific item as well as for specific cell. StackGroup determines the logic of items distribution inside the group as well as between different groups. Minimal set of configurations allows to get various behavior at items transactions and to create such groups as player's inventory, trader, trash bin. Also, there is items splitting is supported.
To get **infinity** stack amount set stack number as maximum int value (2147483647) for StackCell and for StackItem.
ATTENTION: when you make you game scene please be sure that StackItem's stack amount not greater than StackCell's stack limit. Otherwise it may case unexpected asset's behavior.

ClickItem

It handles double click on item and notify the game's logic. That allows to make such features as item using, fast items equipment of fast items selling.

CooldownItem

This script helps to implement cooldown on item using. The cooldown must be triggered from your game logic (for example on item or skill double click).

QuickItem, QuickCell, QuickSlotGroup

This scripts make fake cells for items and skills. Click on this cell will equal to double click on original item or skill. QuickSlotGroup has list of permitted source groups for quick items (Equipments, Inventory and so on).

Tooltip

Shows the item's “in game” information on mouse press or mouse over.

Pick up items

DummyItemPickUp with **DummyInventoryControl** contain an example how to realize item's pick up using DaD Inventory. Prefabs folder contains prefabs for 2D and 3D pick up items. Click on blinking bottle on demo scene to apply this logic.

Inspector hints

Place cursor over the name of variable in inspector window to get the hint, which provides additional information and helps to configure asset for your game.

Items splitting interface

When the player split or move items between different groups, special interface occurs. It allows to choose needed number of items. If there are different price groups, items prices will be shown in interface.

ATTENTION: splitting interface must block the whole screen. Otherwise it may case unexpected asset's behavior (for example, player try to use or move other item while splitting interface is active).

Asset integration

Use Window->DaDI to add the whole user interface or parts of it.

All item and skill prefabs place into corresponding Resources folder (to make possible inbuilt saving/loading).

When player make double click on item, the **ClickItem** send notification to all parent game objects.

There are few public methods in **PriceGroup** that may be used in your game: **ShowPrices**, **HidePrices**.

“Demo” scene helps you to better understand the asset work.

Feedback

Don't forget to rate asset in the Asset Store and leave your comments, it gives us a chance to make it better. Feel free to contact us, using the contacts from the asset page.