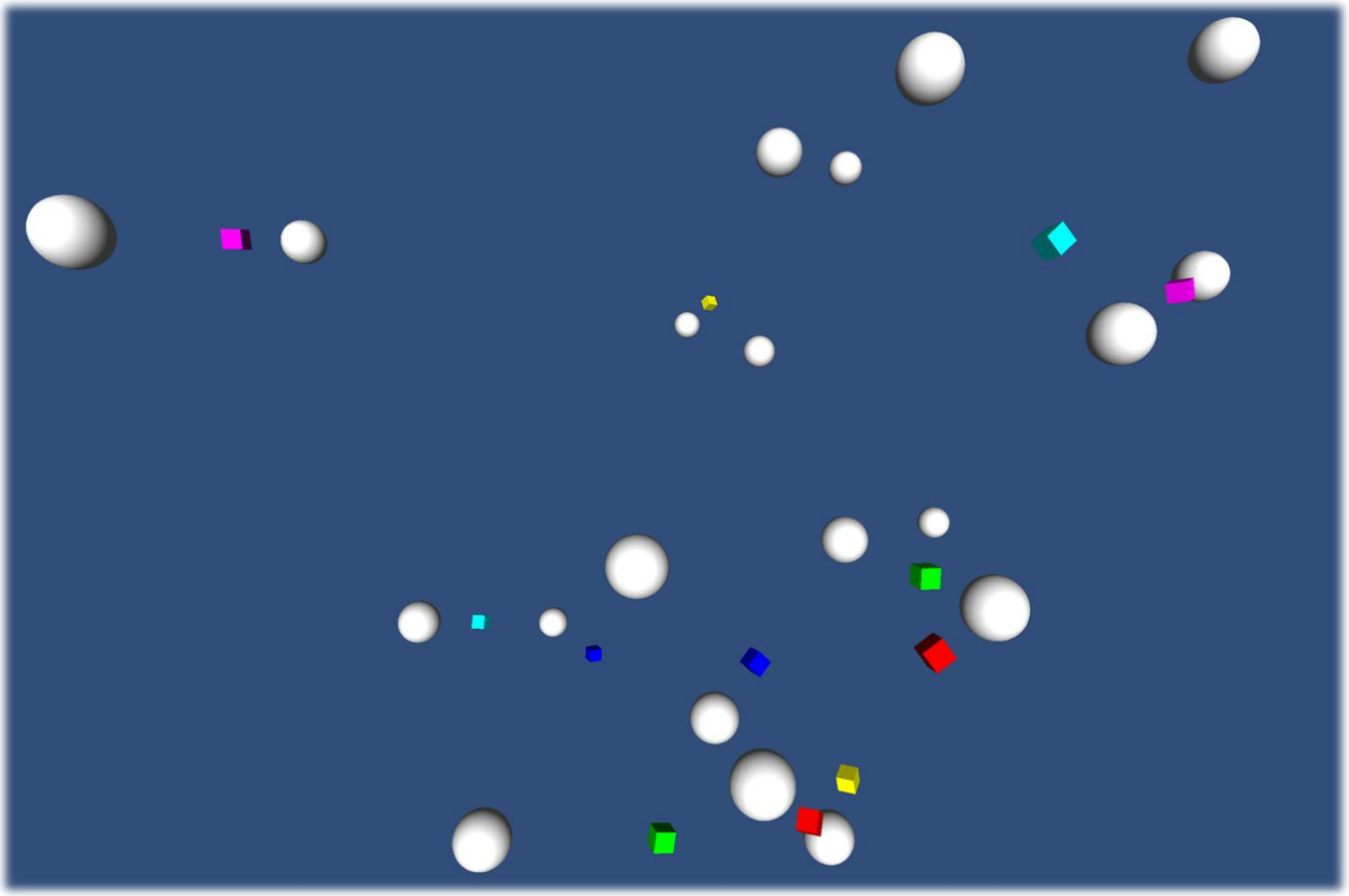
TradeSys

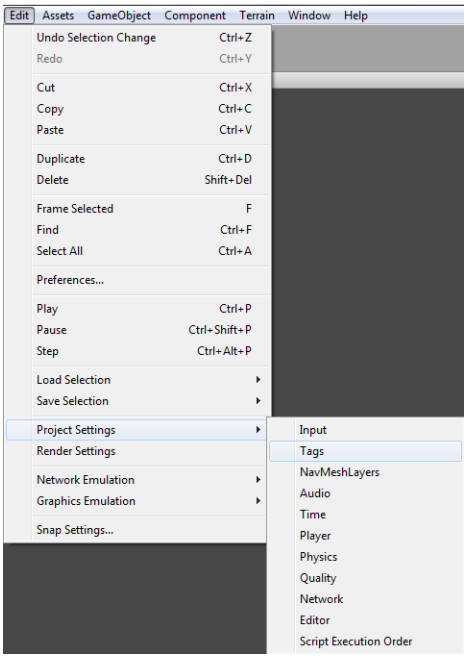


A trading and manufacturing system

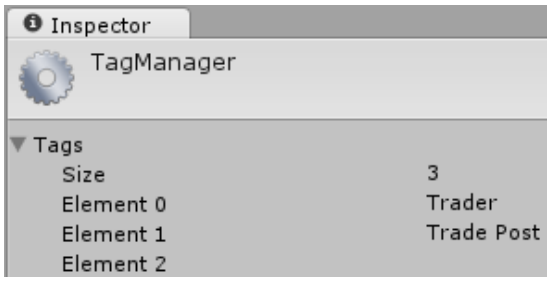
Version 1.1

Tradesys requires some setup before you can start sending out your traders, but fortunately this is very simple.

Setting up TradeSys:



1. New tags need to be added to your game. The tags need to be called ‘Trader’ and ‘Trade Post’. Tag adding can be found by Edit > Project Settings > Tags. Here, there is a dropdown of Tags, and the new tags are added here.



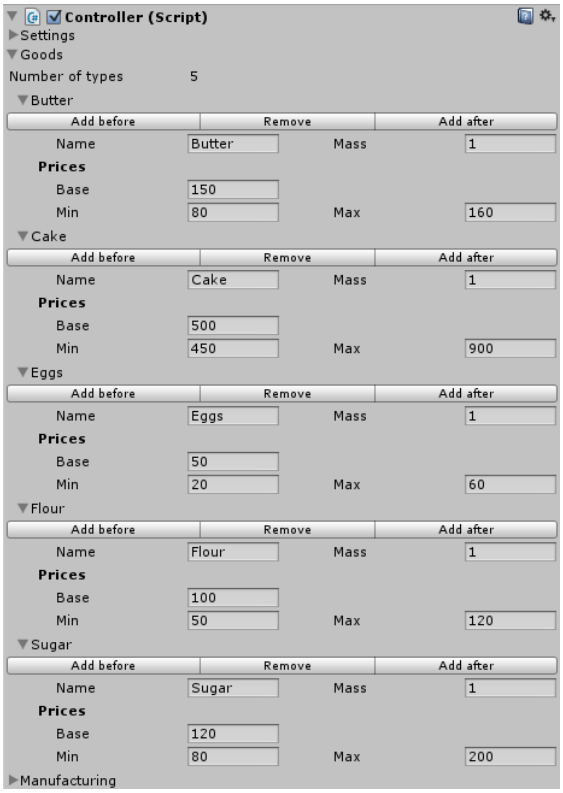
2. An empty GameObject needs to be created and called ‘Controller’. Add the controller script to this object. This will handle all of the possible trades that you set up.
3. Now it is time to setup the Trade Post. Either add a new GameObject (this time not empty, otherwise you won’t be able to see it!), or on a previously created object that you want to be your Trade Post, set the tag to Trade Post. Add the TradePost script to this object; repeat this step for all of your posts.

If you are having certain traders within your game going from post to post, follow step 4A, if you want the traders to be created at a post and then destroyed when they get to their destination, follow step 4B.

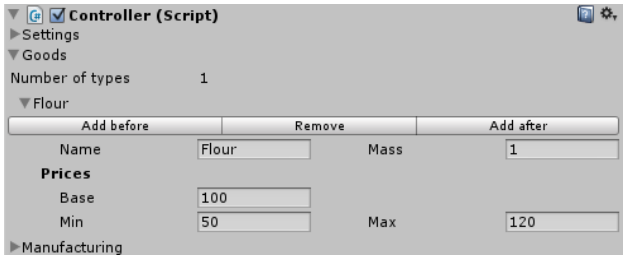
4A. The Traders need to be sorted now. Once again, either create a new object, or use a previously added object. Change the tag to Trader and make sure that the trader is inside one of your Trade Posts (this won’t affect the game, it just looks better). Add the Trader script, and change the Target Post variable to the Trade Post where they currently are. This is so the controller knows which post to send the trader to. The stop time value is how long the trader has to stop at each post for before it is allowed to leave, it helps make it slightly more realistic, and the cargo space is the maximum mass of cargo it can carry. The speed multiplier is part of demo code, which would be removed to work with your own AI. Repeat this step for each of your traders.

4B. In controller, there is a group called settings, turn on expendable traders if you want your traders to be created and destroyed. Then, you need to set the maximum number of traders allowed at any one time. Setting this to 0 means that an infinite number are allowed. You also need to set the different traders that can be created, and the controller will randomly select a trader. Pressing ‘Add’ will create a new trader type, where you then select the prefab for the trader. To create the prefab, add the object you want as your trader, and follow step 4A, but the target post does not need to be set because this will be set by the controller. Once this has been done, drag the object from the Hierarchy into the Projects folder, where it will create the prefab. The object in your scene view can now be deleted, and the prefab added to the trader type list. There is also the option to make the traders pause for the time defined in each trader, or leave unchecked if you want the trader to be created and they instantly leave.

Adding goods to trade:



1. On the controller object, under goods, press the 'Add' button. This will create a new item where you can set its name. The mass of each item affects the number that can be carried by the trader. The minimum value for this is 0.000001, which if 1 = 1 ton, then that is 1 gram! The prices also need setting; there are three different values that require setting. The base price is the price used to work out the price at each trade post, and the min and max will limit the prices to within the range specified.

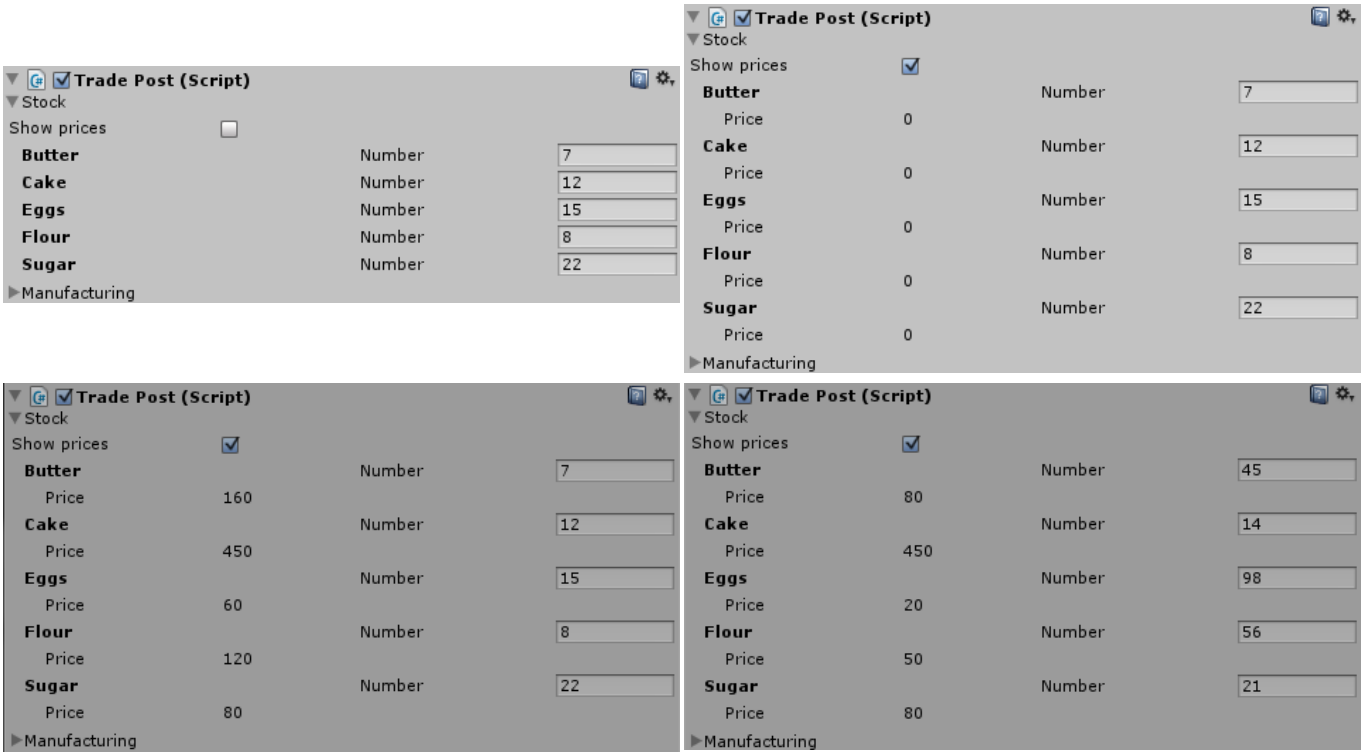


2. Each item can be removed using the 'Remove button', and new items can be added before or after, so it is possible to keep your items in alphabetical order.
3. Manufacturing processes are also set up in the controller, and a new process is made by pressing the 'Add' button. Here, you can set its name, remove it, or set the items in the needing and making sets. In both these groups, it will only need or make items that you have specified in the goods section.

4. Now that you have added the items in your game and the different manufacturing processes, it is now time to sort out the trade posts. Under the settings group, it has the names of each different item that has been set up in controller. To disable a certain item from being traded at that station, just uncheck the box and the item will disappear in the stock group and any manufacturing process that involved that item will disappear. Under the stock group, it will show all of the items that you have set in the controller and have enabled. All you have to do is set how many are found at the post at the beginning of the game. The price cannot be set because the controller sorts this out for you, and will change throughout the game! The toggle is there so that it is possible to see the prices change while you are testing the game. (See pictures below)

5. Under the manufacturing group, if you want the post to follow one of the processes, then check the box. This will enable you to set how long it waits before the process is repeated. It can only manufacture items when it has the required number of each item set in the controller. Steps 4 and 5 need to be repeated for each of your Trade Posts.

You have now successfully set up TradeSys!



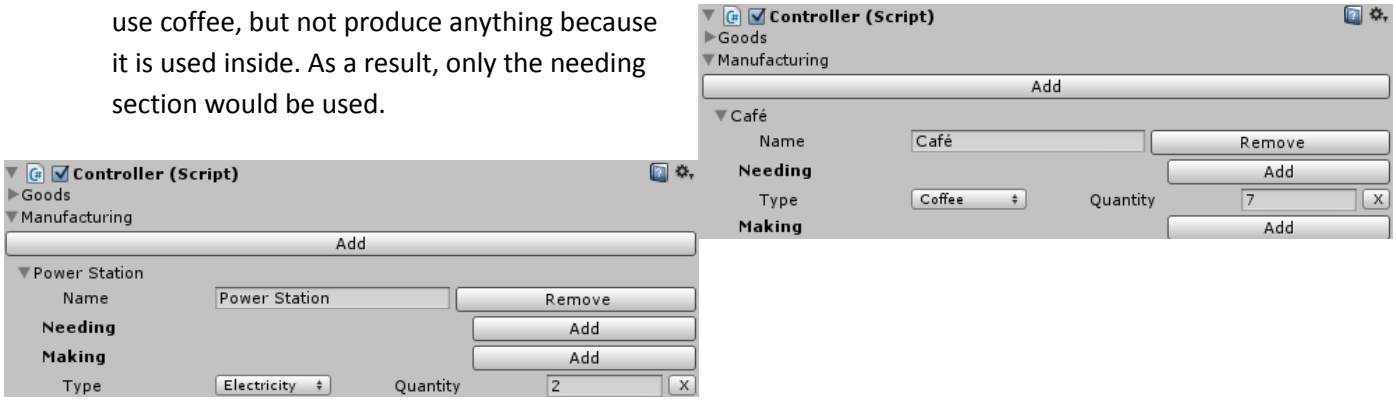
On the left, the numbers for a station that has been set up, the right shows that the prices before the start are 0. Below are the numbers and prices for two stations at the start of the game, once the prices have been generated.

Information: In the trader, the Update() is demo code, added to show trader movement. This would be replaced with your own AI with the target post as the final target destination. When it reaches the destination, it needs to call the AtLocation() method.

Update Instructions: Import the updated assets, click onto any object other than the controller, then the controller. This will make the controller update all the required post information. If you want specific traders and not expendable ones, the cargo size will need setting as the default is 1. Each mass in the controller may also need changing as these will be set to the lowest, 1e-06 (0.000001).

Tips:

- To add a new Trade Post, just create the new object from your script, and call NewPost(times) on the new object. This will set everything up, and will generate a random number for each stock item. Times is an int[] which is required because if the number is > 0, the process will be set to true and will have the time that is specified. The length of the int[] needs to have as many elements as the number of manufacturing processes.
- To add a new Trader, create the new object through the script, call NewTrader(post), where the post is the Trade Post GameObject that the trader starts at.
- The manufacturing processes do not need to have items in needing or making; for example, a café would use coffee, but not produce anything because it is used inside. As a result, only the needing section would be used.



- Equally, a solar power station may only produce electricity and not require inputs, so only electricity on the making would be needed.
- A more industrialised trade post would have quicker manufacturing processes, so reduce the time to create. Less industrialised places would take longer, so would have a higher value.

- It is possible to enable or disable certain manufacturing processes during game play by:

```
nameOfTradePost.GetComponent<TradePost>().manufacture[intOfManufacturing].yesNo = true;
nameOfTradePost.GetComponent<TradePost>().manufacture[intOfManufacturing].seconds = timeToCreate;
```

You will need to change the variables written in bold to the chosen ones.

Change Log

VERSION		CHANGES
V1.0	-	Initial release
V1.1	-	Traders have cargo size and items require mass specified so traders cannot take infinite items.
	-	Traders can now take multiple types of item as long as they all go to the same place.
	-	Traders can be set as expendable, so will be created at required station and destroyed on arrival.
	-	Can disable items at posts so will not be traded or manufactured.
	-	Some UI tweaks.