TradeSys

A trading and manufacturing system
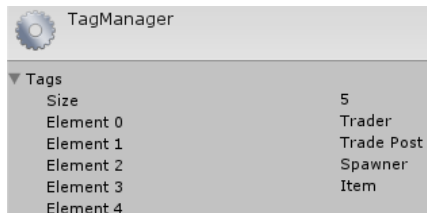
Contents

TradeSys requires some initial setup before the traders can get to work, but fortunately this is very simple.

Initial setup:

1. New tags need to be added. The tags need to be called 'Trader', 'Trade Post', 'Spawner' and 'Item'. To add a tag, go Edit > Project Settings > Tags. Here, there is an array of tags, and you can add in the new ones.

2. Add an empty GameObject, and call this 'Controller'. (GameObject > Create Empty). Add the controller script to this object. This will handle all of the possible trades that can occur in the game.

3. Now it is time to add the Trade Posts. 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 wish to be a Trade Post, set the tag to Trade Post, and add the TradePost script. Repeat this step for all posts. Alternatively, it is now even easier to create all of your Trade Posts in a script. See **useful methods** for more information.

4. If you want your traders to be able to pick up items that have been dropped or spawned, then in controller, under settings, tick 'Allow item pickup'.
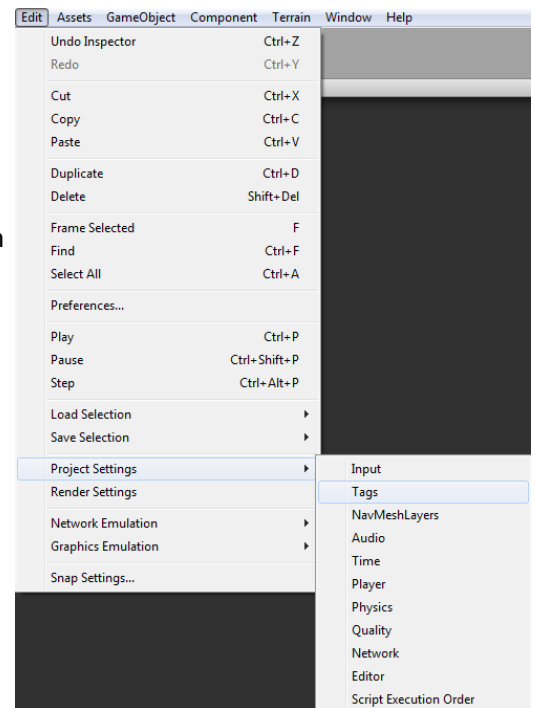
If you are having set traders within your game going from post to post, follow 4A. If you want traders to be created at the post that they are required and then destroyed when they get to their destination, follow 4B.

4A. The Traders are now going to be created. Once again, either create a new GameObject, or use a previously created one. Set the tag to Trader and add the Trader script. Either put the trader near the location of a Trade Post (needs to be <3 units away in order to be found) and press the find post button. This will find a nearby post, and set the target correctly and move it to be at the post. The other option is to set the post that you wish the trader to be at, and press the set location button which will move the trader to the correct location. The other options available to traders is the stop time, which is how long the trader must wait at a Trade Post before it is allowed to leave, making it slightly more realistic. The cargo space is the maximum mass of cargo that can be carried. If you want the traders to be able to pick up dropped or spawned items, then ensure that the 'Allow item pickup' option has been selected in the controller, and tick 'Allow trader collection'. This will then show two more options, the radar distance, which is how far away the trader can see an item and this can be seen in the scene view, and drone stop time. This is how long a trader will stop to pick up an item.
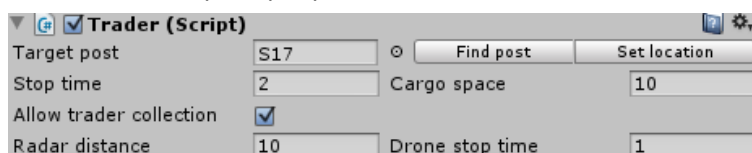
4B. In controller, under settings, there is an option for expendable traders. Enable this so that traders will be created and destroyed. Then, you need to set the maximum number of traders allowed at any one time. Setting this to 0 will mean that there may be an unlimited number of traders. The different types of traders

that can be created need to be set, as the controller will randomly select a trader to create from this list. Pressing 'Add' will create a new trader type, where you can select a prefab for the trader. To create the prefab, add the object that you want as your trader, and follow step 4A to set the trader up. But don't set the target post as this is set by the controller when it is created. Once this has been done, drag the Trader into the Projects folder, where it will create the prefab. The object in the scene view can now be deleted, and the prefab added to the trader type list. In the expendable traders, there is the option to pause before leave. If this is enabled, the trader will pause for the time that has been set in the prefab before it will leave the Trade Post. Leave unchecked if you want them to instantly leave.

Setting up the controller:

1. It is possible to set the update interval. This is the time between each update of pricing at trade posts and possible trades. By decreasing this number, it may be possible to improve performance.

2. Show trade routes is an option where once all of the trade posts have been sorted, when enabled, it will draw lines between posts where it is possible for traders to go between. There is also an option to enable groups. When this is enabled, all of your Trade Posts can belong to different groups, which have been set here. I is possible to add as many groups as you want, remove groups or change the names.

3. In order for the items to display a unit, the units found under settings in the controller needs to be sorted. Press 'Add' to add a new unit, and enter the suffix you wish the unit to have. The min and max values also need to be set. An item will have that unit if the mass is greater than or equal to the min, and less than the max. The final unit needs to have a max value of infinity, so type in 'infinity'. If there are any overlaps, or no units extending to infinity, a warning will show in the inspector, to help with correctly setting up the units.

4. Now goods can be added, under the goods setting in the controller, press the 'Add' button to create a new
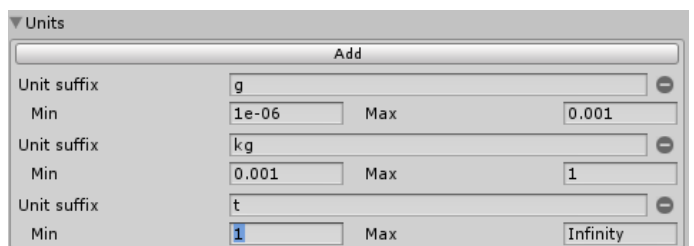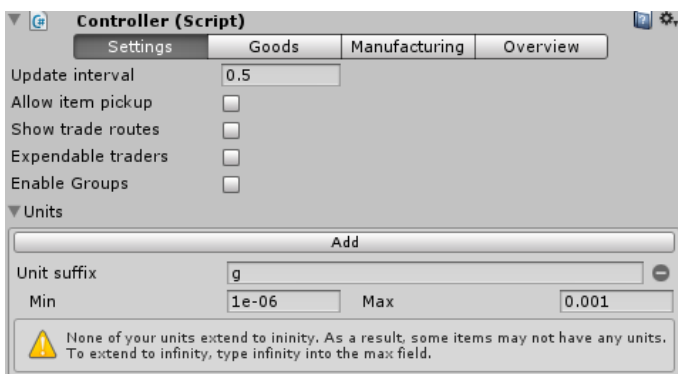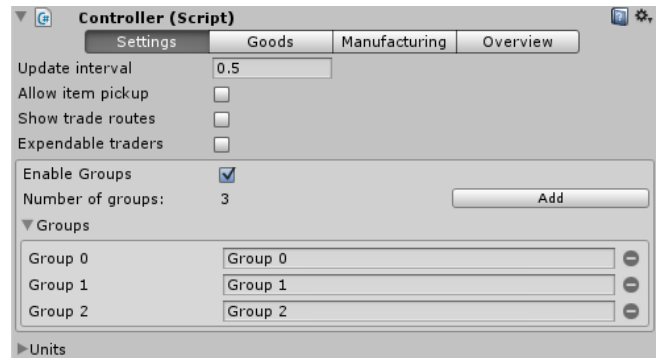
tradable item. Here, you can set its name, mass, and trade prices. If the units have been set up, next to the mass, the unit will display in brackets. The minimum mass is 0.000001, which if 1 = 1 ton, then this value is 1 gram! If the allow item pickup option in settings has been enabled, then there will also be an option for the item crate to be set. This is so that spawners can create the item, and unique crates or items are created for each type of good. To easily find the crate, press the 'Find crate' button. This will search through the assets for a GameObject with the same name as the item, and set the crate. There are three pricing options to be set. The base price is used to work out the price at each trade post, and the min and max will limit the prices so that it cannot increase too much or be too low. Each item can be removed using the 'Remove' button, and new items have the option to be added before or after the current one, so it is possible to keep your items in alphabetical order.
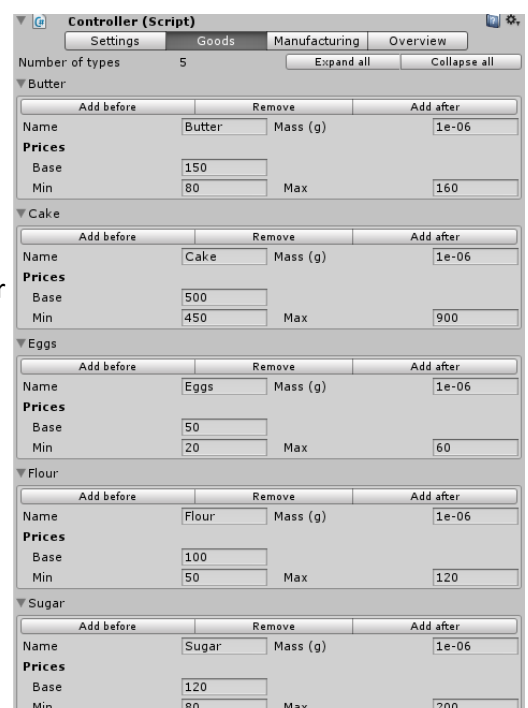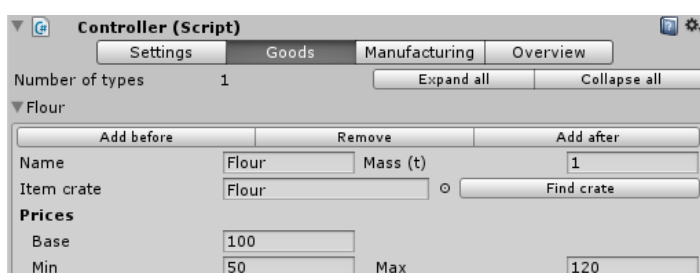
8. Manufacturing processes are also set in the controller. Under the manufacturing tab, add a new process by pressing 'Add'. Here, you can set the name, remove it, or set the items in the needing and making sets. In both of these groups, you can only add items that have already been defined under goods. Under the needing and making, add items, and set the quantities that are required or are made by the process. Note the check button, more on that later!

Trade Post setup:

1. Under the settings on a Trade Post, there is an option to set groups. If this has not been enabled in the controller, then there will an information message. If grouping has been enabled, it is impossible to disable the first group. To add a new group, simply enable the last group, and a new one will be created automatically. The grouping works by if a trade post is in one of the same groups, then a trade is possible. Multiple groups are allowed because a Trade Post can belong to many groups at once, and trade with all of the posts in the different groups.

2. Also in settings is an option to enable or disable different items at the Trade Post, preventing the purchase or sale of the item at that post. To make viewing the items easier, there is an option to view the items vertically or horizontally ascending.
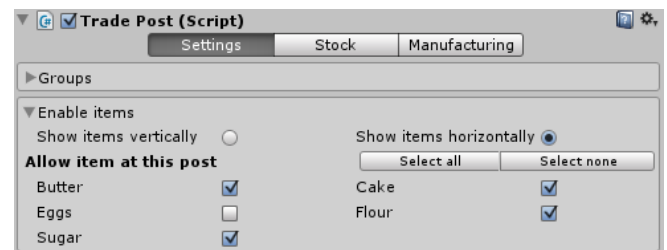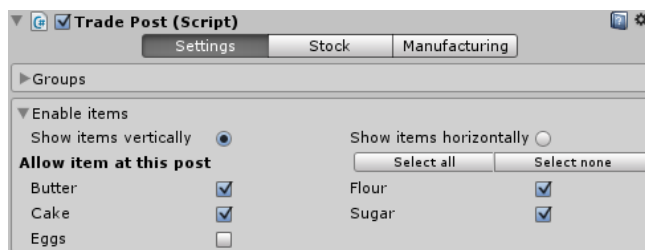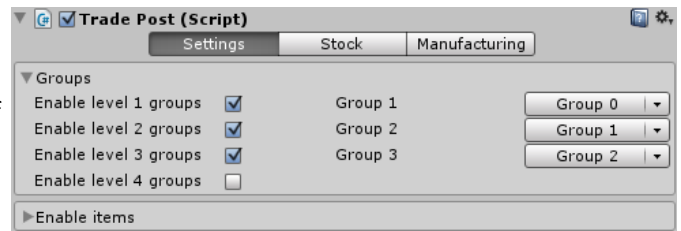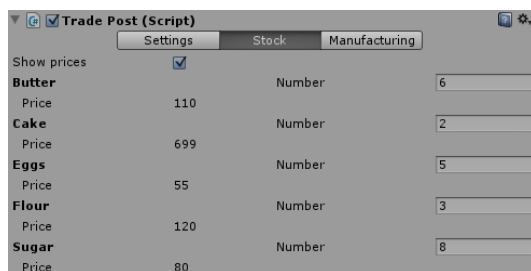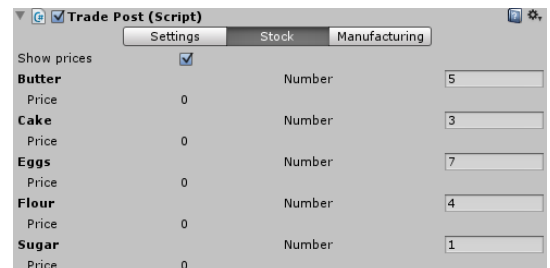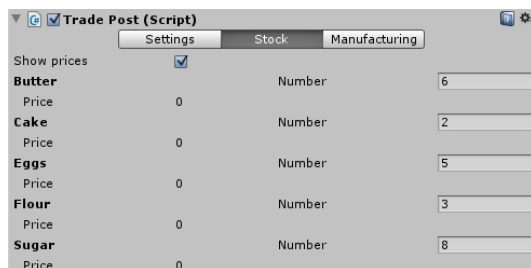
3. In stock, you can set how many of each item the Trade Post starts with. There is an option to view the prices, which are not editable and will show as 0 because the controller sorts all that out for you! The option is there so that it is possible to see the changes in prices when testing.
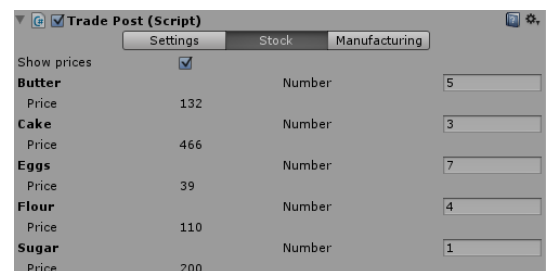
4. Under the manufacturing tab, if you want the post to be able to use the manufacturing process, then check the box, and set the manufacture time. It can only manufacture items when it has enough of each item that is in the needing part of the process.
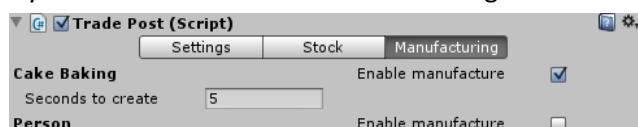
5.   Remember the check button in the controller under manufacturing? Well, once all of the manufacturing processes have been set at the trade posts, press this. It will work out if the number of each item may be expected to increase, decrease or stay the same. After checking all of the items, there is another which is for global change, so whether the total number of items in the game will change. This can only be used as a guide because the numbers are based on the posts always being able to follow a manufacturing process when there has been enough time between manufactures. It does not take into account spawners, so can only be an estimate.

Checking manufacturing

NOTE: This can only act as a guide because there may be pauses in manufacturing if there are not enough items, so there will be some variances.

It is useful to give an idea of whether the number of each item is expected to increase, decrease or stay the same as time tends to infinity. A larger number means that it will increase or decrease faster.

Butter decrease (0.75)
Cake increase (3.18)
Eggs increase (1.15)
Flour decrease (0.75)
Sugar decrease (1.70)
Global change: increase (1.13)

Ok

6.   Once all of the trade posts have been fully sorted, if you look at the overview tab in the controller, it will show some information about the items. The groups section shows the number of trade posts that belong in each group, and also says the number of unique links that exist, so the number of different links between trade posts. This shows unique links, so for example, if there were two trade posts, then the number would be one, although traders can go in either direction.

7.   The item totals show how many of each item exist in the game, when it is playing, you can see each of these change. The number in brackets is the number of trade posts each item is available at.

8.   Manufacturing totals simply shows the number of trade posts that are able to follow that manufacturing process.

9.   On the settings tab of the controller was an option to view the trade routes. This shows all of the possible routes that traders can follow, so is useful to see how the groups are working.
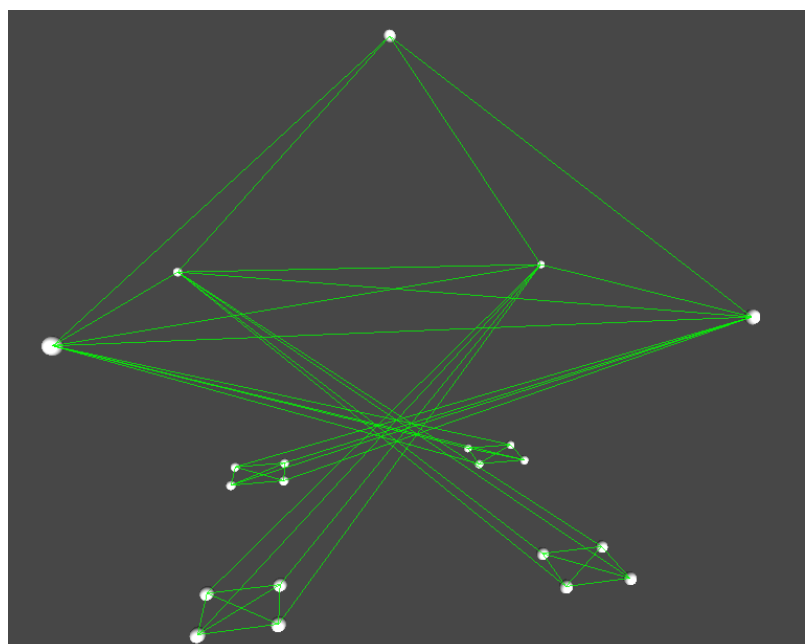
Spawner setup:

1. Spawners can be added to the game by creating an empty gameObjetct, setting the tag to 'Spawner' and adding the spawner script.

2. Once the spawners have been added, select the controller, and then you can go back to the spawners. This is so that the controller can tell the spawners what goods there are.

3. This has the option of setting the min and max times for spawns, which a random time will be generated between these two. When there are spawners, it is not possible to turn off item pickup in the controller.

4. The maximum number of items can be set to limit the number of items, so that the effects on performance can be limited. The sphere radius is the size of sphere in which an item will be generated, and it is possible to see the sphere in the scene view.

5. The options for allow spawn means that it is possible to limit the item that gets spawned randomly.
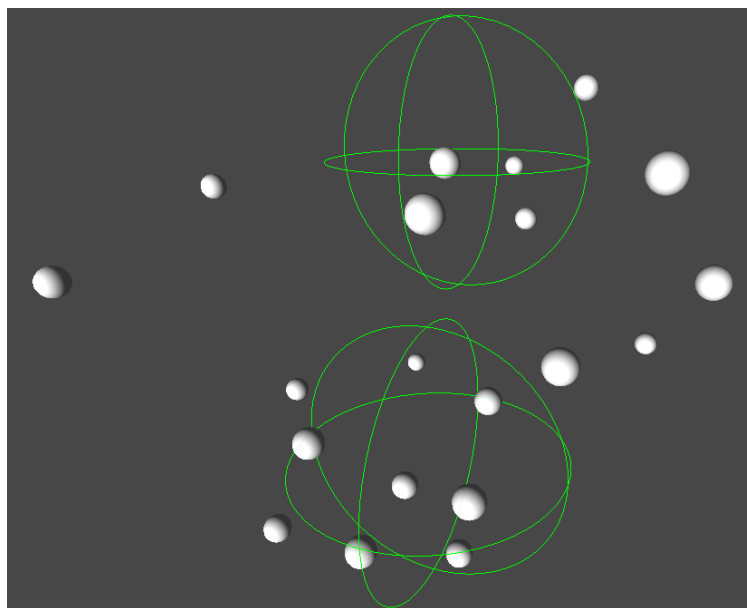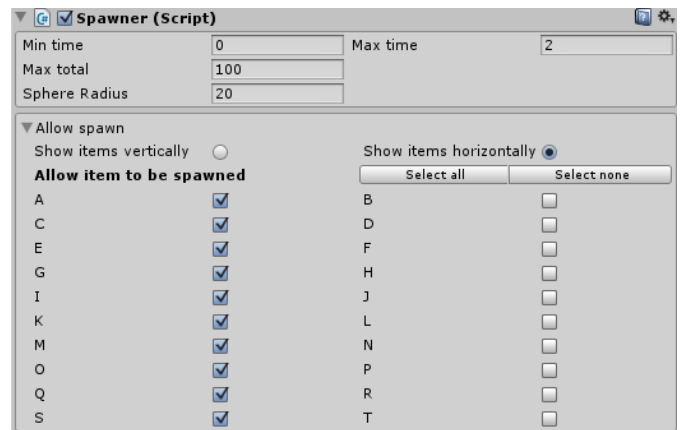
Information:

- 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.

- You may wish to replace the destroy and instantiate methods with other methods which pool the gameobjects. Instantiate methods can be found in controller, and spawner. Destroy methods are in controller, and trader.

Update Instructions: Some methods have been edited, so may produce errors, but this is because of a name change to be similar to other named methods or that they now require extra inputs. Check in the useful methods section for more information. New methods have also been added, so check below. New tags will need to be added, these are 'Spawner' and 'Item'.

Tips:

- 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.
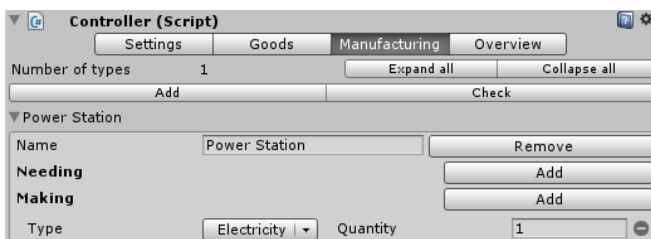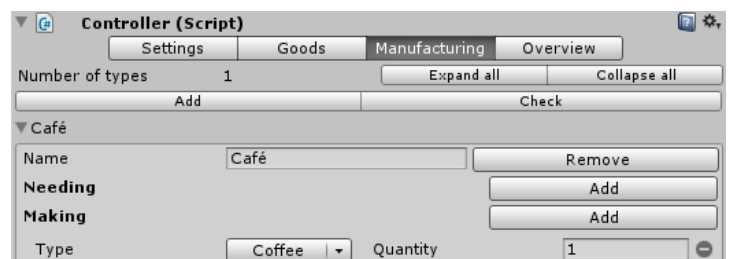
- Equally, a solar power station may only produce electricity and not require inputs, so only electricity on the making would be needed.

- 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.

Useful Methods:

- To enable a new Trade Post, use:

> **nameOfTradePost**.GetComponent<TradePost>().NewPost(**int[] itemNumbers, int[] manufactureTimes**);

You will need to change the variables written in bold to chosen ones. ItemNumbers is an array which is the number of each item found at the post. If the number is greater than or equal to 0, the item will be enabled, and will have that number of items in stock. ManufactureTimes is an array of times which is used to set whether a manufacturing process occurs or not. If this number is greater than 0, it will be enabled and the time will be set to the number passed in. The ItemNumbers array needs to be the same length as the number of items available, and the ManufactureTimes needs to have the same length as the number of manufacturing processes possible.

- To enable or disable a post that has previously been enabled:

> **nameOfTradePost**.GetComponent<TradePost>(). PostEnableDisable(**bool enable)**;

You will need to change the variables written in bold to chosen ones**.** Setting enabling to true means that the prices will be updated, but your code will need to enable or disable the post, otherwise trading to the post will continue as usual.

- To enable or disable a certain manufacturing process during gameplay:

> **nameOfTradePost**.GetComponent<TradePost>().ManufactureEnableDisable(**bool enable, int manufactureID, int timeToCreate**);

You will need to change the variables written in bold to chosen ones**.** Enable is true if the manufacturing process is going to be enabled, manufactureID is the manufacturing process number in the array, so the first process has and ID of 0. The time to create is how long the manufacture time.

- To enable or disable a certain item:

> **nameOfTradePost**.GetComponent<TradePost>().ItemEnableDisable(**bool enable, int itemID**);

You will need to change the variables written in bold to chosen ones**.** Enable is true if the item is going to be enabled. The ItemID is the item number as found in the controller, so the first will have an ID of 0.

- If you are not using expendable traders, to enable a new trader, use:

> **nameOfTrader**.GetComponent<Trader>().NewTrader (**GameObject post, float space, float stopTime**);

You will need to change the variables written in bold to chosen ones**.** Post is the Trade Post that the trader is starting at, space is the amount of cargo space available. The stopTime is the amount of time that a trader has to wait for before being allowed to leave a trade post.

- To make a trader drop items from the cargo hold:

> **nameOfTrader**.GetComponent<Trader>().DropItems(**int itemID**);

You will need to change the variables written in bold to chosen ones**.** The itemID is the item number as found in the controller, with the first being 0. This method could be used when a trader dies, and a loop used to drop all the items.

VERSION	CHANGES
V1.0	- Initial release
V1.1	- Traders have cargo size and items require mass specified so traders cannot take an infinite number of 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 the required station and destroyed on arrival. - Can disable items at posts so will not be traded or manufactured. - UI tweaks: • Can hide or show prices at trade posts.
V1.2	- Traders have custom inspectors. This also includes buttons for post finding based on location, or location setting based on the target post. - Goods now have units which are based on the mass of each item. - Can check the numbers of items expected as time increases through manufacturing. - Added a fully commented player script and created a new scene with this in. This includes a fully functioning shop mode, and click on a trader to view info about the trader. - Added a post enable disable method. - Added a method for easy allowing trade of an item at trade posts. - Fixed the pricing of items as before any items being carried would be ignored because it would recount each time. Now this does not happen, so should improve performance! - UI tweaks: • Can now view the items to enable / disable at trade posts increasing horizontally or vertically. • Select all / select none on enable / disable at trade posts. • Expand all / collapse all on goods and manufacturing in the controller.
V1.3	- Spawners have been added, and the code to allow traders to collect dropped items and continue to the correct trade post, and added a DropItem method for traders - Groupings. It is now possible to group trade posts together, so that not every trade post trades with every other one. - Added an option to change the update frequency of the post prices, and possible trades to try and improve performance. - Added in an overview to controller, showing information about groups, goods and manufacturing. - Can view trade links, spawn sphere radius and radar sphere in the scene view. - Removed speed multiplier from traders - this was part of the demo code, but changed the scales of everything in the scene view so it is no longer needed. - Fixed bug where items could not be added to manufacturing processes without trade posts being present. - Fixed issue where it was not possible to set the maximum number of expendable traders to 0, so could not be infinite - Some other performance enhancements. - Massive UI changes: • Controller and trade posts have a toolbar to make editing easier. • Boxes have been added to many editor windows to make groups more obvious and clearer. • Changed style of remove buttons, drop down lists and horizontal / vertical items to improve experience