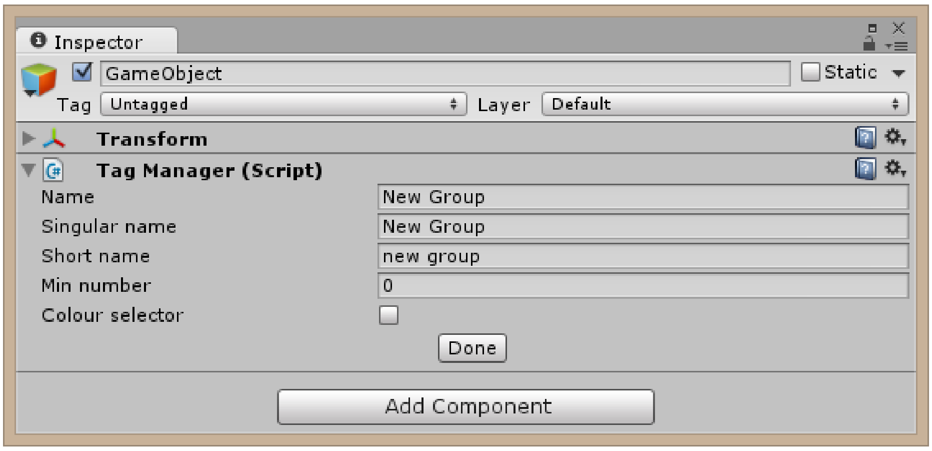
TagManagement

A simple way of adding multiple tags to any GameObject. It also allows tags to be in different groups, useful for only showing the tags that are needed to be shown.

How to use

Add a TagManager script to a GameObject. This will give you a list of options used to set up the tag group.



☑ GameObject □ Static

Tag Untagged Layer Default

Name New Group

Singular name New Group

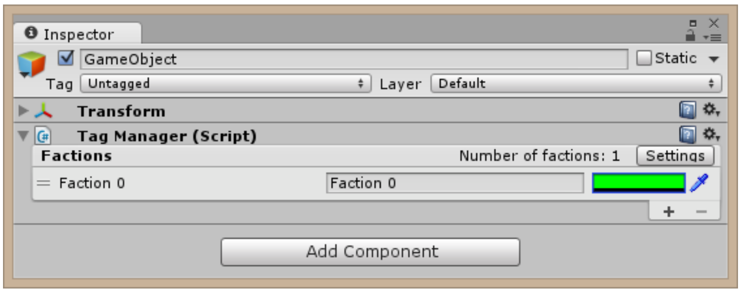
Short name new group

Min number 0

Colour selector □

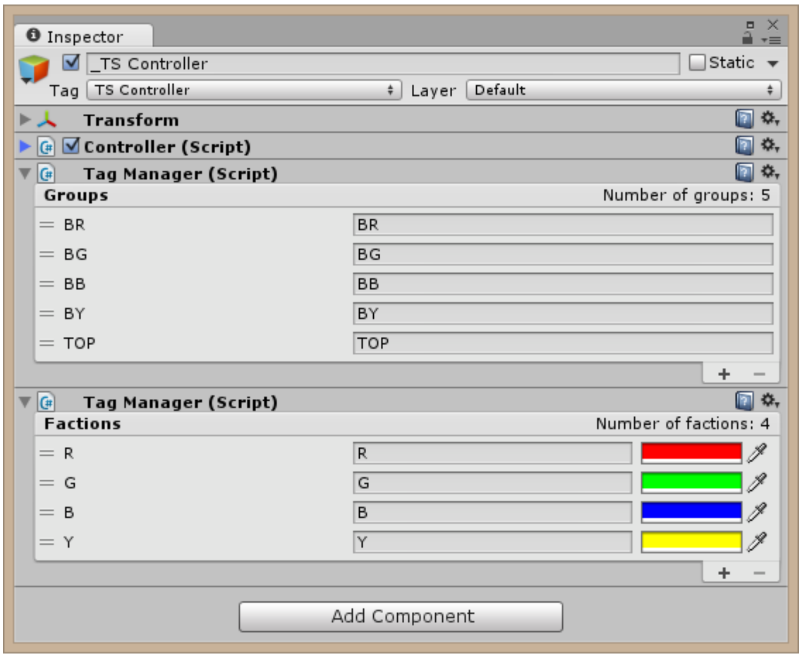
- Name: this is the name used to reference the specific tag group later. This is case sensitive
- Singular name: this is used when a new tag is added and is for the initial name
- Short name: this is used when the number of tags in the group is shown
- Min number: the minimum number of tags that are allowed in the group
- Colour selector: whether each tag will have an option to select a colour

Press done once this is complete and the information shown will be similar to this:



☑ GameObject □ Static
Tag Untagged Layer Default
Tag Manager (Script)
Factions Number of factions: 1
Faction 0 | Faction 0 | 🟩

In the debug inspector, a locked option can be seen. If this is selected, the settings button disappears to prevent any changed being made. This may be useful for creating packages or objects that need specific names. For example, TradeSys has the groups and factions tag managers locked because these are important to how TradeSys works.



☑ _TS Controller □ Static
Tag TS Controller Layer Default
☑ Controller (Script)
Tag Manager (Script)
Groups Number of groups: 5

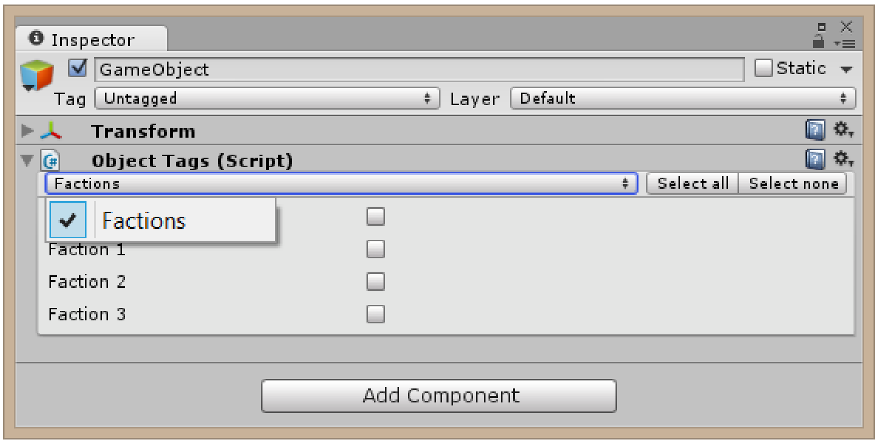
BR	BR
BG	BG
BB	BB
BY	BY
TOP	TOP

Tag Manager (Script)
Factions Number of factions: 4

R	R	🟥
G	G	🟩
B	B	🟦
Y	Y	🟨

If any options are not correct, press settings and you can apply the modifications. Press the + button to add more tags, select a tag and press – to remove one.

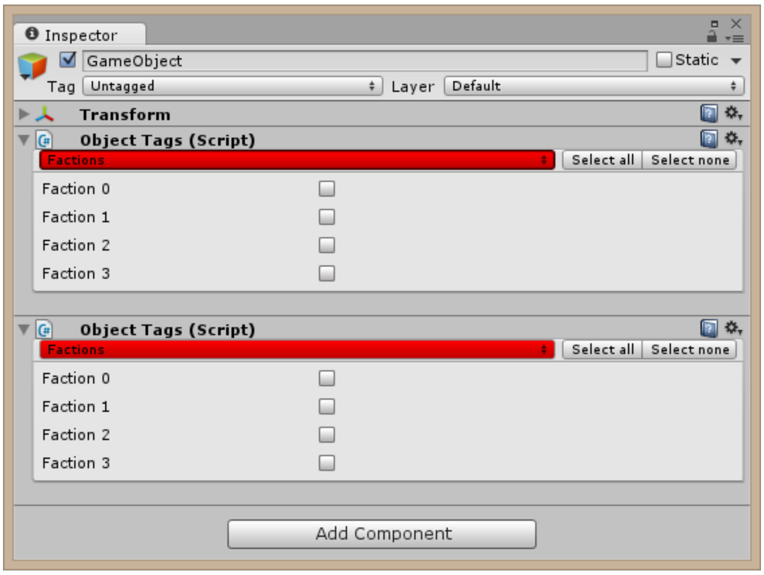
To select the tags that a GameObject has, add an ObjectTags script to the object. The script can be linked to any TagManagement component in the scene by pressing the drop down menu and selecting the desired option.



☑ GameObject □ Static
Tag Untagged Layer Default
Object Tags (Script)
Factions

Faction 0	□
Faction 1	□
Faction 2	□
Faction 3	□

When the option is changed, the list will be updated with the tags belonging to the selected group. If multiple groups are desired, add more ObjectTags components to the GameObeject and select a different group. Errors may occur if multiple ObjectTags point to the same TagManager, and the name will be highlighted in red to reflect an issue.


Inspector

☑ GameObject □ Static

Tag Untagged Layer Default

Transform

Object Tags (Script)

Factions

Faction 0 □
Faction 1 □
Faction 2 □
Faction 3 □

Object Tags (Script)

Factions

Faction 0 □
Faction 1 □
Faction 2 □
Faction 3 □

If the tags are reordered in a TagManager, the changes will be updated on any ObjectTags component, keeping whether that option had been previously selected or not.

Methods

TagManager

Init: useful for other scripts that require a TagManager component to be added as it sets up the given information

```
void Init (string groupNameIn, string singularIn, string shortNameIn,
bool lockedIn, bool coloursIn, int minNumberIn)
```

```
void Init (string groupNameIn, string singularIn, string shortNameIn,
bool lockedIn, bool coloursIn, int minNumberIn, SC[] namesIn)
```

```
void Init (string groupNameIn, string singularIn, string shortNameIn,
bool lockedIn, bool coloursIn, int minNumberIn, List<SC> namesIn)
```

- groupNameIn: the name of the group
- singularIn: the singular name for the group
- shortNameIn: the short name for the group
- lockedIn: whether the tag manager settings are locked so the settings button is not shown
- coloursIn: whether there is an option to select colours
- minNumberIn: whether there are a minimum number that can be in the group at any time
- namesIn: this is either not used or is an array or list. `SC` contains `string tagName` and `Color colour`

GetAllScripts: get all of the ObjectTags scripts connected to the current manager

```
ObjectTags[] GetAllScripts ()
```

Static methods

GetManager: find and return the TagManager from the name specified

```
TagManager GetManager (GameObject curObj, string groupName)
```

- curObj: the GameObject that has the TagManager script
- groupName: the group name specified when setting up the TagManager

ShareEnabled: return whether two GameObjects have the script and both share an enabled option

```
bool ShareEnabled (GameObject obj1, GameObject obj2, string groupName)
```

```
bool ShareEnabled (GameObject obj1, GameObject obj2, TagManager manager)
```

```
bool ShareEnabled (ObjectTags group1, ObjectTags group2, string
groupName)
```

```
bool ShareEnabled (ObjectTags group1, ObjectTags group2, TagManager
manager)
```

- obj1 & obj2: the GameObjects to be compared
- groupName: the name of the group that is to be compared
- manager: the TagManager of the group to be compared
- group1 & group2: the ObjectTags component on each GameObject that are to be compared

ObjectTags

Init: useful for adding the ObjectTags script and applying the settings from other scripts

```
void Init (bool lockedIn, string managerGroupName)
```

```
void Init (bool lockedIn, TagManager connectedManager)
```

- `lockedIn:` whether the object tags settings are locked and cannot be changed to be connected to a different tag group
- managerGroupName: the name of the group that the script is to show the tags of
- connectedManager: the TagManager that the script is to show the tags of

Static methods

GetTagComponent: find the ObjectTags component from the name given

```
ObjectTags GetTagComponent (GameObject curObj, string connectedName)
```

- `curObj:` the current GameObject that the ObjectTags component is to be found on

- `connectedName`: the group name that the ObjectTags script is showing

GetTagsList: return a list showing what tags have been selected

```
List<bool> GetTagsList (GameObject curObj, string connectedName)

List<bool> GetTagsList (ObjectTags tagObj)
```

- `curObj`: the current object that the ObjectTags script is on
- `connectedName`: the name of the group that the ObjectTags script is showing
- `tagObj`: the ObjectTags script where the list is to be found

NOTE: the first calls GetTagComponent, so if it is being used repeatedly, it may be better to store it as a variable in your code.

GetTagNumber: get the index of a tag from the name of the tag

```
int GetTagNumber (ObjectTags tagObj, string tagName)
```

- `tagObj`: the ObjectTags which is showing the group with the tag to find
- `tagName`: the name of the tag to find in the group

GetTag: get whether a tag has been selected or not

```
bool GetTag (GameObject curObj, string connectedName, string tagName)

bool GetTag (GameObject curObj, string connectedName, int tagNumber)

bool GetTag (ObjectTags tagObj, string tagName)

bool GetTag (ObjectTags tagObj, int tagNumber)
```

- `curObj`: the current object that the ObjectTags script is on
- `connectedName`: the name of the group that the ObjectTags script is showing
- `tagName`: the name of the specific tag in the group to find if selected
- `tagNumber`: the index of the tag in the group
- `tagObj`: the ObjectTags script where the list is to be found

NOTE: The first two call GetTagComponent and the third calls GetTagNumber. If either are being used repeatedly, it may be better to store as variables in your code.

SetTag: set a specific tag to be enabled or not

```
void SetTag (GameObject curObj, string connectedName, string tagName,
bool selected)
```

```
void SetTag (GameObject curObj, string connectedName, int tagNumber,
bool selected)
```

```
void SetTag (ObjectTags tagObj, string tagName, bool selected)
```

```
void SetTag (ObjectTags tagObj, int tagNumber, bool selected)
```

- `curObj`: the current object that the ObjectTags script is on
- `connectedName`: the name of the group that the ObjectTags script is showing
- `tagName`: the name of the specific tag in the group to find if selected
- `selected`: whether the tag is selected or not
- `tagNumber`: the index of the tag in the group
- `tagObj`: the ObjectTags script where the list is to be found

If there are any methods that you feel that I am missing and you would like added, please contact me at: pryercj@outlook.com

Change Log

1.0.1

GetTagComponent will return a component that has not been connected to a manager if the required component cannot be found.