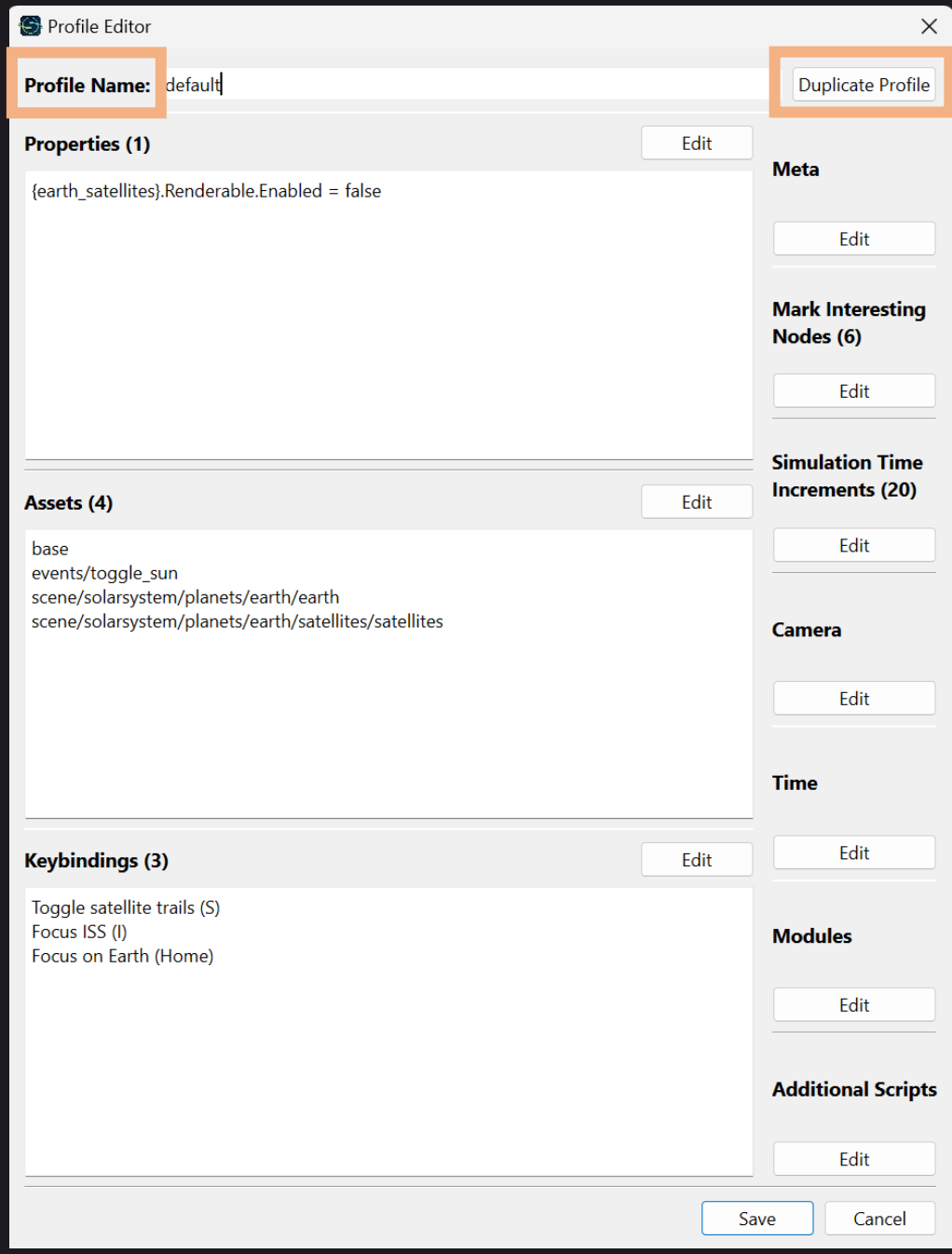
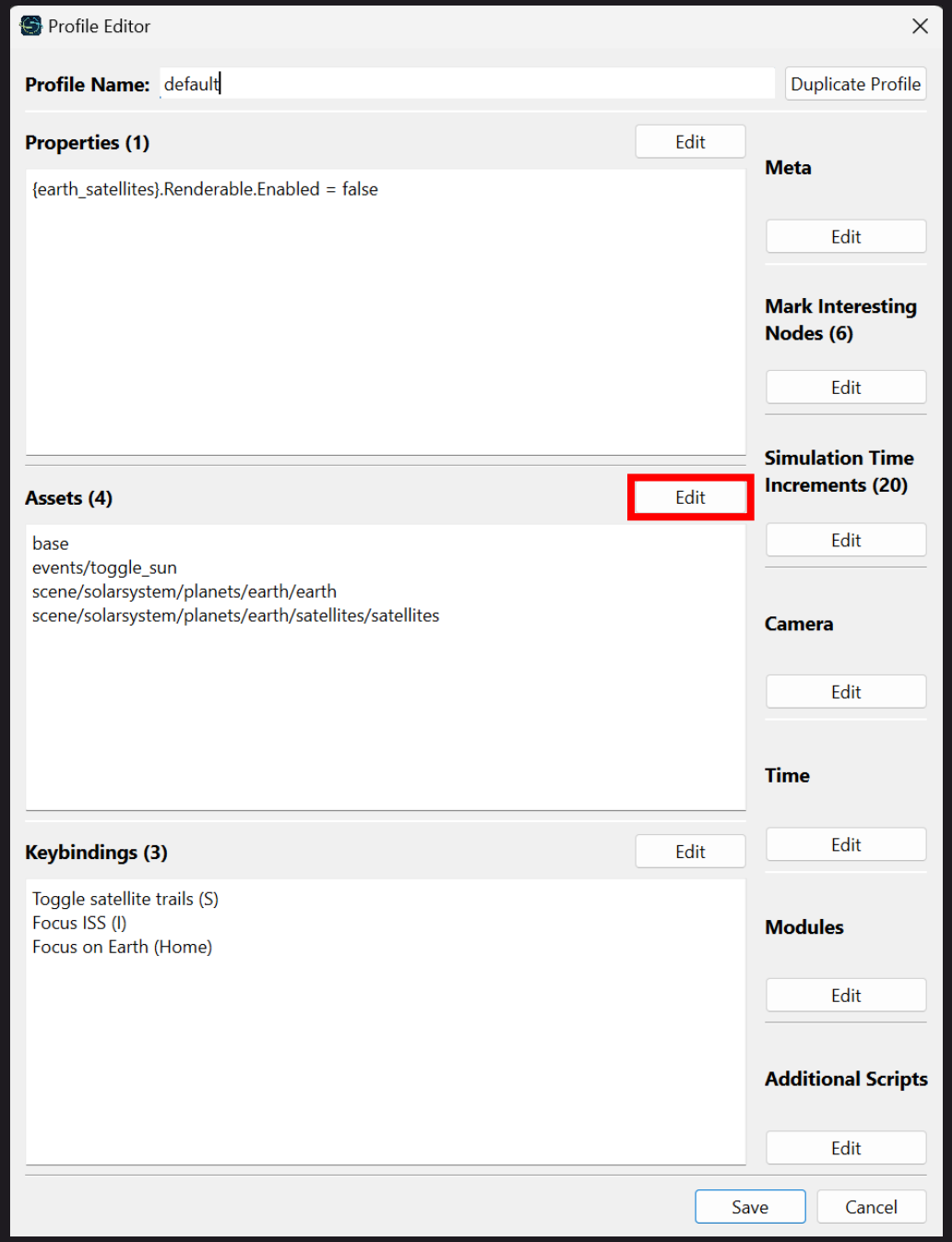
Profile Editor

- Profile Name: The name of the *profile*
- Duplicate Profile: Creates a copy of the selected *Profile*
- Properties
- Assets
- Keybindings
- Meta
- Mark Interesting Nodes
- Simulation Time Increments
- Camera
- Time
- Additional Scripts



Profile Editor

- All *assets* that the *profile* explicitly loads
  - Individual *assets* can *require* other assets in order to load. This requirement is defined within the *asset file* and will not be visible in the *Profile Editor*



Profile Editor/Assets

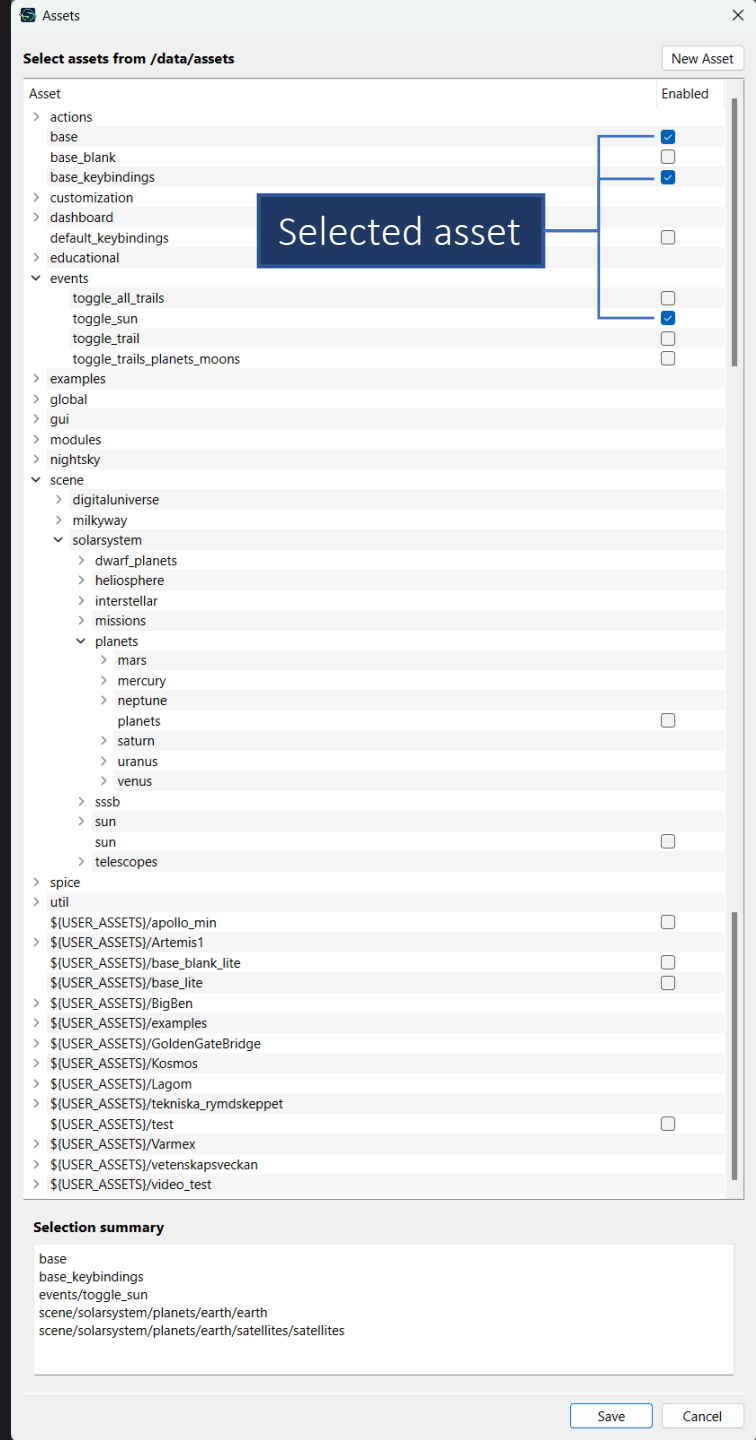
Assets located in folder data/assets/

- These are all the *assets* that ships with OpenSpace by default

Assets located in folder user/data/assets/

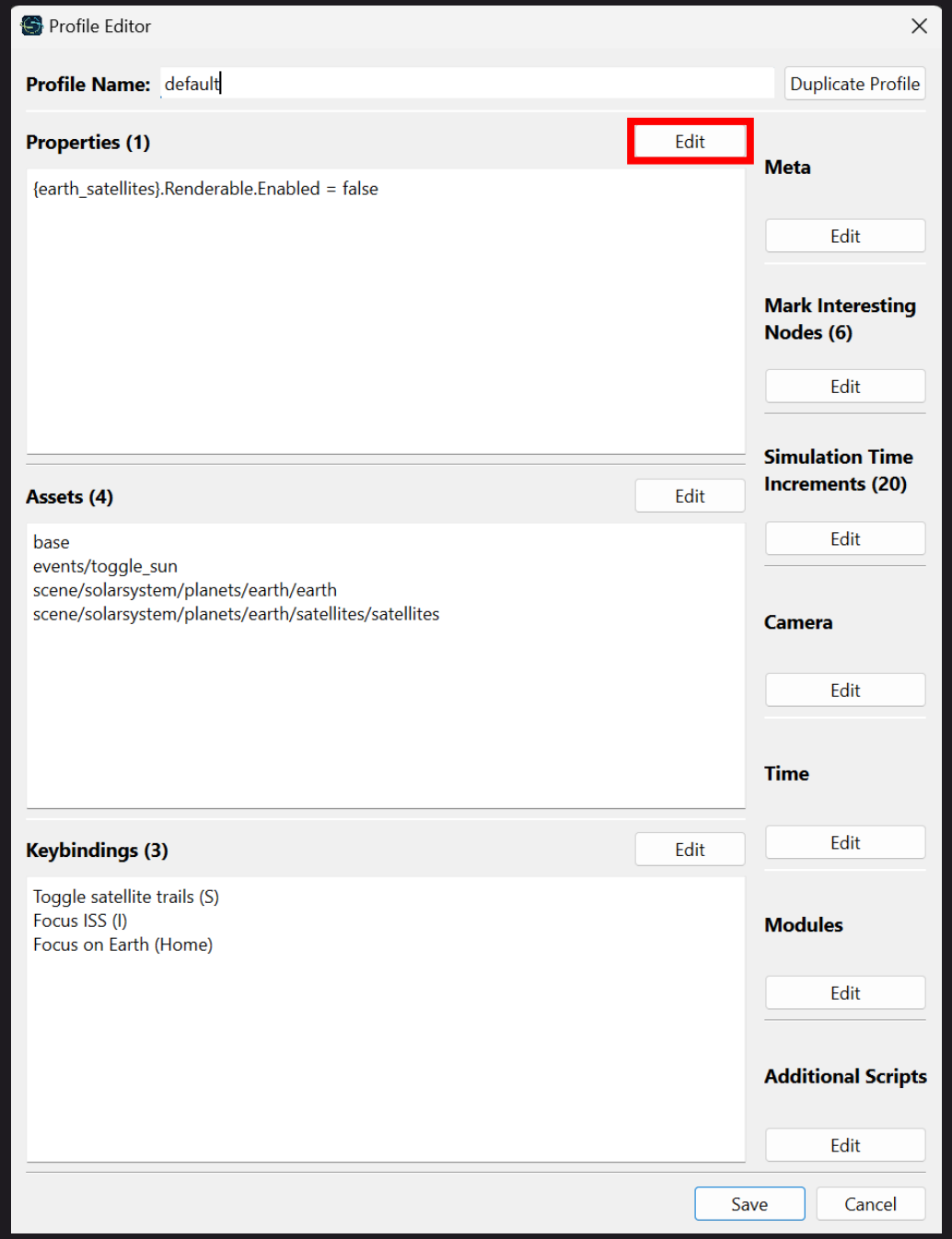
- User created *assets*
- The user folder is by default located in the openspace folder

Summary of selected assets



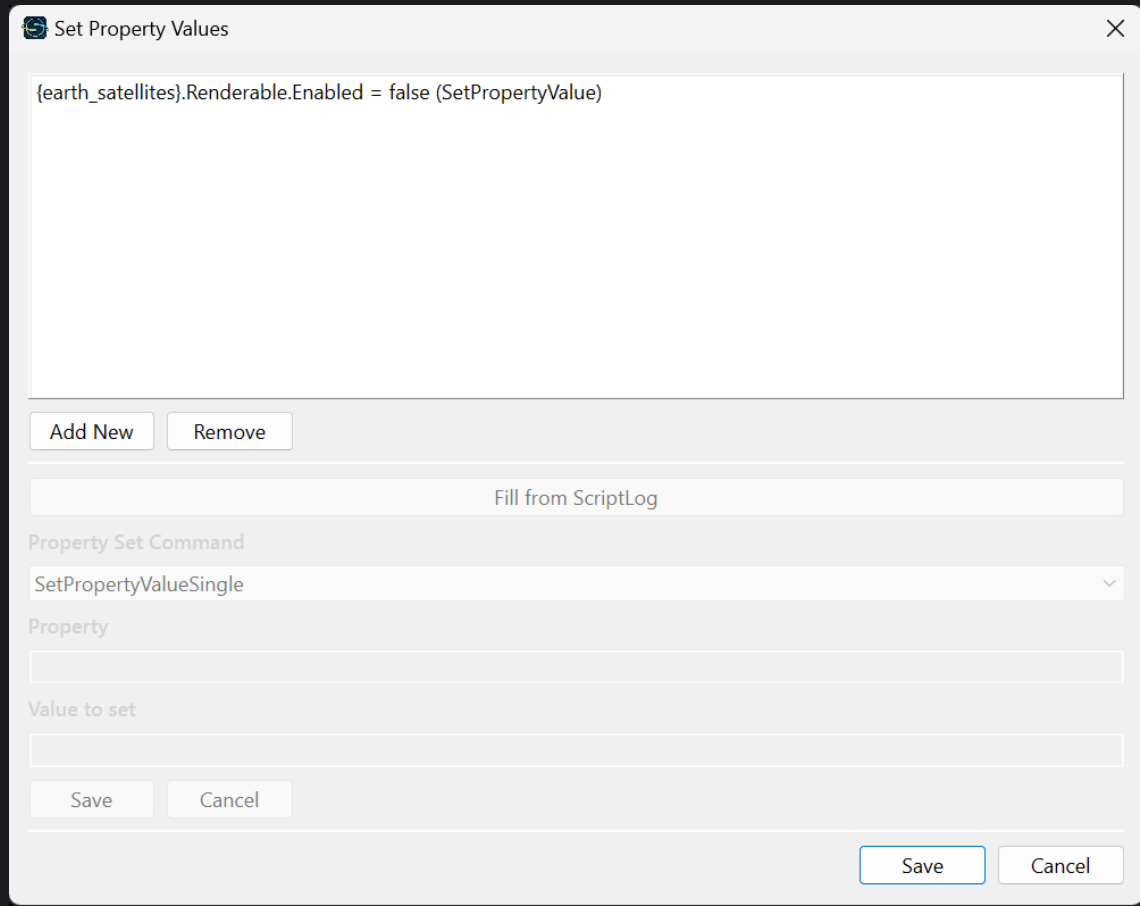
Profile Editor

- The default setting that should be applied to the *Properties* in the scene



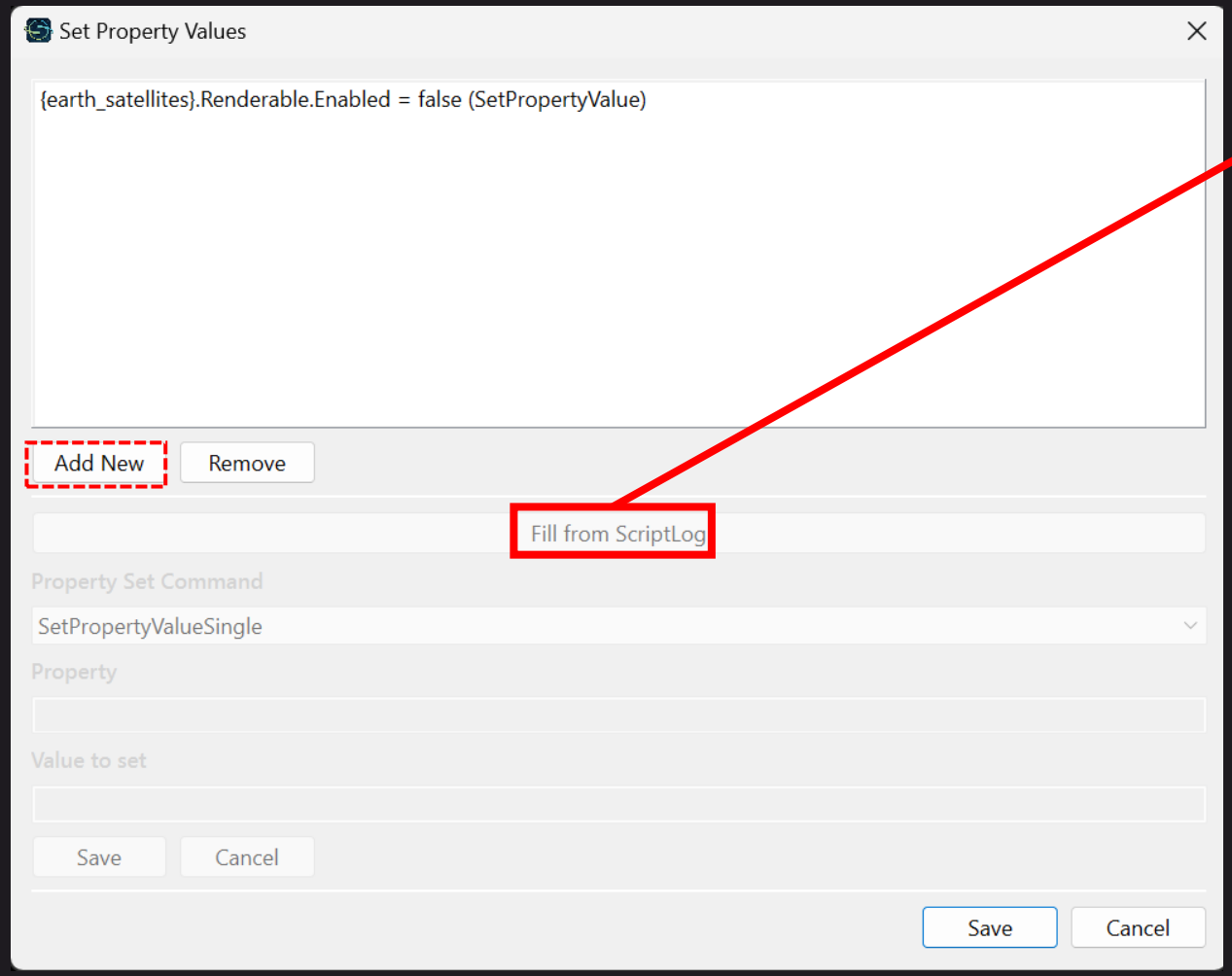
Profile Editor/Properties

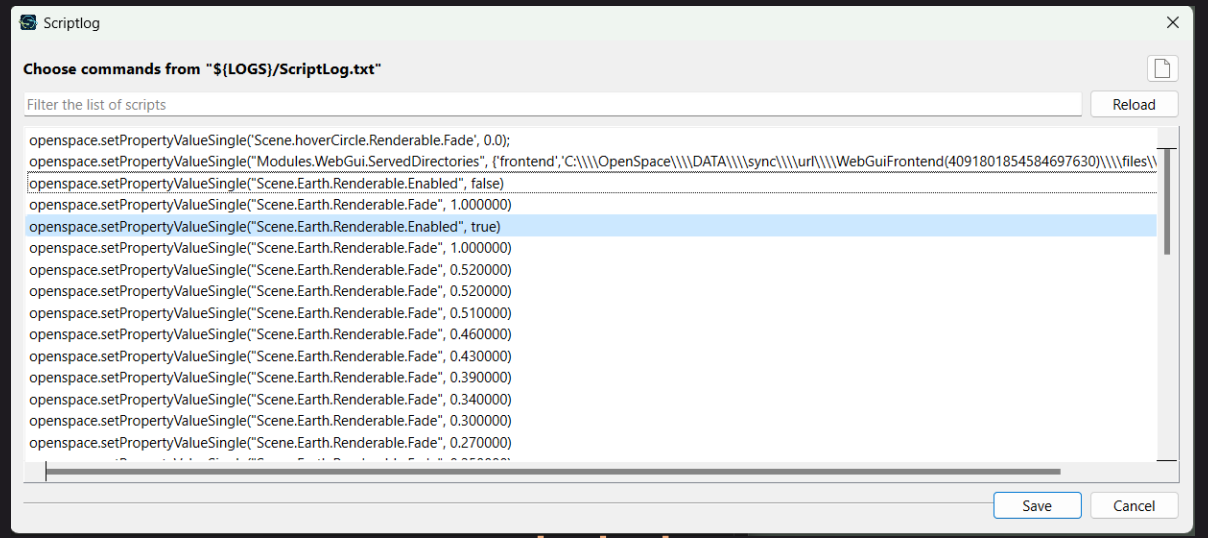
- The default setting that should be applied to the *Properties* in the scene
  - Example: Making the Earth transparent by default



Note: The *Profiles* that are provided by us are read-only be default. The “Duplicate Profile” button creates a copy of the currently active *Profile.* Alternatively, open an existing *Profile* and save with a new name

Profile Editor/Properties



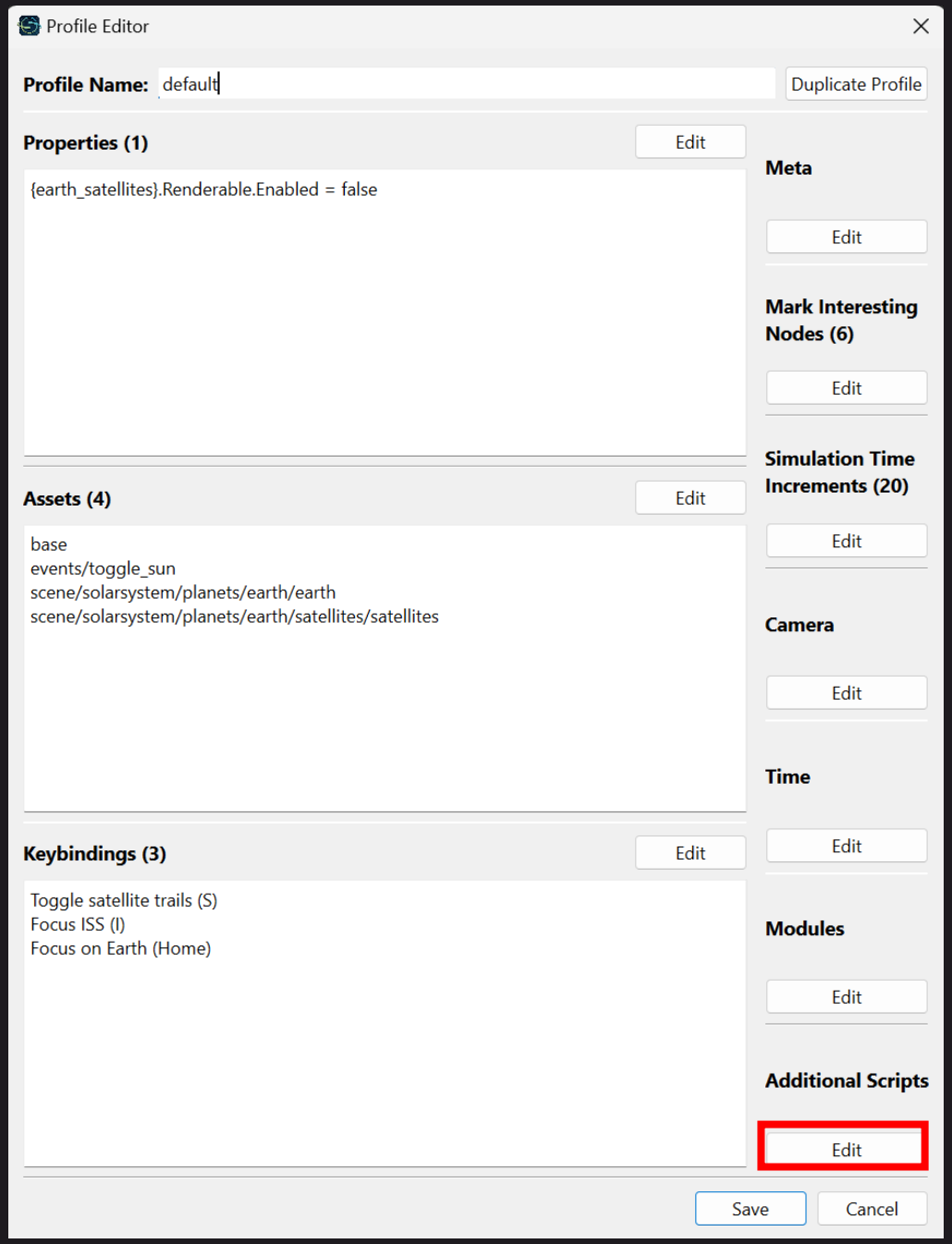



```
1  openspace.setPropertyValueSingle('Scene.hoverCircle.Renderable.Fade', 0.0);
2  openspace.globebrowsing.goToGeo([[Earth]], 58.5877, 16.1924, 17000000)
3  openspace.setPropertyValueSingle("Modules.WebGui.ServedDirectories", {'frontend','C:\\\\OpenSpace\\\\DATA\\\\sync\\\\url\\\\WebGu
4  return openspace.exoplanets.getListOfExoplanets()
5  return openspace.skybrowser.getListOfImages()
6  return openspace.skybrowser.getWwtImageCollectionUrl()
7  openspace.setPropertyValueSingle("Scene.Earth.Renderable.Enabled", false)
8  openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 1.000000)
9  openspace.setPropertyValueSingle("Scene.Earth.Renderable.Enabled", true)
10 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 1.000000)
11 openspace.time.interpolateNextDeltaTimeStep()
12 openspace.time.interpolatePreviousDeltaTimeStep()
13 openspace.time.pauseToggleViaKeyboard()
14 openspace.time.pauseToggleViaKeyboard()
15 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.520000)
16 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.520000)
17 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.510000)
18 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.460000)
19 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.430000)
20 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.390000)
21 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.340000)
22 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.300000)
23 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.270000)
24 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.250000)
25 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.240000)
26 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.230000)
27 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.230000)
28 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.240000)
29 openspace.setPropertyValueSingle("Scene.Earth.Renderable.Fade", 0.270000)
```

Script log from Logs/ScriptLog.txt

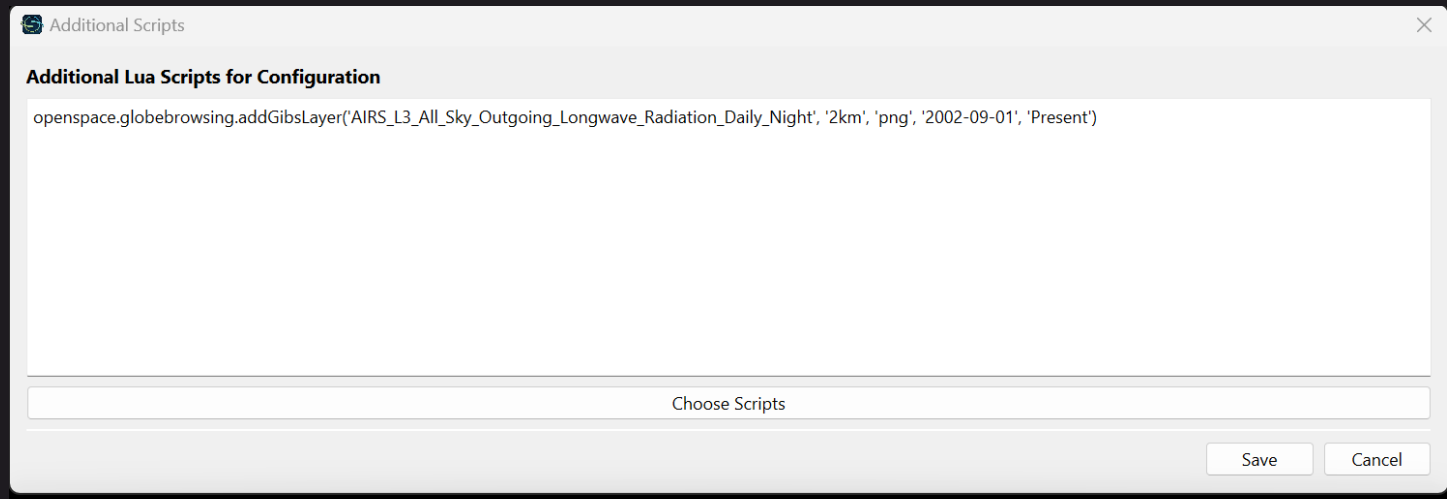
Profile Editor

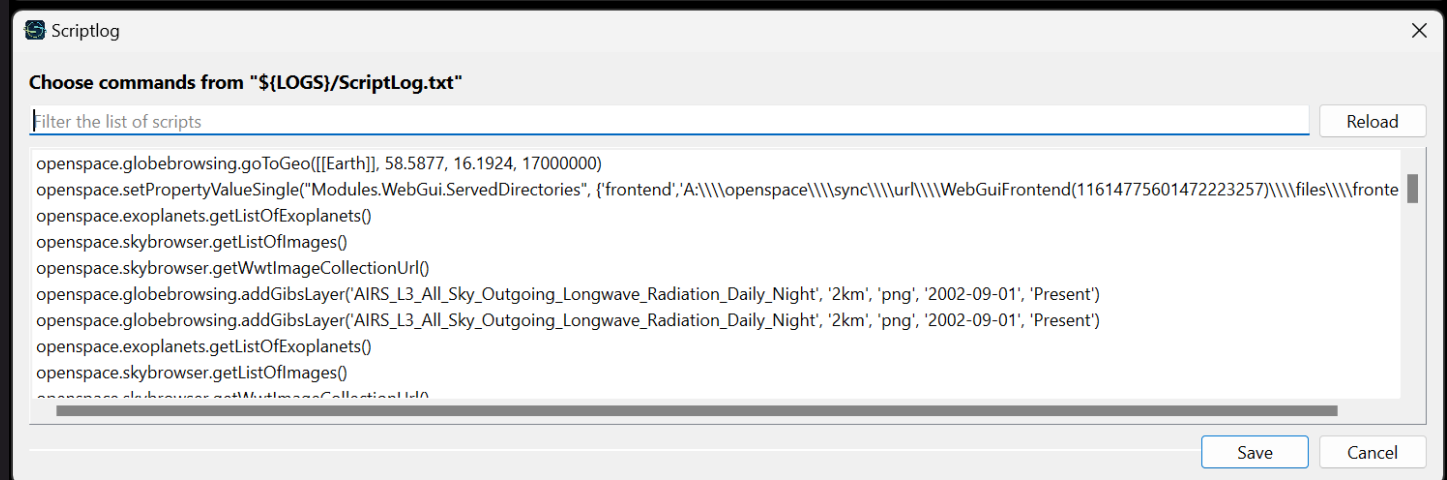
- Additional Lua scripts that should be executed when the *Profile* is loaded



Profile Editor/Additional scripts

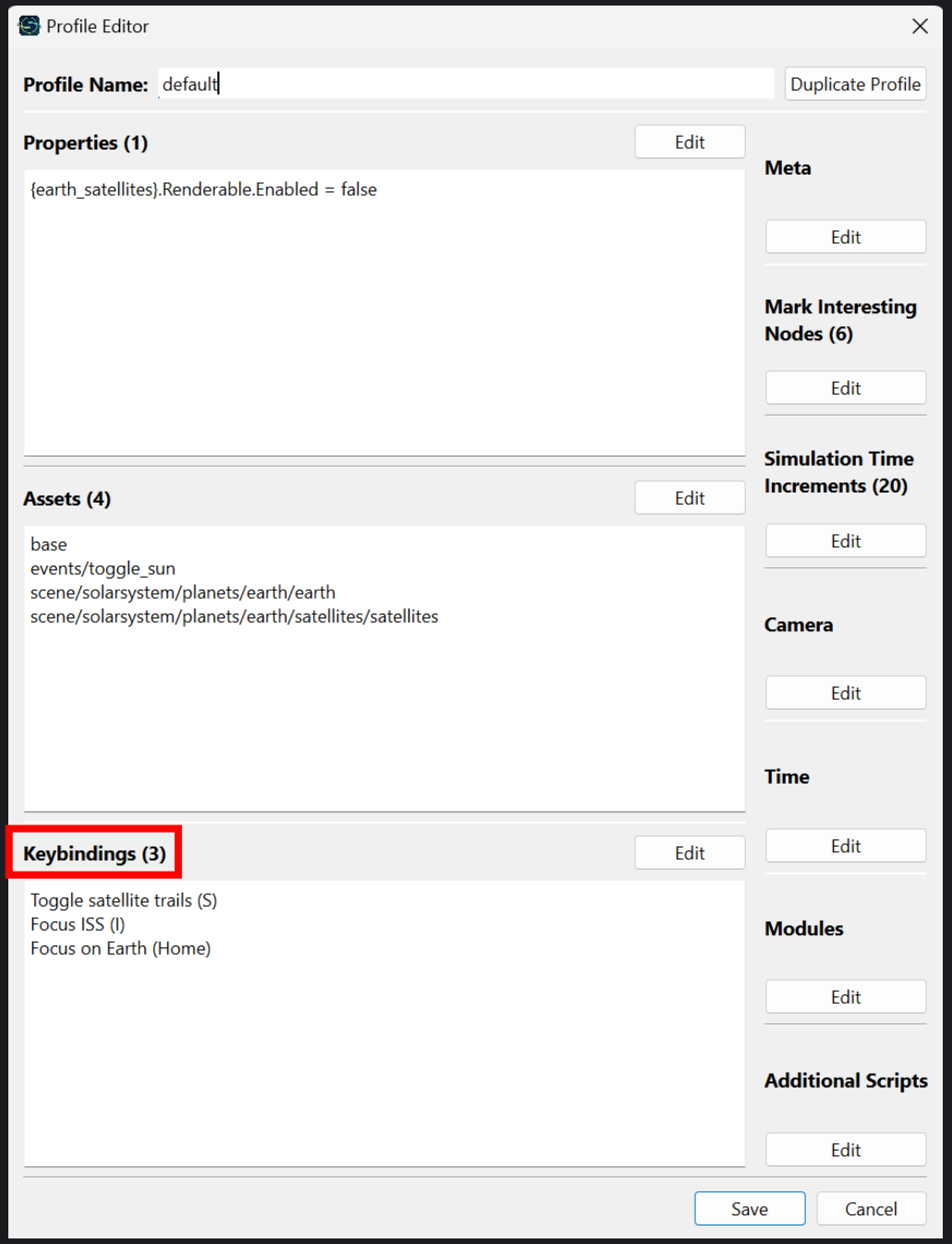
- Example: Run a script when a profile is loaded
- “Choose Scripts” shows the list of scripts that were executed the
  last time OpenSpace ran



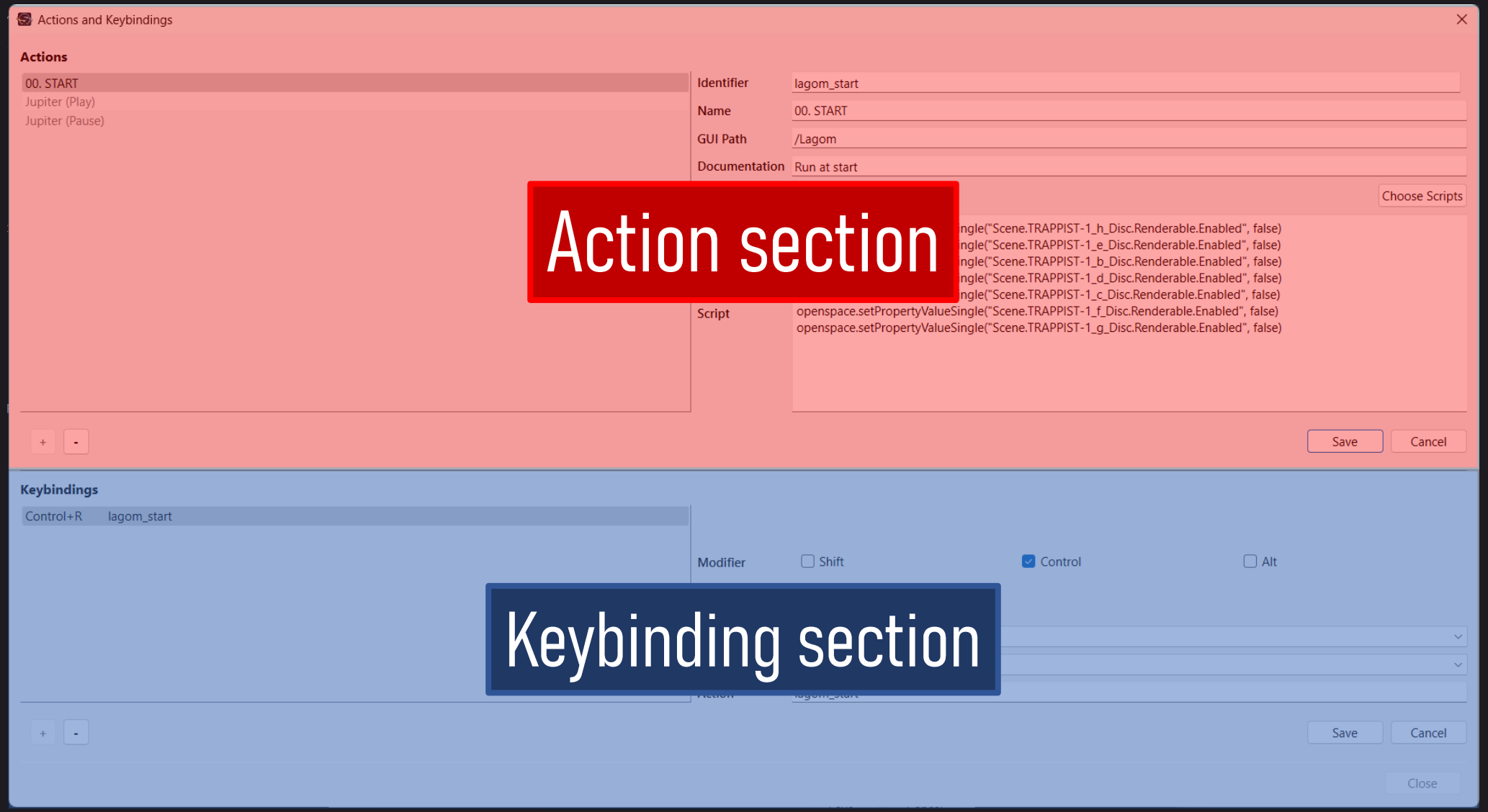


Profile Editor

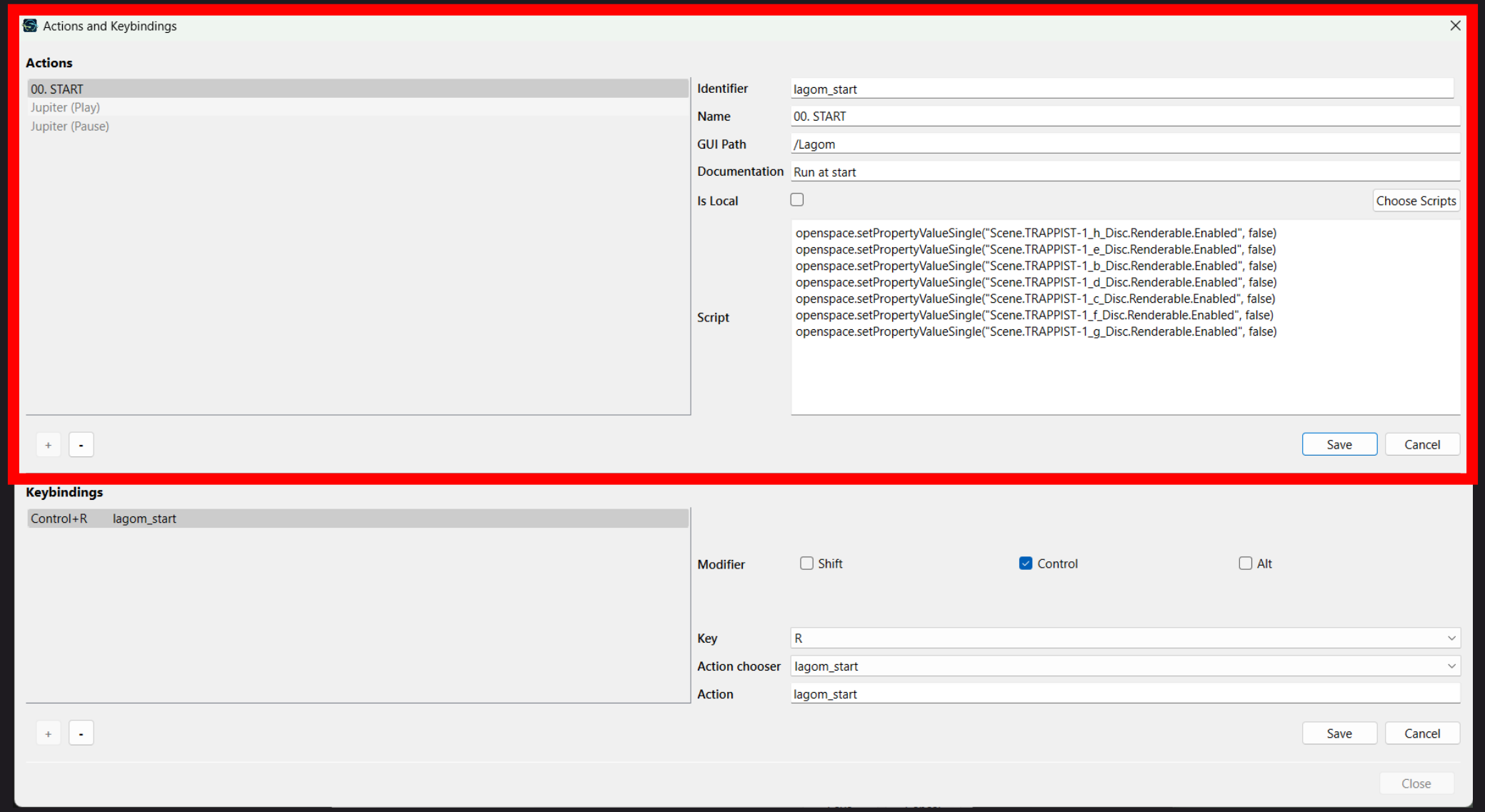
- The keyboard shortcuts that are defined in this *Profile*
  - Keybinding and actions can also be defined and loaded via an *asset*. These keybindings and actions will not be visible in the *profile editor*



Profile Editor/Actions & Keybindings

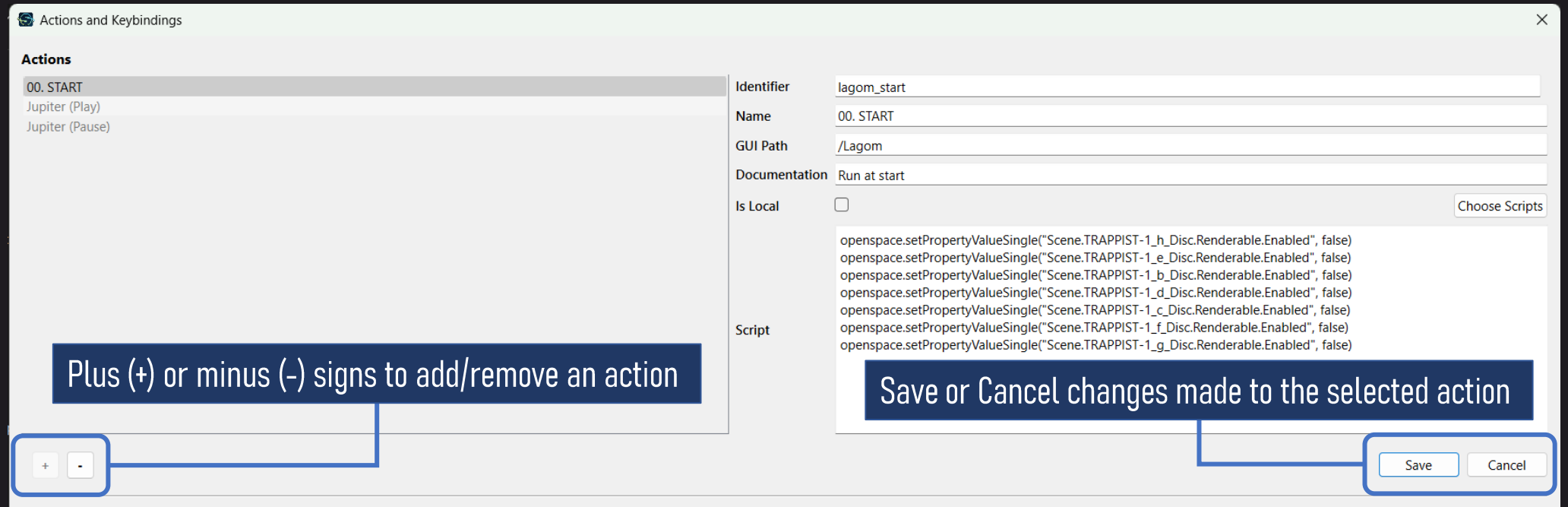


Profile Editor/Actions & Keybindings

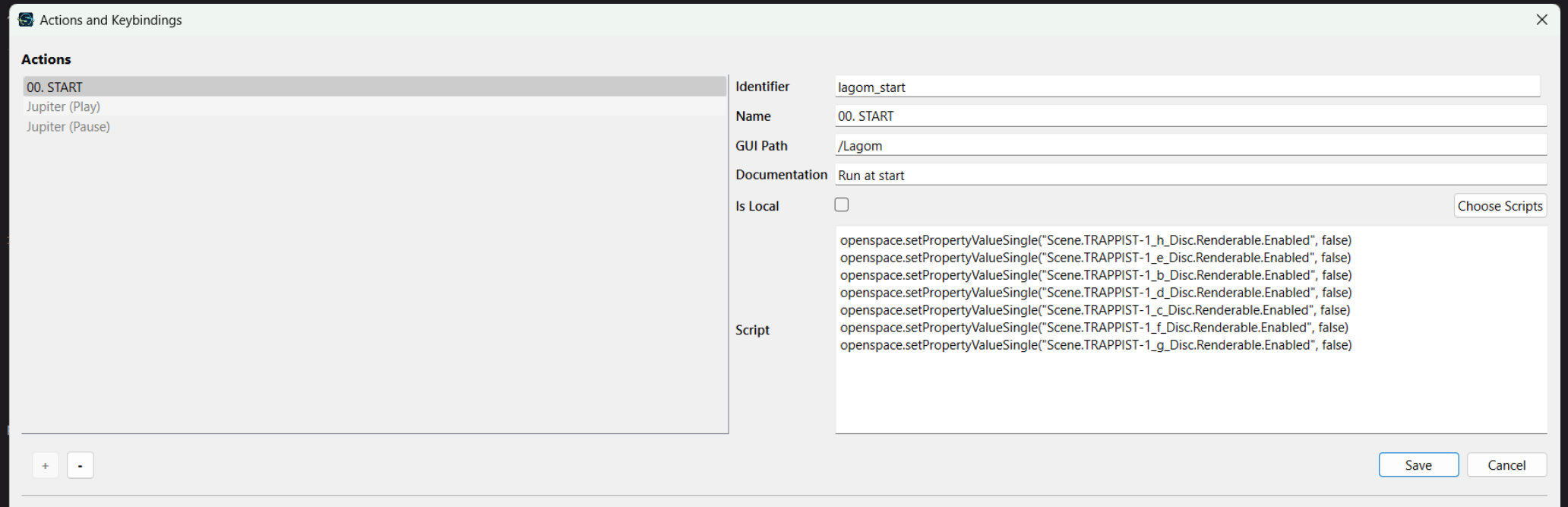


Profile Editor/Actions & Keybindings

- Left side contains a list of all *actions* created using the *profile editor* for the selected *profile*
- Right side shows configuration and contents of the selected *action*
  - Identifier: A unique name
  - Name: The name of the action that will be shown in the interface
  - GUI Path: Where in the folder structure the action will appear in the *action panel*
  - Documentation: Information that is shown when hovering over the action button in the interface
  - Is Local: Determines if the *action* will be run only on master or on nodes as well in a clustered environment
  - Script: Contains the Lua code that will be executed when running the *action*
    - You can either select scripts or write your own Lua code



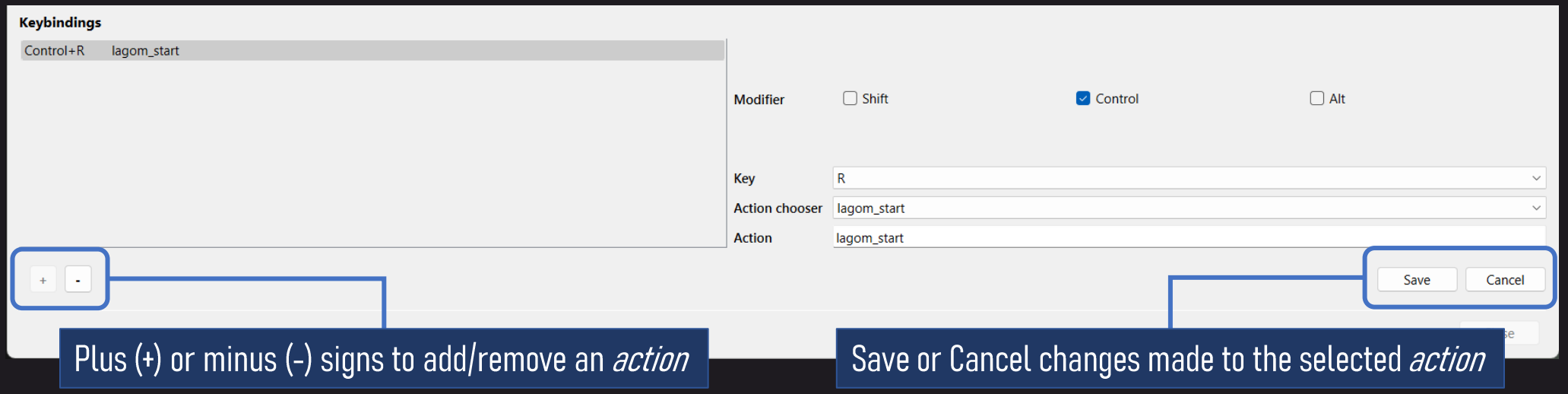
Profile Editor/Actions & Keybindings



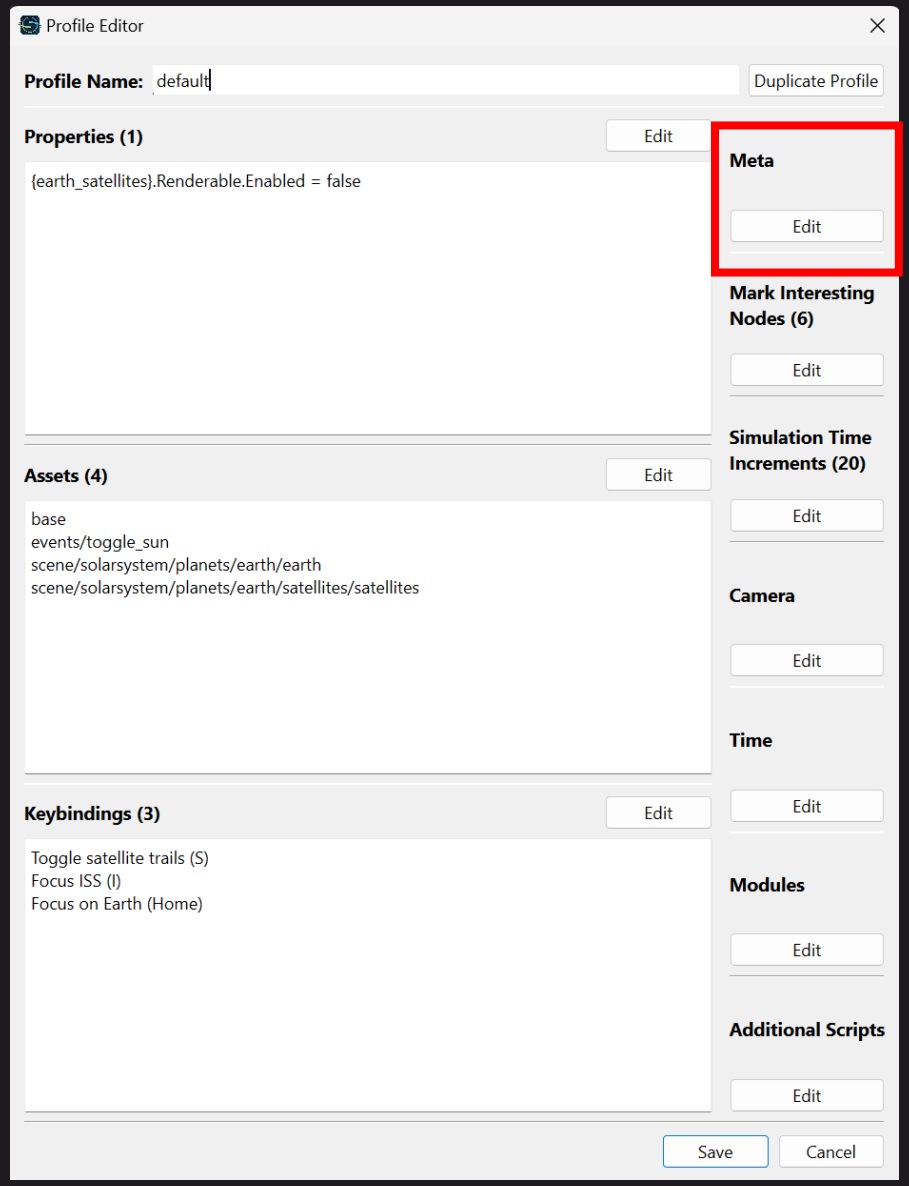
Profile Editor/Actions & Keybindings

- Left side contains a list of all keybindings created for the *profile*
  - First column is key combination and second column is the *identifier* of the *action* that will run

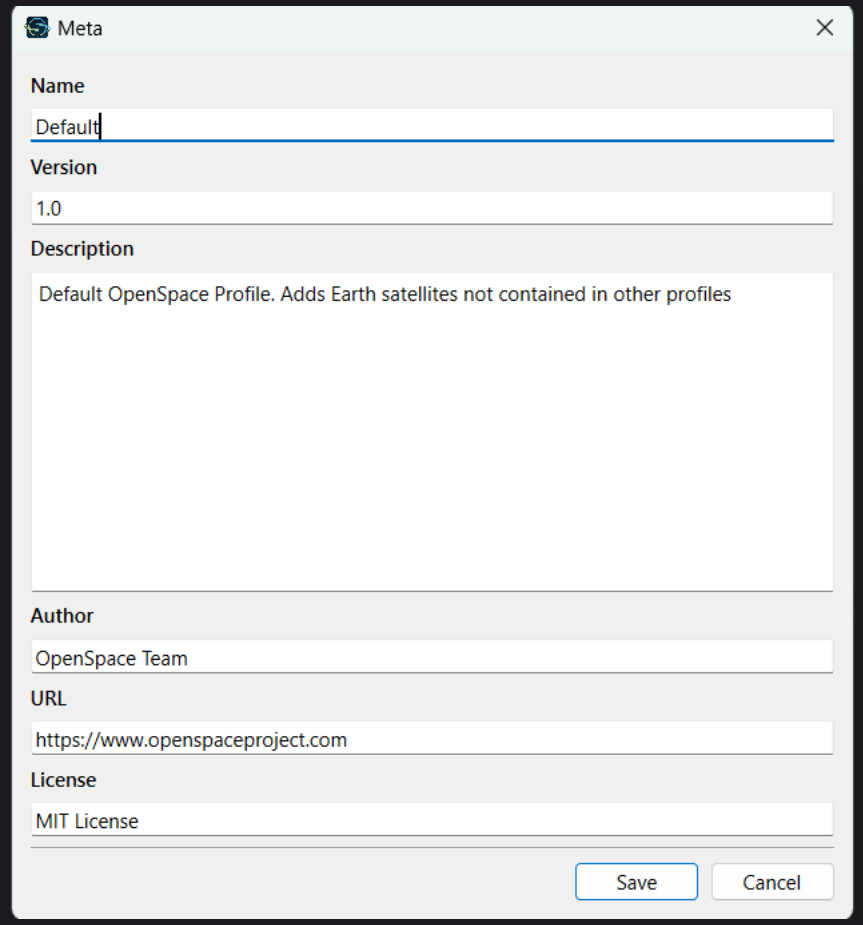
- Right side shows configuration for the selected keybinding
  - Modifier: Set which additional key you need to press in addition to the main key
  - Key: The main key on the keyboard that triggers the *action*
  - Action chooser: A drop-down menu where you can select an *action* that you created using the *profile editor*
  - Action: The *identifier* of an *action* that was either created using the *profile editor* or loaded via an *asset file*



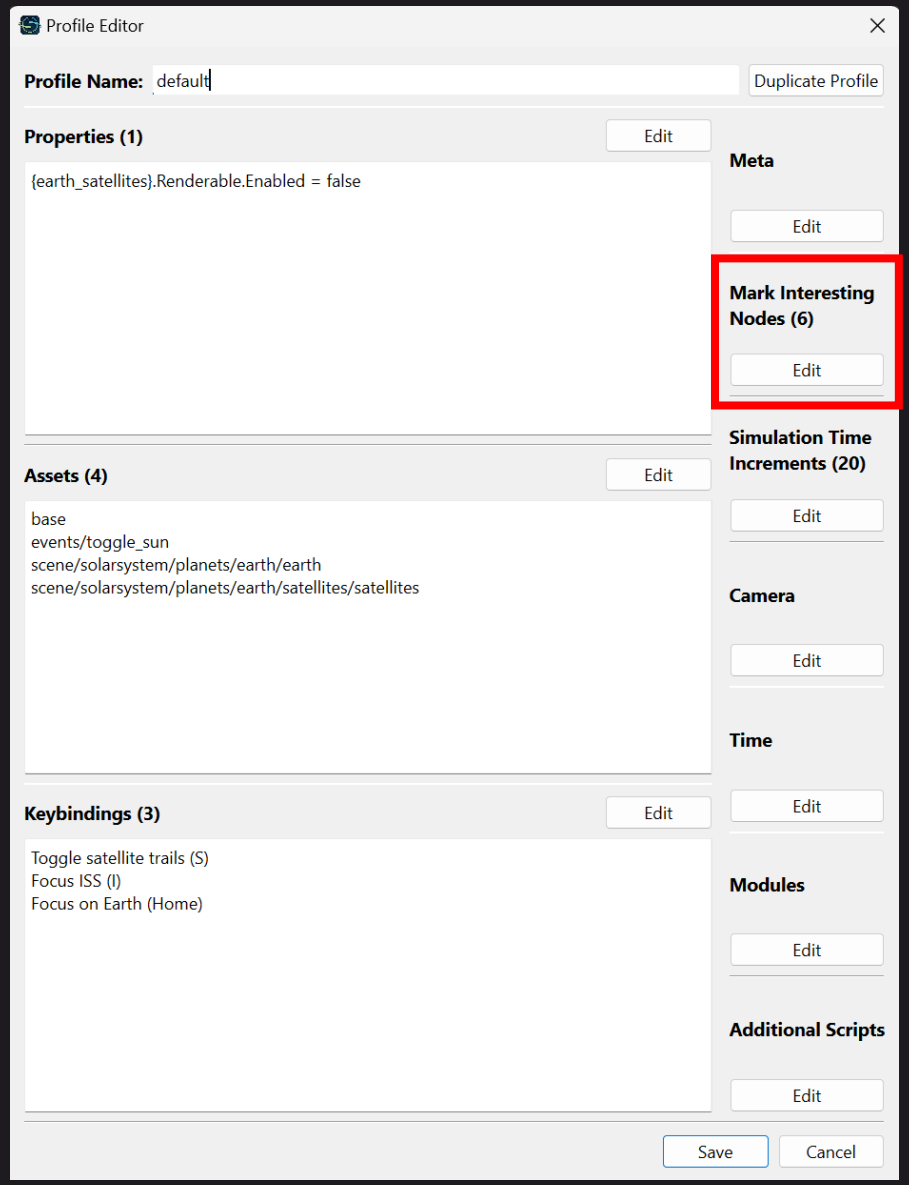
Profile Editor



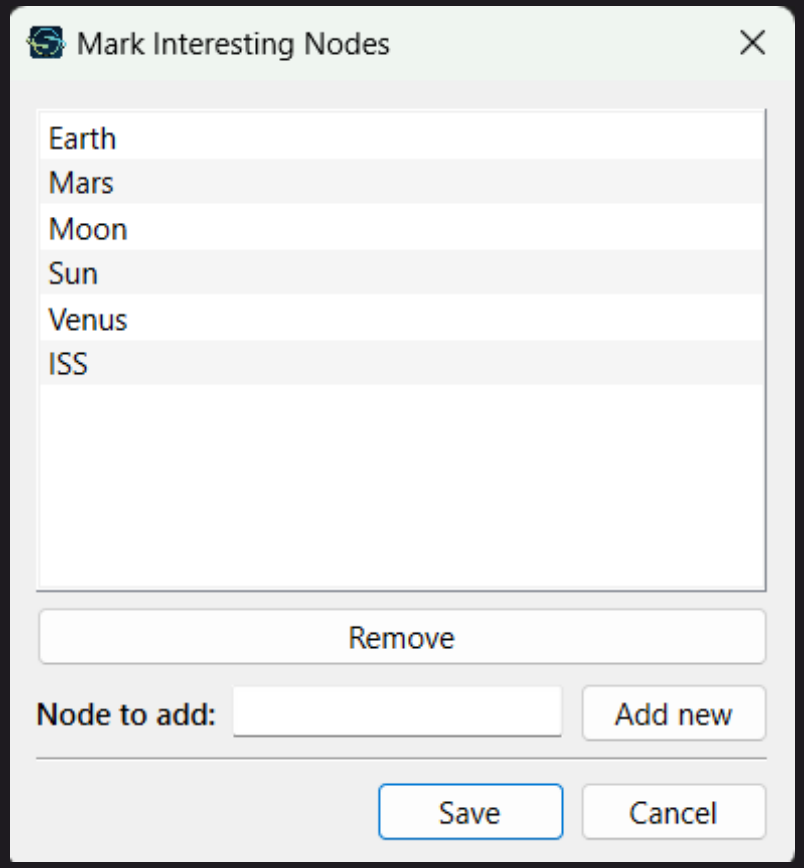
- Meta information about the Profile.
  - Who wrote it, what is its license, etc



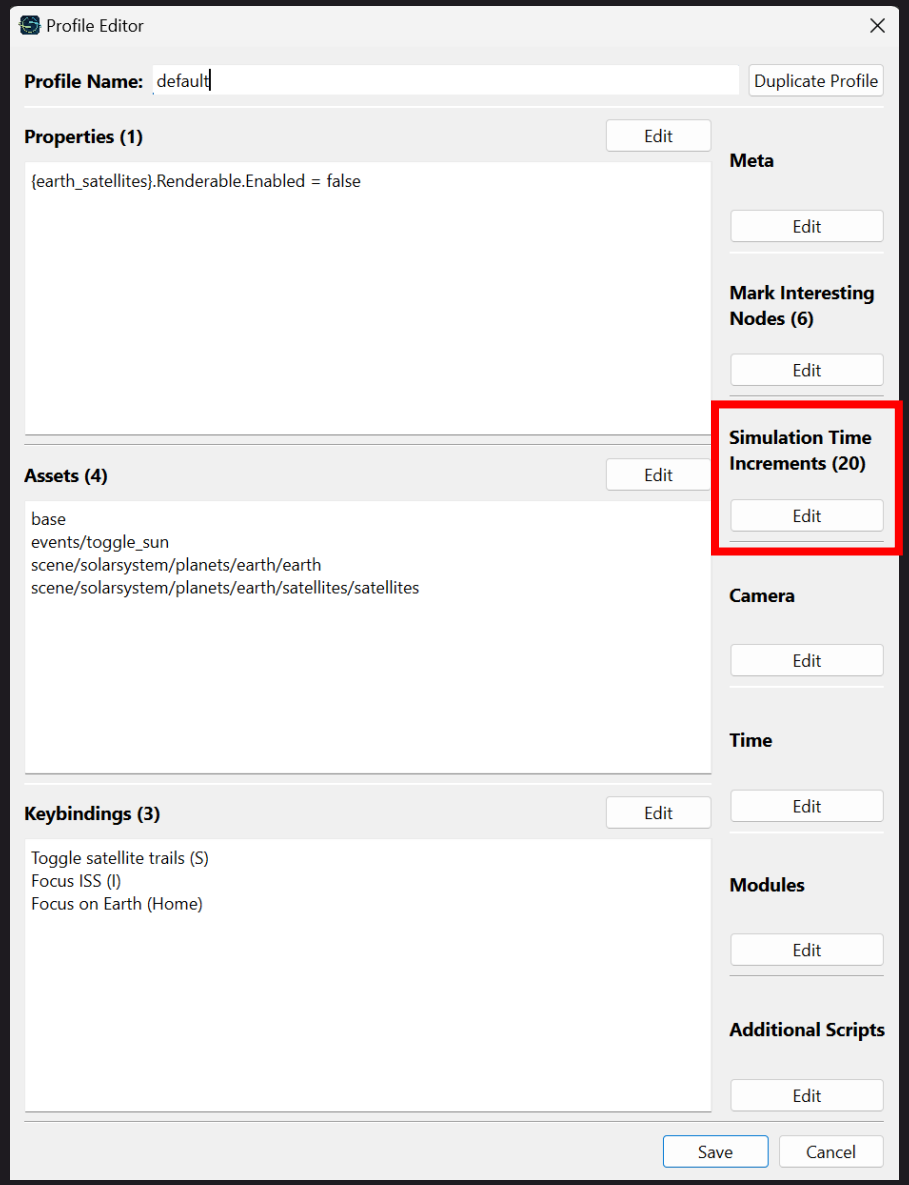
Profile Editor



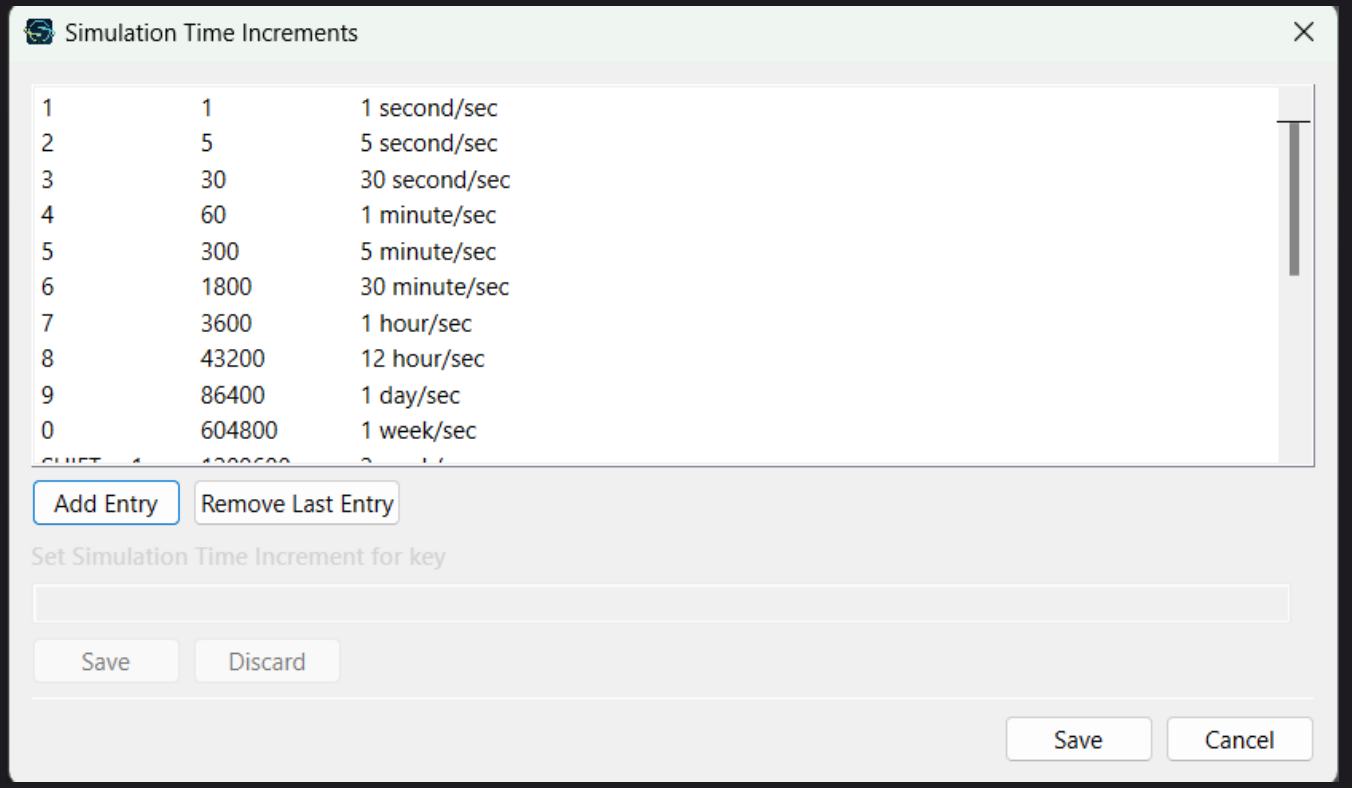
- Which *Scene Graph Nodes* should appear in the Camera Focus user interface



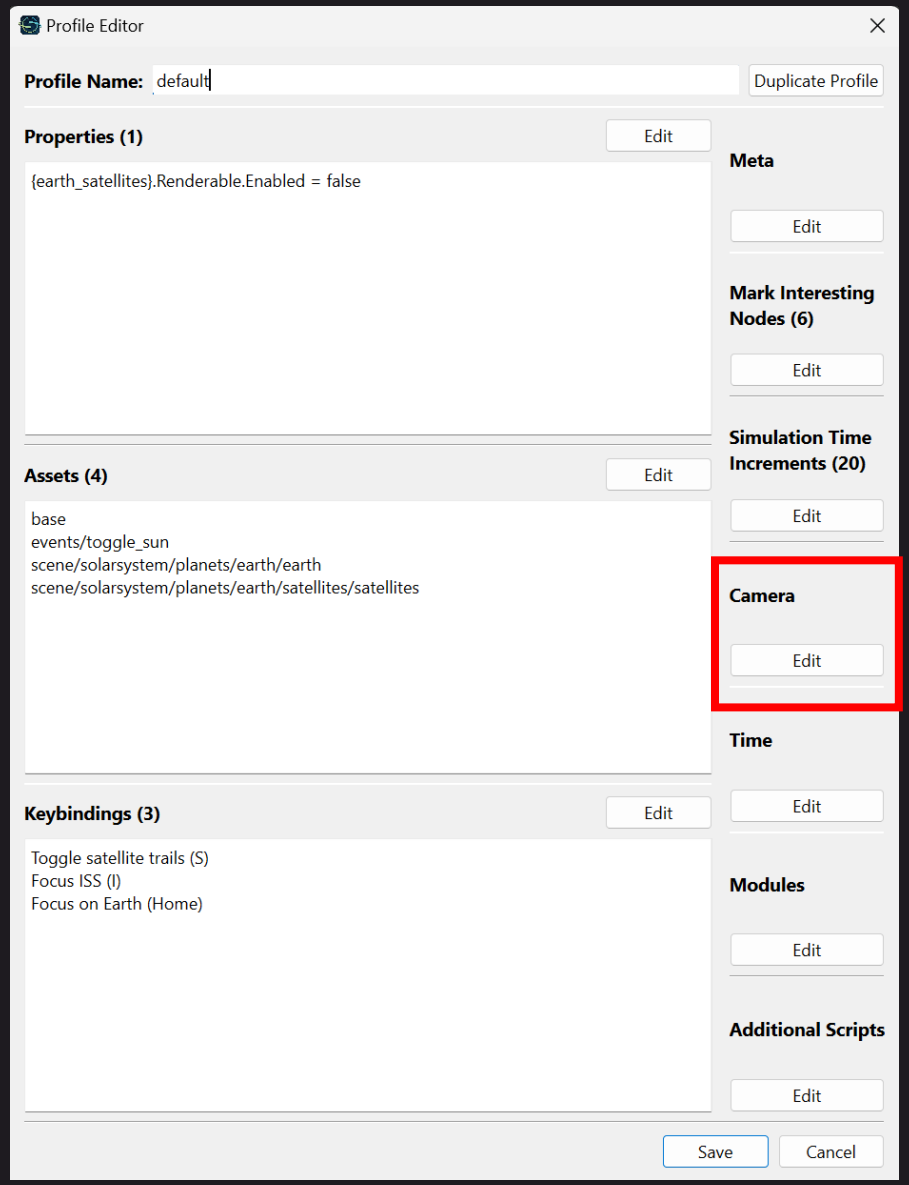
Profile Editor



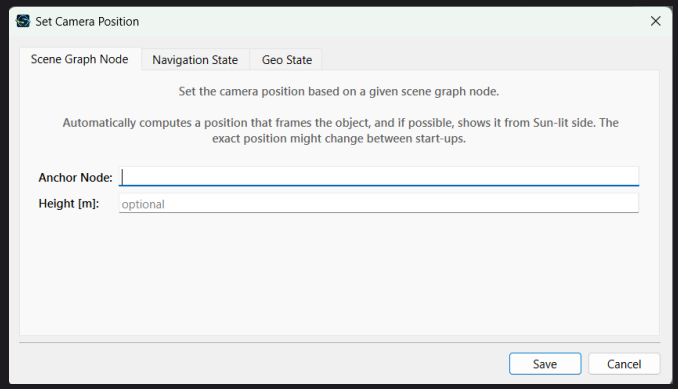
- Which are the *Simulation Time Steps* that should be available through the number keys

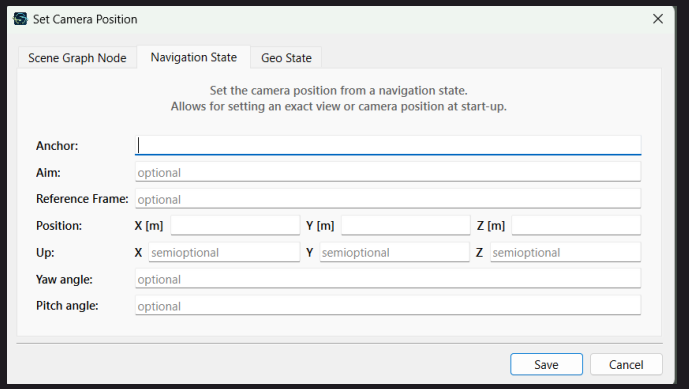


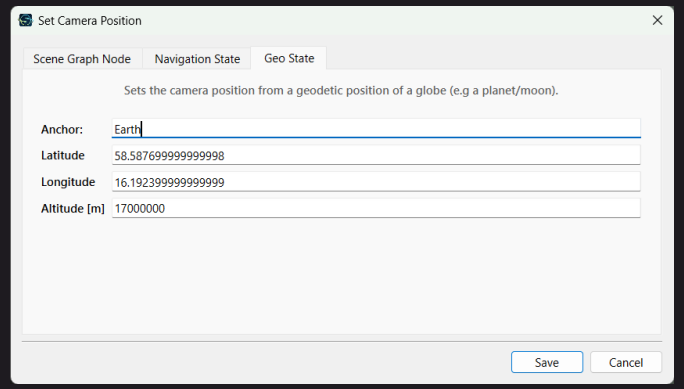
Profile Editor



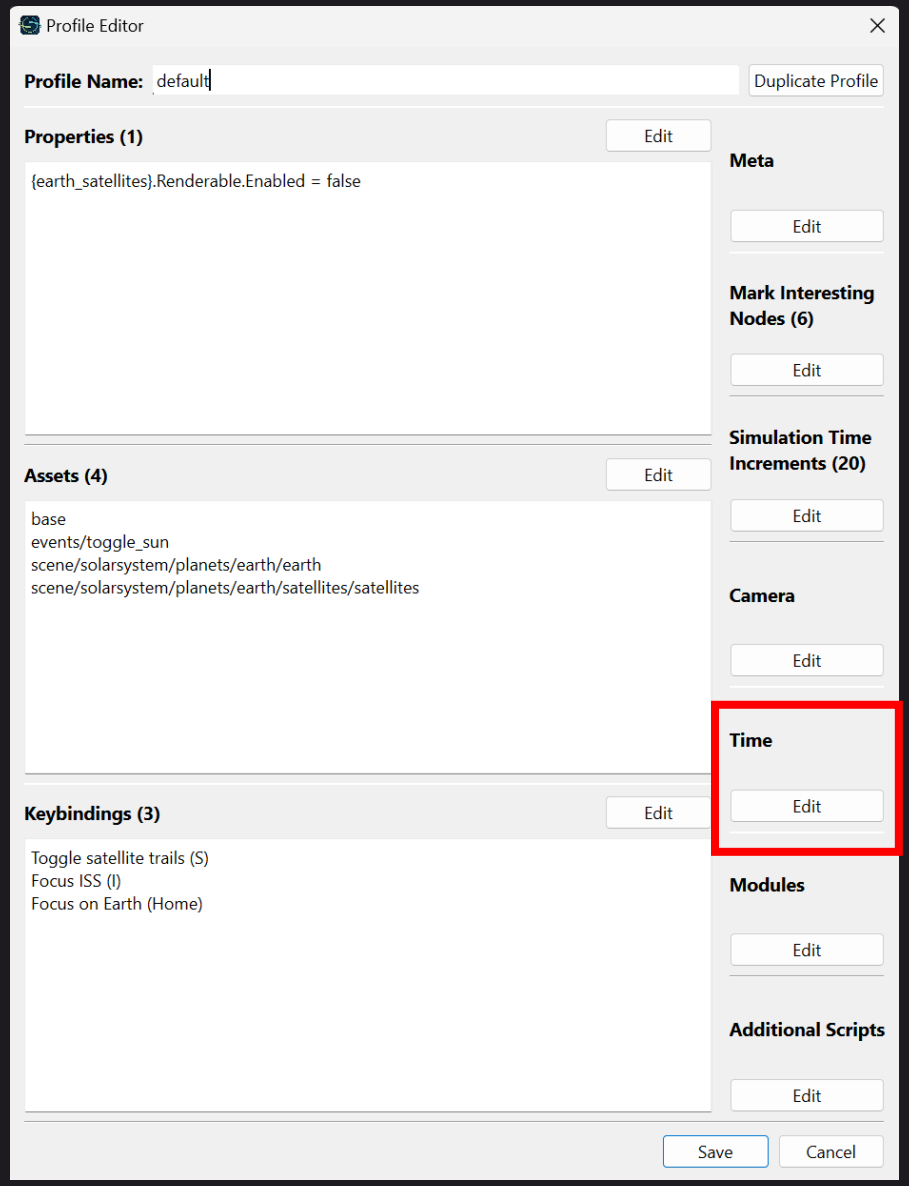
- The initial camera location when loading the Profile
  - Scene Graph Node: Places camera over a specific node
  - Navigation State: Places camera at a specific point with the option to also set camera viewing direction, etc
  - Geo State: Places camera over a specific location on a globe



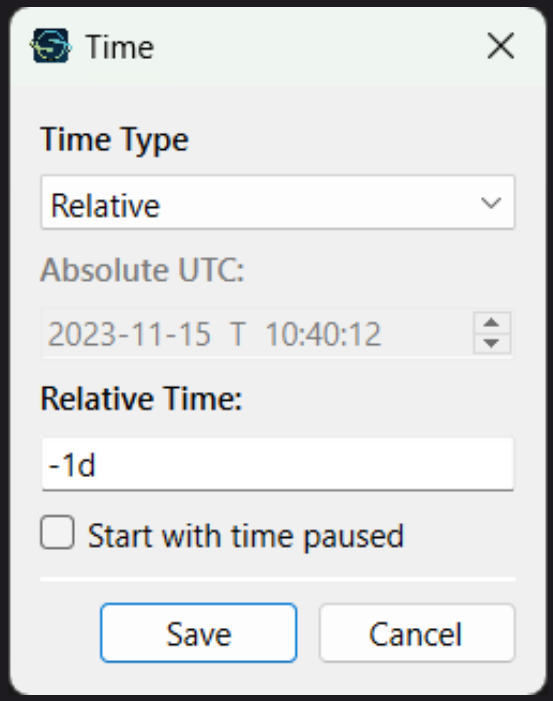


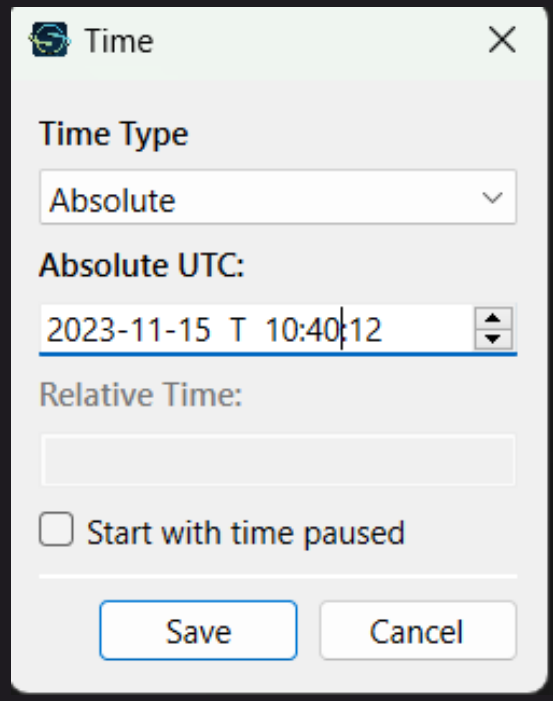


Profile Editor



- The *Simulation Time* when loading the Profile
- Can either be relative time or a specific date





Profile Editor

- Profile Name: The name of the *profile*
- Properties: The default setting that should be applied to the *Properties* in the scene
- Assets: All *assets* that the *profile* explicitly loads
- Keybindings: The keyboard shortcuts that are defined in this *Profile*
- Meta: Meta information about the Profile. Who wrote it, what is its license, etc.
- Mark Interesting Nodes: Which *Scene Graph Nodes* should appear in the Camera Focus user interface
- Simulation Time Increments: Which are the *Simulation Time Steps* that should be available through the number keys
- Camera: The initial camera location when loading the Profile
- Time: The *Simulation Time* when loading the Profile
- Additional Scripts: Additional Lua scripts that should be executed when the Profile is loaded

