



Modding Operational Instructions

from ISA's Quantum Artificial Intelligence Laboratory

Supervised by
Dr. Nathan Foster

===== TO BE READ BY ISA'S PERSONNEL ONLY =====

Version 1.7.1



[Modding Operational Instructions](#)

[Welcome!](#)

[The future is counting on you.](#)

[About this version](#)

[Supported features](#)

[First and foremost.](#)

[Setting the structure](#)

[Game Versions](#)

[Changing Game Versions](#)

[Embrace change.](#)

[Creating New Items](#)

[Stacking your mods](#)

[Overriding Pre-existing Items](#)

[Try everything.](#)

[Console Commands](#)

[General](#)

[Console Bindings](#)

[Gameplay](#)

[We're all in this together!](#)

[Uploading your mod](#)

[Downloading mods from Workshop](#)

[Come Join The Community!](#)

[Put your mind to work.](#)

[Annex I - Building YAML description](#)

[Buildings](#)

[Annex II - Resources YAML description](#)

[Resources](#)

[Annex III - Special Project YAML description](#)

[Special Projects](#)

[Annex IV - Languages](#)

[Languages](#)

[Extending included Languages](#)

[Annex V - Tech Tree](#)

[Lanes](#)

[New Lane Logic](#)





Welcome!

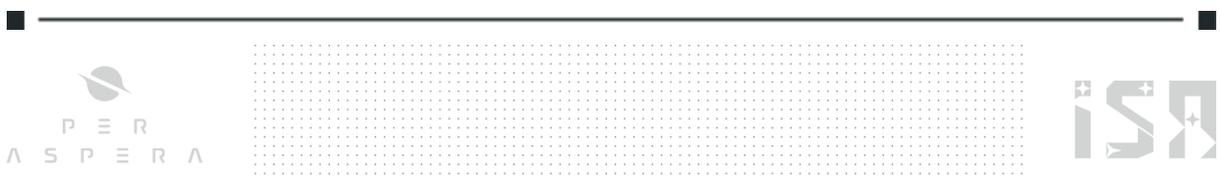
For the last 50 years, the International Space Agency has been breaking barriers to drive advances in science, technology, and space exploration for the benefit of the Oxy-UN. And now, as a new member of the Quantum Artificial Intelligence Laboratory, you can also be part of it!

The future is counting on you.

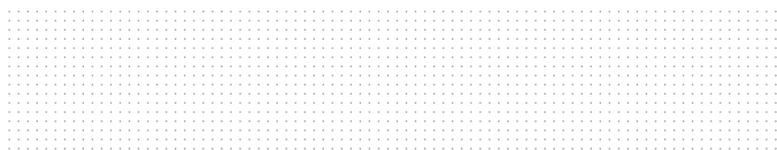
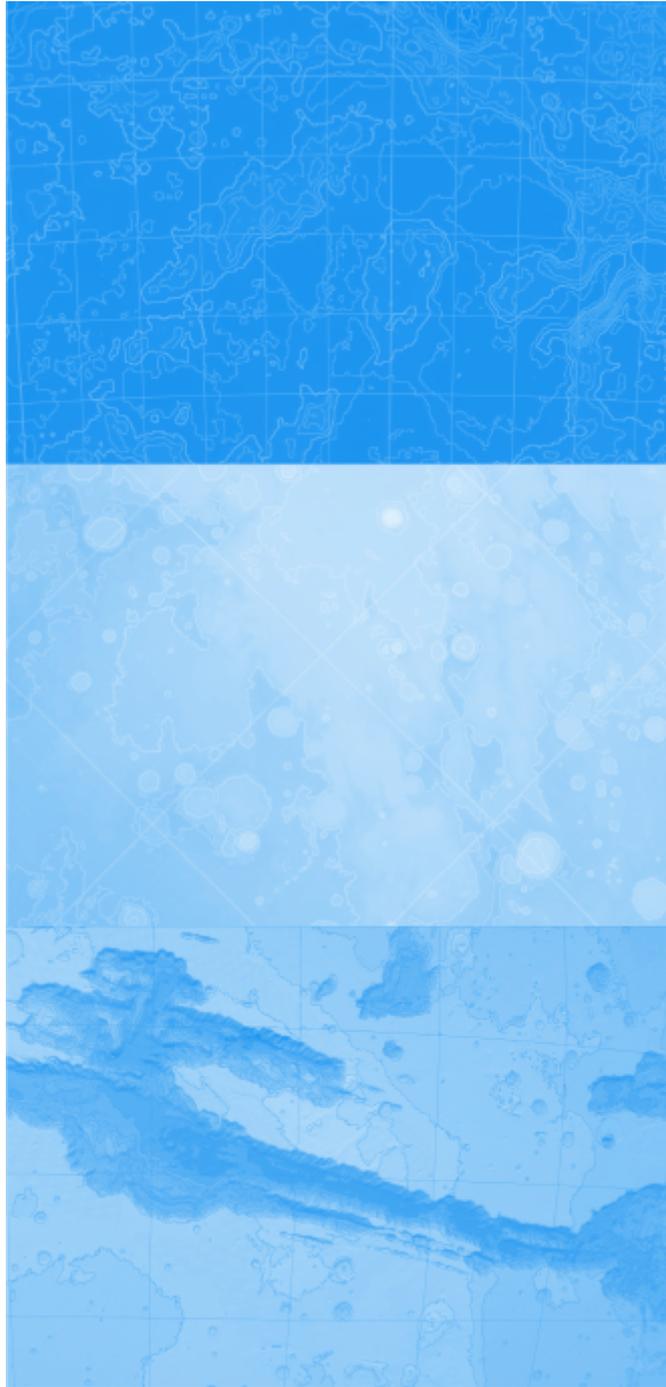
We are preparing for the unknown, but we are also forging what the relationship between humans and machines is going to be. A collaboration that with no doubt will define the future to come.

You have been provided with all the necessary tools to generate new and advanced ideas that can be used in our next Mars terraforming plan. In this manual, we will help you get started with mods.

Let's get started!









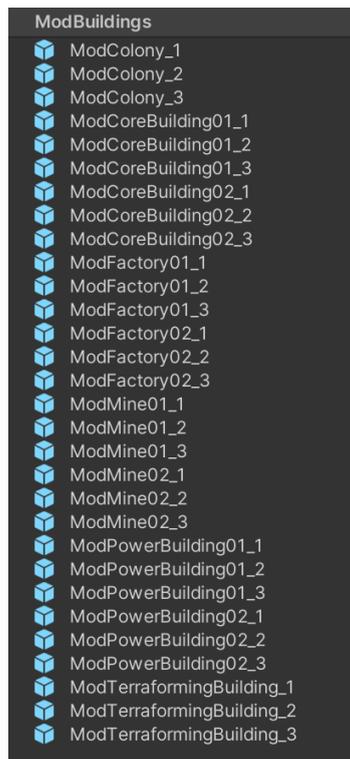
About this version

Supported features

Before diving into the world of modding Per Aspera, keep in mind that this version supports some specific features that are ready for implementation, but some others are not.

For instance, you will be able to add or modify buildings, tech-tree technologies, resources, quests, scatter zones, and knowledge base entries as much as you like. But, while it supports custom icons settings, it is **not** possible to add **new** custom prefabs or meshes for things like buildings, special projects, resources, or ways.

We added some extra building prefabs to help you identify your creations from those in the base game. Use the path “*ModBuildings/*” in the YAML file to assign the prefab to your work.



Although it is not officially supported, you can try as many combinations and modifications as you want. Inside the “data model” folder you will find all the YAMLS used in the base game. Check how the YAMLS for the campaign



and sandbox modes were created to know what can be configured in **setups** and **type lists**.



Warning: Using mods or console commands will permanently disable the achievements for the current save game.

First and foremost.

Each mod you create must be located in: “*{Per Aspera installation folder}\Per Aspera_Data\StreamingAssets\Mods*”

And remember that each YAML path defined is local to the mod root folder. E.g. “*Icons\MyIcon*” will search in “*{Per Aspera installation folder}\Per Aspera_Data\StreamingAssets\Mods\{Your mod folder}\Icons\MyIcon*” if the mod is local.



Setting the structure

You will need a folder containing the following file:

- manifest.yaml

The manifest.yaml is the entry point of the mod and defines various things:

- **modId:** it will be the unique identifier of the mod, so think it through. This will allow other mods to reference your mod as a dependency, and it will be its name when the mod is local.
- **requiredMods:** a list of “modIds” of those mods on which yours has a dependency.



- **optionalManifests:** a list of optional sub-manifests that will be loaded when their respective requiredMods are found. For example, adding a sub-manifest called “manifest-campaign.yaml” with “campaign” as requiredMods will only add that content if you are in campaign mode.
- **compatibleGameVersions:** a list of game versions compatible with this mod. It will also warn users if the mod is outdated.
- **setups:** Each one of these files is optional, and they define global variables, basic system parameters, or world initialization variables.

Surely you’ll be creative with all these choices!

- **type lists:** Each of these is also optional, and multiple files per entry are allowed. They define the “types” and their variables (Buildings, Resources, Techs, etc.).

Each entry can be set to replace any previously loaded types in that category (false works as additive).

The folder must contain all the YAML files referenced in the manifest file.



```
manifest.yaml  ✕
1 # Mod Id
2 modId: "ModExample"
3 # Compatible Versions
4 compatibleGameVersions:
5   - "1.3"
6   - "1.4"
7 # Dependencies
8 requiredMods:
9   - "RequiredMod_1"
10  - "RequiredMod_2"
11 # Setups
12   initialSetup: InitialSetup.yaml
13   droneSetup: DroneSetup.yaml
14   frontendSetup: FrontendSetup.yaml
15   planetSetup: PlanetSetup.yaml
16 # ...etc
17 # Type Lists
18 building:
19   filenames:
20     - building.yaml
21   replace: false
22 resource:
23   filenames:
24     - resource.yaml
25   replace: true
26 technology:
27   filenames:
28     - technology-engineering.yaml
29     - technology-space.yaml
30     - technology-biology.yaml
31   replace: true
32 # Optional sub-manifest
33 optionalManifests:
34   - optional-manifest.yaml
```

```
optional-manifest.yaml* ✕
1 # This mod will only load if mode is campaign
2 modId: "CampaignModExample"
3 compatibleGameVersions:
4   - "1.4"
5 requiredMods:
6   - "Campaign"
7 building:
8   filenames:
9     - newCampaignBuildings.yaml
10  replace: false
```

Game Versions

The Game Version is composed of the first three numbers that appear on the upper left corner of the main menu.



v.1.3.0.0.6.12471.2021.5.12

For example, in the picture above, it would be “version 1.3.0”.

Changing Game Versions

Sometimes you might want to change to a previous version of the game. Here are the steps to change the game’s branch.

- **Steam:**
 1. Find “Per Aspera” in your Steam Library and access its properties by right-clicking and selecting “Properties”.
 2. Navigate to the “betas” sub-menu.
 3. In the drop-down menu, choose the branch of your preference.
- **GOG:**
 1. Find “Per Aspera” in your GOG Galaxy Library, then, next to the “Play” button, access the “More” drop-down menu.
 2. In the “More” menu, go to “Manage Installation” and right-click “Configure”
 3. Open the “Beta Channels” drop-down menu.
 4. Choose the branch of your preference.

That’s it! Your game should begin with the selected version.

Embrace change.

We must continue to grow and adapt to the unexpected and improve what we have learned to finally conquer the Red Planet.

Creating New Items



You can add new items to the game, create your collection, or add them to the existing ones.

Say you want to create a new “power” building that does not produce energy but only expands the power range. To do this, you must:

Create the `building.yaml` file with the desired settings:

```
building.yaml ↗ ✕
1  building_power_extension:
2    categoryType: !buildingCategory category_power
3    compactName: PowExt
4    description: A building that extends the power range
5    maxHealth: 75.0
6    name: Power Extension
7    powerConsumption: 0.0
8    powerPriority: 1.0
9    extendsPowerCluster: true
10   prefabName: WorkerRelay
11   rubblePrefabName: RubblePile_S
12   requiredConstructionResources:
13     !resource resource_aluminum: 1
14   reservedRadius: 20.0
15   iconName: BuildIcons/Icon_WorkerRelay
16   jumpRadius: 100.0
```

And add it to your mod manifest building type list:

```
manifest.yaml ↗ ✕
1  modId: "Power Extension"
2  compatibleGameVersions:
3    - "1.3"
4  building:
5    filenames:
6      - building.yaml
7    replace: false
```

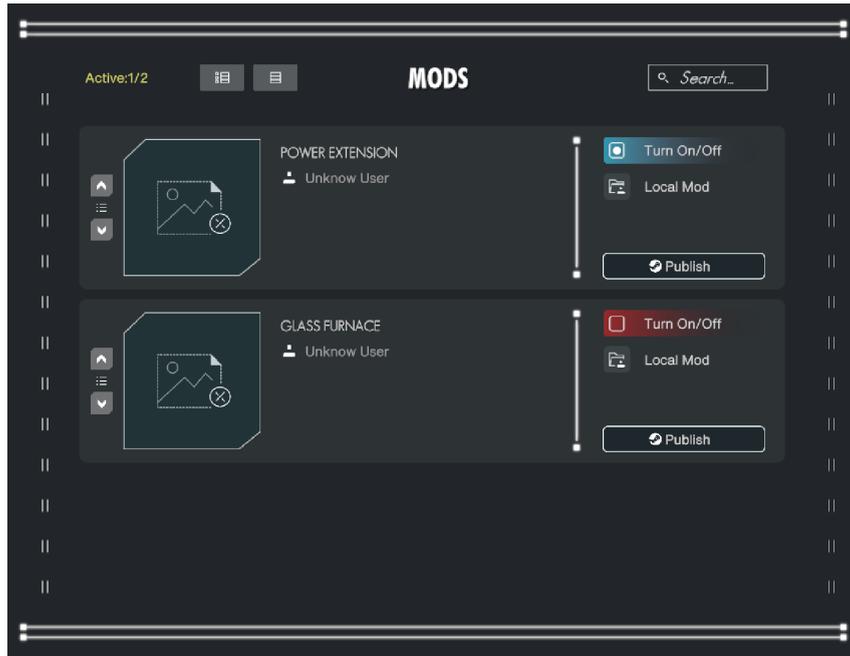
Eager to test your creation? Unlock the new item (building, technology, resource, etc.) in the game by using the proper [console command](#):



We recommend that you use notepad++ and select Yam1 as a language. But be wary, tabs are not the same as spaces in these kinds of files.

☰ Stacking your mods

You can add multiple mods simultaneously. The enabled mods will be stacked and loaded from top to bottom, allowing you to override previous mods or the base game itself.



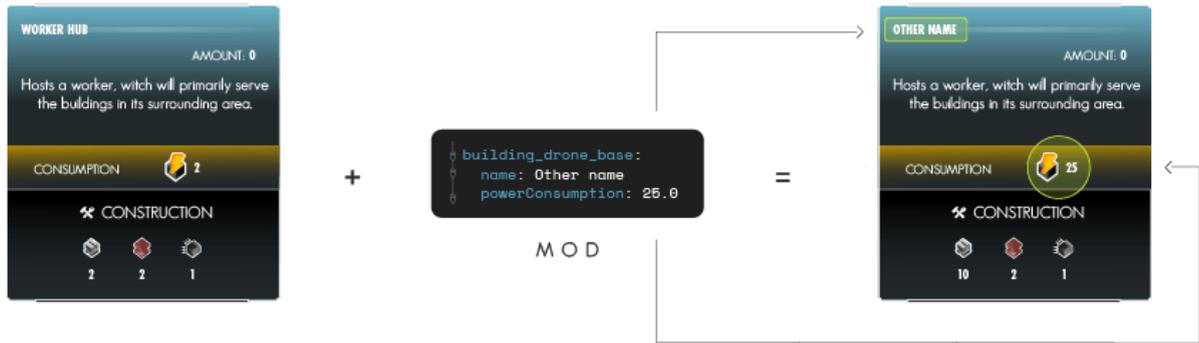
Overriding Pre-existing Items

Overriding specific fields of an item instead of creating new ones will allow you to edit the base game or other mods by Mod stacking.

To do this, add an item in your mod using the same key as the item you want to modify. For example, if you need to change the Worker Hub, add a new object in the mod whose key is also `“building_drone_base”`. The changes made to it will override the base game or any previous mod.

Here are some examples of the type of fields that you can override:

- Edit strings and numbers to modify specific properties of the overridden item. The unedited fields will remain the same.



- Modify an entry that already exists in a **dictionary**, replacing it with a new one.



- Add a new item to a collection.





- Modify a collection using the **!replace** tag to override it entirely.





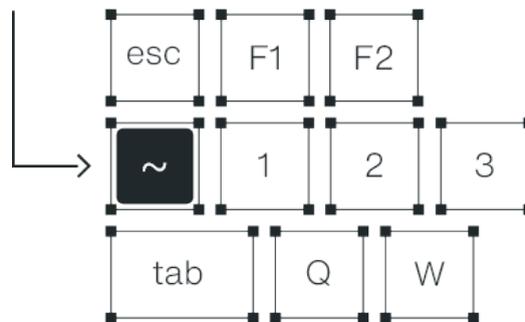


Try everything.

As you have read on these pages, there are many things you can try to ensure human presence on Mars. And you may want to test your ideas before sharing them with others.

Console Commands

You can do this by accessing the console and entering a command line. When playing the game, just press the ~ key and type the desired command.



Here we share a list of commands that you can use to test your mods. But beware, achievements won't compute for the working save game once you enter one of them. So, use them properly!

General

- **Commands** - Lists all available console commands, with an optional filter string.
- **Help** - Shows help for the given command.
- **History** - Shows command history with optional last count.
- **TakeScreenshot** - Takes a screenshot, with an optional resolution multiplier.
- **SetEngineTimescale** - Sets engine timescale.



Console Bindings

- **Bind** - Binds a specific KeyCode with a Console command. E.g. Bind U UnlockAllBuildings.
- **SaveBindings** - Saves current bindings to a file.
- **Unbind** - Unbinds a previously bound KeyCode from a Console command.
- **UnbindAll** - Unbinds all current bounded commands.
- **ShowBindings** - Shows current bindings.

Gameplay

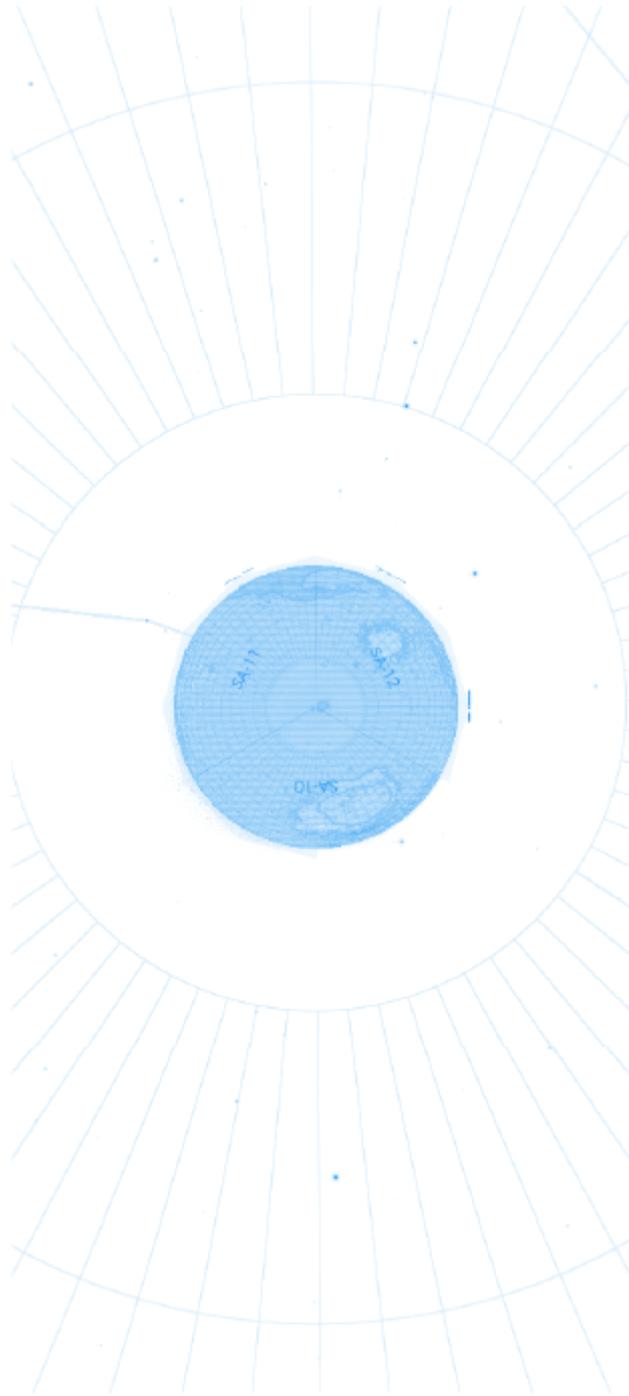
- **AdditionalLandingSite** - Adds a new landing site to the player's faction.
- **AdditionalOrbitalBuilding** - Adds a new orbital building to the player's faction by code.
- **AddYearsSinceStart** - Adds a specific number of years to the current time.
- **BuildingAddResource** - Adds stocks of specified resources in the selected building.
- **BuildingSetHealth** - Sets the relative health of the currently selected building.
- **BuildingSetVeinQuantity** - Sets the amount of a resource for the selected building.
- **BunchOfResources** - Adds a bunch of non-virtual resources to the selected building or the main building.
- **DespawnAllHazards** - Deactivates all hazards on the planet.
- **FactionAddResearchPoints** - Adds research points to the selected faction.
- **FactionAddResourceDistributed** - Adds stocks of specified resources spread across all buildings of the selected faction.
- **FactionDisableEnhancement** - Disables the specified enhancement for the selected faction.
- **FactionEnableEnhancement** - Enables the specified enhancement for the selected faction.
- **FactionKillMilitaryDrones** - Kills all military drones of the selected faction.
- **FactionList** - Lists all factions.



- **FactionListDrones** - Lists all worker drones and their information for the selected faction.
- **FactionListEnhancements** - Lists all enhancements for the selected faction.
- **FactionListMaintenanceDrones** - Lists all the maintenance drones and their information for the selected faction.
- **FactionSelect** - Selects a faction as a target for faction commands.
- **FactionSpawnDrones** - Spawns a specific number of worker drones for a selected faction. If a building is selected, spawns worker drones in that building.
- **FactionSpawnMaintenanceDrones** - Spawns a specific number of maintenance drones for the selected faction.
- **FactionSpawnMiliDrones** - Spawns a specific number of military drones for the selected faction.
- **FillPopulation** - Increases the population of all colonies to their maximum capacity.
- **FinishConstructions** - Immediately finishes pending constructions.
- **FinishUpgrades** - Immediately finishes pending upgrades.
- **IncreaseGHG** - Increases the GHG pressure on Mars by a specific amount.
- **IncreaseN2** - Increases the pressure of N2 on Mars by a specified amount in mbar.
- **IncreaseO2** - Increases the pressure of O2 on Mars by a specified amount in mbar.
- **IncreaseWater** - Increases the amount of liquid water on Mars by a specified amount in meters. The change is not immediate, it will have a maximum delta per day as configured in PlanetSetup until reaching the target level.
- **Kill** - Kills the selected building or worker drone.
- **ListDialogues** - Lists all the dialogues.
- **ListPersons** - Lists the 'dialogue persons' and their info.
- **ListSectors** - Lists the player's sectors and their status.
- **RevealAllCraters** - Reveals all the usable craters on the planet.
- **RevealAllResourceVeins** - Reveals all the resource veins on the planet.
- **RevealAllSpecialSites** - Reveals all the special sites on the planet.



- **RevealScannerArea** - Reveals a scanned area with a radius centered at the mouse position.
- **RevealSpecialSite** - Reveals a specific special site by key
- **RunInteractionRule** - Runs a specific interaction rule by key
- **RunTextAction** - Runs manually a specific TextAction
- **SetDaysSinceStart** - Sets the current date in days, counted from the game start
- **ShowBlackboard** - Prints blackboard vars to console
- **ShowCurrentTechPopup** - Forces to show the popup of the tech that is currently researching
- **SpawnAsteroid** - Spawns an asteroid at the mouse position.
- **SpawnDevil** - Spawns a devil on mouse position.
- **SpawnRivalBase** - Spawns a base for the rival faction.
- **SpawnSandstorm** - Spawns a sandstorm on mouse position.
- **StartDialogue** - Starts a specific dialogue by key.
- **UnlockAllBuildings** - Unlocks all buildings, with an optional parameter for filtering by upgrade level.
- **UnlockAllKnowledges** - Unlocks all knowledge.
- **UnlockAllSpecialProjects** - Unlocks all special projects.
- **UnlockAllTech** - Unlocks all technologies.
- **UnlockAllWays** - Unlocks all ways.
- **UnlockBuilding** - Unlocks a specific building by key.
- **UnlockMilitaryBuildings** - Unlocks all military buildings.
- **UnlockQuest** - Unlock a specific Quest by key.
- **UnlockSector** - Unlocks a sector by index.
- **UnlockSpecialProject** - Unlocks a specific special project by key.

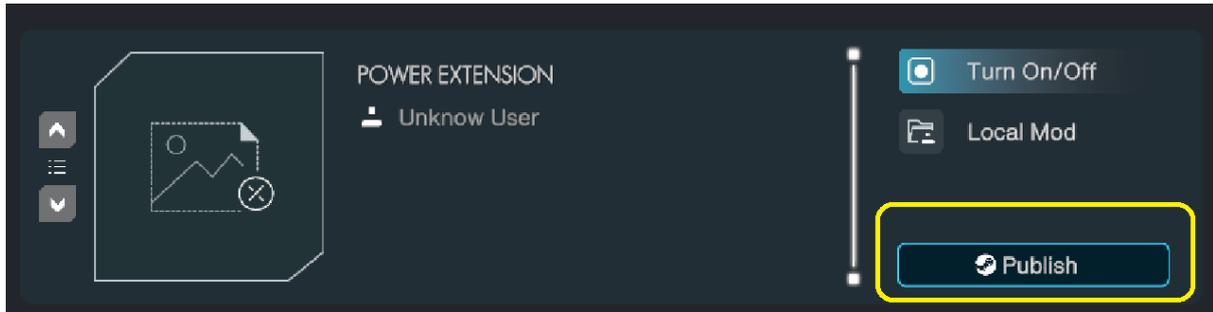




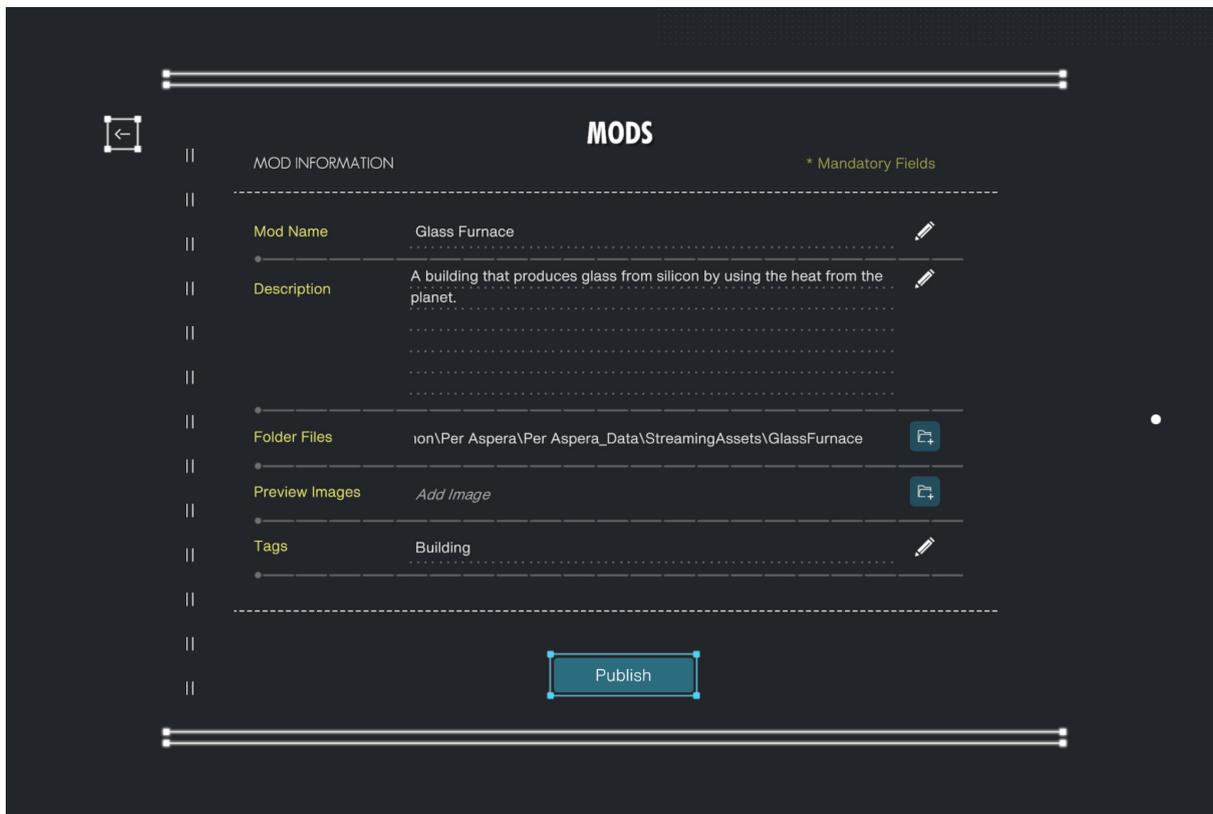
We're all in this together!

📁 Uploading your mod

Once you are done creating your mod, and you're ready to share it with others, upload it to Steam by pressing the "Publish" button in the Mods menu.



It doesn't take much to be a good team player. Please, attach additional information to your mod so other players can understand all the fuss about it.

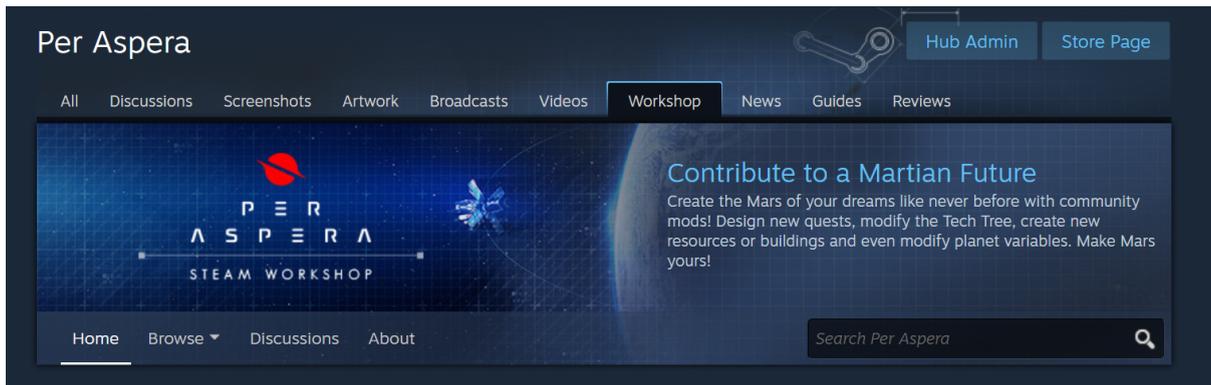




Press Publish, and share it with the world! Your mod will be ready to be downloaded from the Steam Workshop.

Downloading mods from Workshop

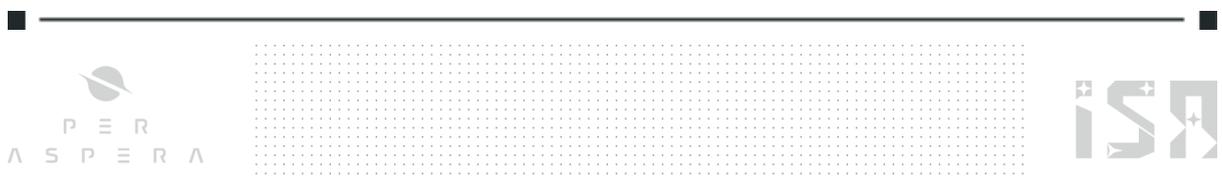
You can also download mods from other users from the [Steam Workshop](#).



Explore the workshop, find a mod that you like, and hit the Subscribe button. Once you restart the game, the selected mod will appear in the Mods menu.

Come Join The Community!

We encourage each one of you to share your creations, exchange ideas, and ask other players for feedback. Be part of our modding community by joining the [#per-aspera-modding](#) channel on the Raw Fury Discord Server.





Put your mind to work.

If you got to this point, it means you know everything you need to know to create a Mod. Put your mind to work, create some amazing new scenarios for ISA's terraforming plan, and push the boundaries of human potential!

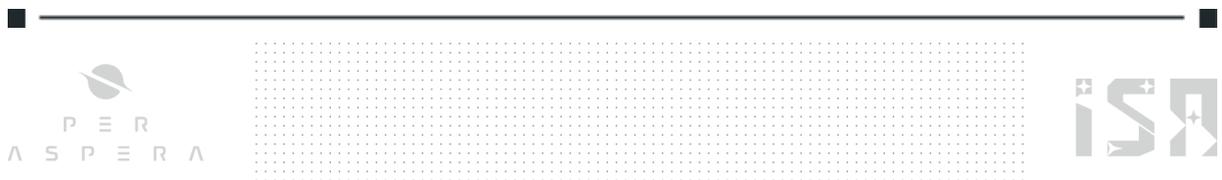


- Ad Astra





===== TO BE READ BY ISA'S PERSONNEL ONLY =====





Annex I - Building YAML description

Buildings

```
building_maintenance_facility:
  categoryType: !buildingCategory category_core
  compactName: MainFac
  description: BE_building_maintenance_facility_description
  droneCapacity: 1
  healthLossPerDay: 0.05
  inputResources:
    !resource resource_polymers: 1
    !resource resource_electronics: 1
  knowledge: !knowledge knowledge_maintenance_facility
  maxHealth: 100.0
  name: BE_building_maintenance_facility_name
  outputQuantity: 3
  outputResource: !resource resource_repair_drone
  powerConsumption: 20.0
  powerPriority: 2.0
  prefabName: MaintenanceFacility
  rubblePrefabName: RubblePile_S
  progressPerDay: 0.02
  requiredConstructionResources:
    !resource resource_aluminum: 3
    !resource resource_steel: 3
    !resource resource_electronics: 3
    !resource resource_polymers: 3
  requiredResourceVein: null
  reservedRadius: 20.0
  waySnapRadius: 10.0
  rivalIconName: BuildIcons_Rival/Icon_MaintenanceFacility
  iconName: BuildIcons/Icon_MaintenanceFacility
  progressBarNames:
    - Progress_Torus
    - Progress_Cylinder
building_water_plant:
```

- **name:** Name of the building or the name's key.
- **description:** Description of the building or the description's key.
- **categoryType:** The building category, which will be displayed in the building panel and the structures panel.
- **knowledge:** The knowledge that is unlocked in the knowledge base when the building is unlocked.
- **inputResources:** The type and amount of resources required to meet the building's production cycle.
- **extractionLevel:** It is only used for mines and defines what level of veins can be mined by this building.
- **requiredResourceVein:** Vein required to place this building. If the building produces output, it will also consume one unit of the required vein in each production cycle.
- **scalesToResourceVein:** Indicates if a building mesh should scale to adapt to the underlying required vein size. Used in crater buildings.



- **outputResource:** The type of resource that this building will produce in each production cycle (except for gases, in that case, gasReleaseAmount must be used).
- **outputQuantity:** The amount of resource this building will produce in each production cycle. Requires outputResource.
- **progressPerDay:** Defines how much of the production cycle progress per day. Each production cycle ends when it reaches 1.
- **requiredConstructionResources:** Resources required for constructing this building.
- **initialStock:** Resources that will spawn in the building when added.
- **maxHealth:** Maximum health of the building.
- **healthLossPerDay:** Health lost each day by decay.
- **jumpRadius:** Max distance for connecting to other buildings.
- **prefabName:** Mesh used by the building in close view.
- **rubblePrefabName:** Mesh used by the building when destroyed.
- **iconName:** Icon used by the building in far view and HUD.
- **rivalIconName:** Icon used by the building if it's not from the player faction. If not assigned, it uses the same icon as the player.
- **orbitalIconName:** Icon used by the building in orbital view.
- **emptyHubIconName:** Icon used by the building if it's a drone hub and it's empty.
- **progressBarNames:** Progress indicators that are used for production cycles when hovering a building.
- **textureOverrides:** List of texture override data for changing building textures. Each override data it's composed of an **index**, a **property** (albedo, normal, metallic, emission), and a **texture** path of the material that will be changed, as well as whether the texture **isLinear** or not.
- **colonistCapacity:** Maximum number of colonists that can live in this building. The building must generate research points.
- **maxSolarPowerProduction:** Solar power produced by this building.
- **maxEolicPowerProduction:** Eolic power produced by this building.
- **maxThermalPowerProduction:** Thermal power produced by this building.
- **maxFissionPowerProduction:** Fission power produced by this building.
- **maxFusionPowerProduction:** Fusion power produced by this building.
- **powerConsumption:** Energy requirement of this building to be operational.



- **powerPriority:** Priority for this building to receive energy when it has an energy deficit. Higher values mean higher priority.
- **energyStorageCapacity:** Maximum storage capacity of this building. It will store surplus energy in the building's cluster for use when needed.
- **extendsPowerCluster:** Enables this building to expand the electrical network, without necessarily generating energy.
- **powerClusterRadius:** Reach of this building in the electrical network. Requires to produce energy, and must have energyStorageCapacity or extendPowerCluster.
- **availability:** Defines that the building will not be part of the rival faction's base.
- **militaryDroneCapacity:** Amount that this building adds to the faction's maximum military drone capacity.
- **defensiveMissilesPerDay:** Number of defensive missile launches per day.
- **missileRange:** Range of the defensive missiles.
- **missileDamage:** The damage
- **scannerScanTime:** Days needed by this building to display a scanned tile at base speed.
- **scannerTimerFactor:** Factor of scannerScanTime added to scanner reveal time for each tile distance.
- **spawnRevealRadius:** The radius of scanned tiles that are revealed immediately when the building is built. Requires isBaseStarter.
- **isUpgradeTo:** List of other buildingTypes to which this building is an upgrade. Currently only supports one building.
- **maintenancePriority:** Priority of this building to receive maintenance. Higher values mean higher priority.
- **extendsMaintenanceCluster:** Enables this building to expand the maintenance network, without necessarily generating maintenance drones.
- **maintenanceClusterRadius:** The radius that the building contributes to the maintenance network. Requires the production of maintenance drones or extendsMaintenanceCluster.
- **enablesHyperloopConnection:** Enables this building to support hyperloop connections.
- **maxStorageCapacity:** Enables this building to have the specified storage capacity and act as a storage center.



- **hasSpacePort:** Enables this building to launch special projects, and to access the space view.
- **isBaseStarter:** Enables this building to be built immediately, and to generate all the resources necessary to start a new base. Spawns a drone, reveals an area around it, and places the minimum needed veins possible.
- **canSpawnLichenAndPlants:** Allows the building to generate lichen and plants when it is operational.
- **canSpawnCyanobacteria:** This allows the building to generate cyanobacteria when it is operational.
- **reservedRadius:** Minimum distance a building requires to be placed before colliding with another building's reserved radius.
- **co2ConversionPerDay:** Amount of CO₂ to O₂ conversion per day.
- **gasReleaseAmount:** Amount of gas released as outputResource. The outputResource must be one of the admitted gases (oxygen, nitrogen, carbon dioxide, GHG).
- **needsEquatorialStrip:** If enabled, this building should be placed near the equatorial strip.
- **needsPressure:** If enabled, this building will require minimal pressure on the planet to be placed. This pressure can be configured in PlanetSetup.humanMinimumPressure.
- **needsBreathableAtmosphere:** If enabled, this building will require a minimum of O₂ and a maximum of CO₂ pressure on the planet to be placed. This pressure must be configured in PlanetSetup.humanMinimumOxygenPressure and PlanetSetup.humanCO₂Toxicity.
- **orbitalPlacement:** Enables this building to be placed from orbit as a landing site, not from the building panel.
- **waterNeededFactor:** Minimum water level needed near the building to be operational. It needs waterNeededRadius.
- **waterNeededRadius:** Radio sampled to check the water for waterNeededFactor.
- **pipesConsumption:** Minimum amount of water required from the water supply network for the building to function optimally.
- **extendsPipesCluster:** Enables this building to expand the water supply network, connecting the water treatment plants to the water consumption buildings.



- **pipesClusterRadius:** The radius that the building adds to the water supply network.
- **pipesProduction:** Amount of water produced by the building.
- **isWorkerHub:** Enables this building to host a worker drone.
- **isDistrictHub:** Enables this building to create a district and be its district administrator.
- **deforestationRadius:** Radius on which the building can clear the surrounding forests or plants when operative.
- **deforestationTime:** The days needed for this building to clear a patch of the surrounding forests or plants.
- **isAnimalSanctuary:** Enables this building to spawn animals.
- **nodeType:** Setting this to Port will allow this building to connect to the ocean, while setting to Aquatic will allow placement on the ocean.
- **needsEvenTerrain:** If disabled, will allow placement of the building regardless of irregular terrain.
- **lookToCoast:** If enabled, forces the building model to face the water.
- **lookToCoastMaxDistance:** Maximum distance a lookToCoast building can be from the water.
- **shipCapacity:** Maximum number of ships a port can hold.
- **isAquaticVersionOf:** When set to another building key, hides the aquatic version from the build menu and allows the aquatic version to be placed when placing the other building on water.
- **clearOutQuantity:** Sets the number of resources that should be removed when a mine or factory has a stockpile above the threshold.
- **clearoutThreshold:** Sets the percentage of max capacity at which a mine or factory should clear out the quantity of resources to available storage.



Annex II - Resources YAML description

Resources

```
resource_water:  
  color: 397099  
  index: 2  
  knowledge: !knowledge knowledge_resource_water  
  materialType: Mined  
  name: BE_resource_water_name  
  prefabName: Water  
  veinIconsName:  
    - Resource Icons/Veins/Water_Vein  
    - Resource Icons/Veins/Water_Vein2  
    - Resource Icons/Veins/Water_Vein3  
  iconName: Resource Icons/Water  
  cubeMaterial: ResourceCube
```

- **color:** resource color in RRGGBB hexadecimal value.
- **materialType:** Way the resource is created
 - **Mined:** by extracting from the ground.
 - **Manufactured:** created by a factory
 - **Placement:** Special landmark on the terrain.
 - **Released:** miscellaneous.
- **name:** Resource name or localization key.
- **disabledWhenFlooded:** True by default. When it's False, it shows the resource vein while it's still underwater.





Annex III - Special Project YAML description

Special Projects

```
my_special_project:
  description: BE_my_special_project_description
  criterions: []
  defaultVisible: true
  index: 14
  launchType: Single
  name: BE_my_special_project_name
  iconName: my_special_project.PNG
  position:
    x: 0
    y: 0
    z: 0
  portProjectType:
    requiredDevelopmentResources:
      !resource resource_aluminum: 10
      !resource resource_parts: 10
      !resource resource_electronics: 10
      !resource resource_chemicals: 10
    requiredDevelopmentTime: 300.0
  launchActions:
    - command: CometDown
      arguments:
        - PlayerFaction
        - 10.0
  specialProjectController:
    !SpecialProjectControllerBase
  spaceportCardPrefab: "SpaceportProjects/SpecialProjectMyProject"
  spaceportCardPrefabPerBuilding:
    building_space_elevator: "SpaceElevatorProjects/SpecialProjectCom
```

- **name:** Project name or localization key.
- **launchType:** Launch type for the project.
 - Single: One-time project
 - Multiple: Requires multiple launches to complete
 - Continuous: Never ending, after completion it will start again automatically. E.g. colonist migration.
- **requiredLaunches:** Only for Multiple launchType. The number of launches that are required to complete the project.
- **criterions:** Criteria met to be able to start and/or progress the project.
- **expirationCriterions:** Criteria met for the project to become expired and no longer available.
- **launchActions:** TextActions to be executed on each launch.
- **completeActions:** TextActions to be executed when the project is completed.
- **description:** Description of the project, or description localization key.



- **requiredBuilding:** If assigned, this project can only be launched from the required building type. This building type should have the hasSpacePort flag enabled.
- **position:** 3D position in world space of the project proxy on space. Base projects can't have this overridden.
- **iconName:** Icon used by the project in space, and HUD. Base projects can't have this overridden.
- **resetOnComplete:** If this is enabled, it would reset the project when completed, allowing to have a project with multiple launches that can be completed infinite times.
- **specialProjectController:** Controller for this special project. It mainly defines project stages.
 - **spacePortStages:** List of stages of the project.
 - **key:** Stage key. Used for specific internal behaviors, and localization purposes.
 - **iconName:** Icon used for the stage in HUD.
 - **cancellable:** If enabled, the project can be canceled by the user when on this stage.
 - **monthsDuration:** Duration of this stage in months.
 - **specialCriteria:** Special criteria to count a stage as completed. Currently only supports ResourceMet.
 - **onEnter:** TextActionsto is executed when entering this stage.
 - **focusPosition:** If defined, captures the camera when entering this stage, and moves it to the configured position. Supported: Proxy, Port, LandingSite.
 - **focusCondition:** If focusPosition is set, this indicates the condition to capture the camera. Supported. FirstTime, LastLaunch.
 - **focusCameraZoom:** If defined it overrides the camera zoom when capturing the camera with focusPosition.
 - **focusCameraTiltX:** If defined it overrides the camera tilt on X when capturing the camera with focusPosition.
 - **focusCameraTiltY:** If defined it overrides the camera tilt on Y when capturing the camera with focusPosition.
 - **focusOnExitStage:** If true it triggers the camera focus when exiting the stage instead of when entering.





Annex IV - Languages

Languages

At ISA we believe that the science that we develop in the Oxy-UN states is a gift to all humanity. That is why it is important to make it accessible to other communities. And you can do your bit by adding new languages to the game.

First, add the reference to your language YAML on the manifest, as with other type lists.

```
modId: "Jeringozo Language Pack"
languages:
  filenames:
    - language.yaml
  replace: false
```

In your language YAML, you'll have to define these things:

- **name:** Name of the language, or localization key for the name.
- **systemLanguage:** System language, used by dialogues.
- **files:** List of CVS files used for the localization.

```
jer:
  name: JERINGOZO
  systemLanguage: English
  files:
    - jeringozo_EN.csv
```

Then, your CSV files should have these columns:

- Context: Text usage context, used for internal organization. Use Frontend.
- ID: Localization key. This is the key you can use in your mods. E.g. when creating a new building, put this key in the "name" field in the buildings YAML.
- Text: The text localized in the language you want to add.
- Comments: Optional comments



```
jeringozo_EN.csv ×
1 Context,ID,Text,Comments
2 Frontend,TXT_MainMenu_Buttons_New_campaign,Campapapañapa,
3 Frontend,TXT_MainMenu_Buttons_Continue,Conpotipinupuarpa,
4 Frontend,TXT_MainMenu_Buttons_Load,Capargapar,
5 Frontend,TXT_MainMenu_Buttons_Options,Apajupustepes,
```

Once your mod is enabled, the new language will be selectable in the options menu.

Extending included Languages

You can also add localized text for the included languages, to locate new texts for your mods.

To do this, you must create a YAML language that uses the same key as one of the included languages.

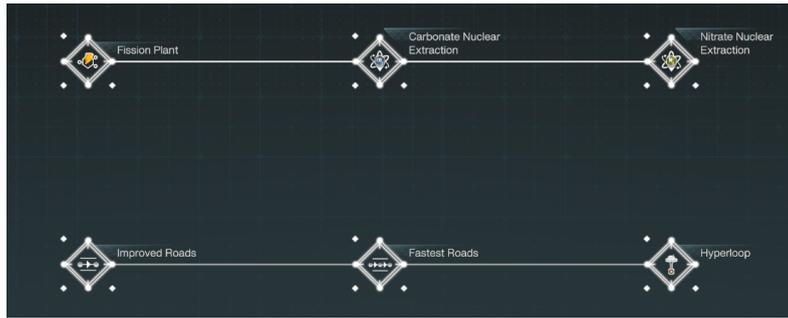
Here is a list of all the preexisting languages and their keys, their names, and systemLanguages:

```
e( key: "base" , name: "TXT_MainMenu_Options_English" , SystemLanguage.English);
e( key: "FR" , name: "TXT_MainMenu_Options_French" , SystemLanguage.French);
e( key: "DE" , name: "TXT_MainMenu_Options_German" , SystemLanguage.German);
e( key: "ES" , name: "TXT_MainMenu_Options_Spanish" , SystemLanguage.Spanish);
e( key: "JA" , name: "TXT_MainMenu_Options_Japanese" , SystemLanguage.Japanese);
e( key: "PT" , name: "TXT_MainMenu_Options_Portuguese" , SystemLanguage.Portuguese);
e( key: "RU" , name: "TXT_MainMenu_Options_Russian" , SystemLanguage.Russian);
e( key: "ZH-Hans" , name: "TXT_MainMenu_Options_Chinese" , SystemLanguage.ChineseSimplified);
```

Annex V - Tech Tree

Lanes

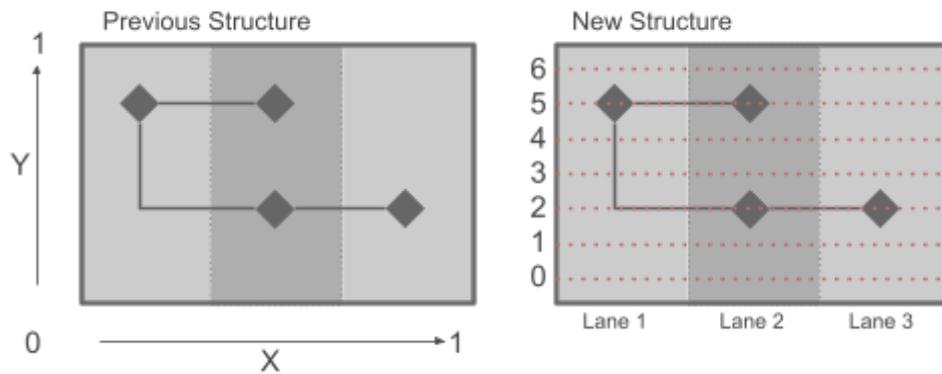
Many of the technologies that you can develop are based on the development of less advanced technology. This dependency between technologies defines what we call lanes, which in turn defines the location of that technology in the Tech Tree.



New Lane Logic

We recently redesigned the Tech Tree, so the lanes now have a new logic. To position a new technology on the Tech Tree, you no longer need to define a value between 0 and 1 for the X and Y axes, now you only define a value between 0 and 6 for the vertical axis.

The horizontal axis is automatically defined according to the level of technology required.



This is how it should look like in the YAML:

Previous YAMLS

New YAMLS





```
technology_colony_small:  
  actions:  
    - arguments:  
      - PlayerFaction  
      - building_colony_small  
      command: UnlockBuilding  
      daysDelay: 0.0  
  isLane: false  
  name: BE_technology_colony_small_name  
  description: BE_technology_colony_small_description  
  iconName: Tectree Icons/ICO_TT_Colony2  
  popupImageName: Popups/Images/IMG_PU_Biology_ColonySmall  
  requiredResearchPoints: 150.0  
  requirements:  
    - !technology technology_lane_life_support_0  
  connections:  
    technology_colony_medium: []  
  isPlaceholder: false  
  position:  
    x: 0.0903165489  
    y: 0.850000024
```

```
technology_colony_small:  
  actions:  
    - arguments:  
      - PlayerFaction  
      - building_colony_small  
      command: UnlockBuilding  
      daysDelay: 0.0  
  isLane: false  
  name: BE_technology_colony_small_name  
  description: BE_technology_colony_small_description  
  iconName: Tectree Icons/ICO_TT_Colony2  
  popupImageName: Popups/Images/IMG_PU_Biology_ColonySmall  
  requiredResearchPoints: 150.0  
  requirements:  
    - !technology technology_lane_life_support_0  
  connections:  
    technology_colony_medium: []  
  isPlaceholder: false  
  position: 6
```

Warning: Previous mods that include new technologies will require adjusting to the new structure for them to work.

