

Maestro controller use cases

User wants to add a new compute resource type

- 1) User clicks on "Add new compute resource type" under label "core types"
- 2) User inputs the name of the resource type (eg "class1-hw-vm") and the number of total compute units
- 3) User presses add/save
 - a. If the name already exists, the user is prompted to update their entry with a different name
 - b. If the name is unique, the resource type is added to ComputeResourceTypes

User wants to add a new product type

- 1) User clicks on "Add new product type" under label "core types"
- 2) User inputs the name of the product type (eg "scenic", "standard", "ocean") and the number of compute units it will use
- 3) User presses add/save
 - a. If the name already exists, the user is prompted to update their entry with a different name
 - b. If the name is unique, the product type is added to ProductTypes

User wants to add a new compute container

- 1) User clicks on "Add a new compute container" under label "compute"
- 2) User inputs the hostname of the container eg "simulation01.inworldz.com"
- 3) User selects a resource type by its name from a drop down list eg "class1-hw-vm"
- 4) User inputs the number of resources provided by this container
- 5) User presses add/save
 - a. If the name already exists, the user is prompted to update their entry with a different name
 - b. If the name is unique, the resource type is added to ComputeContainers. Additionally, X compute resources are added to the ComputeResources table and their type and container_id set to this container. Their state is initially set to zero which means unprovisioned. All other properties are set to null.

User wants to edit a compute container

- 1) User clicks on "Edit container" from a list of compute containers under label "compute"
- 2) An interface is shown containing:
 - a. The name of the compute container (editable)
 - b. A delete button to completely remove the container and all of its compute resources
 - c. A list of all the compute resources provided by this compute container with editable properties for:
 - i. Hostname
 - ii. Internal IP
 - iii. External IP
 - iv. State (dropdown)

User wants to edit a compute container (resource)

- 1) User clicks on "Edit container" from a list of compute containers under label "compute"
- 2) The previously described interface is shown
- 3) User edits the information for the compute resource and presses save
- 4) Container changes and all resource changes made on the interface are saved

User wants to delete a compute container

- 1) User clicks on "Edit container" from a list of compute containers under label "compute"
- 2) The previously described interface is shown
- 3) User presses the "delete container" button"
 - a. If there are any deployed regions associated with any of this container's compute resources an error is shown stating "a compute resource cannot be deleted without first decommissioning all regions utilizing the resource". The action is aborted
 - b. Else: A confirmation message is shown "This will remove this container and all of its resources will no longer be available for provisioning through maestro. Are you sure?"
 - i. If the user presses yes, all compute resources associated with this container are deleted followed by the container itself.
 - ii. If the user presses no, no changes are made.