# Inventories

I

# What are Inventories?

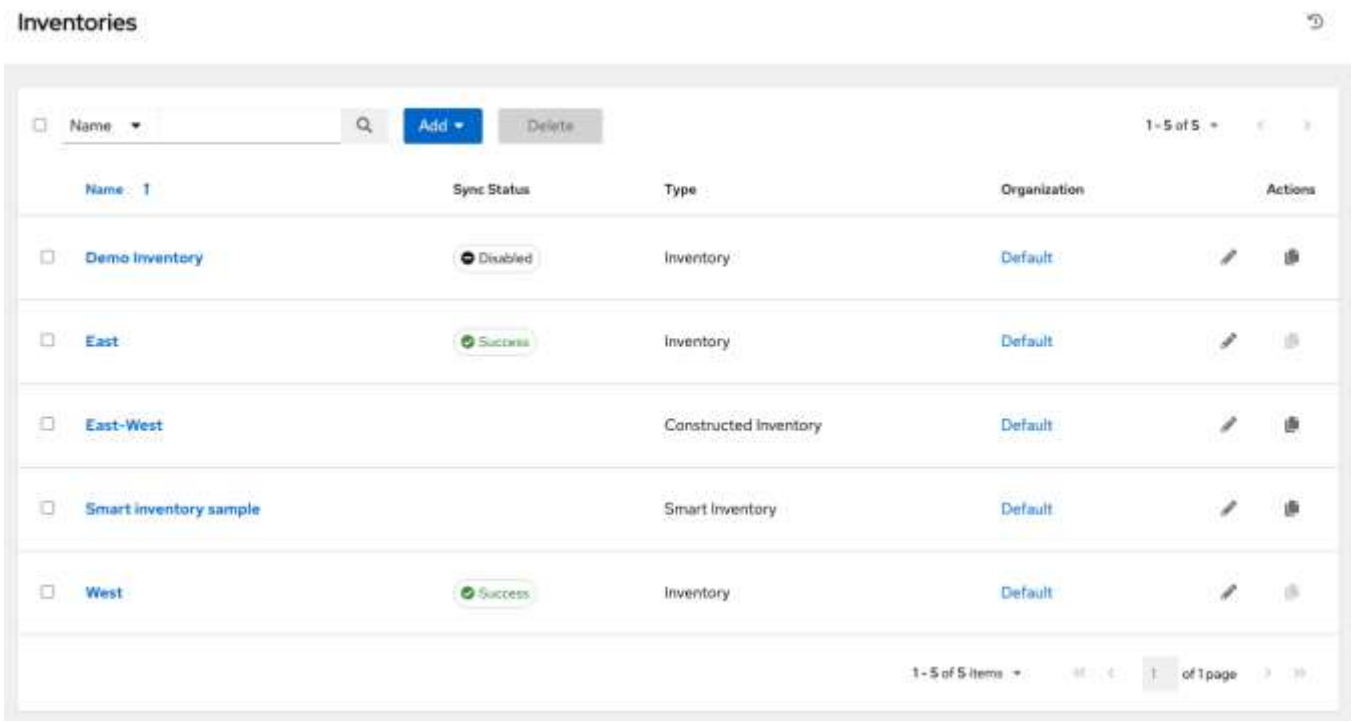


- An Inventory is a collection of hosts against which jobs may be launched, the same as an Ansible inventory file.
- Inventories are divided into groups and these groups contain the actual hosts. Groups may be sourced manually, by entering host names into AWX

# Inventories window.



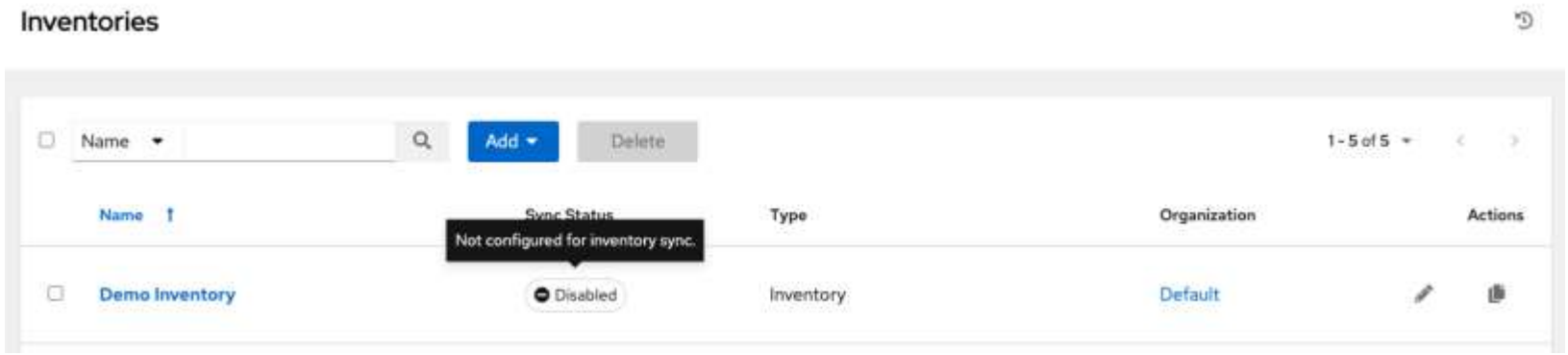
- The tool displays a list of available inventories, which can be sorted by name, searched type, organization, description, owners, modifiers, or additional criteria as needed.



# Inventories window..



- An example of inventories of various states, including one with detail for a disabled state:




- **Type-** Identifies whether it is a standard inventory, a Smart inventory, or a constructed inventory.
- **Organization**: The organization to which the inventory belongs.
- **Actions**

# Smart Inventories



- A Smart Inventory is a collection of hosts defined by a stored search that can be viewed like a standard inventory and made to be easily used with job runs.
- Organization administrators have admin permission to inventories in their organization and can create a Smart Inventories.
- A Smart Inventory is identified by **KIND=smart**
- **InventorySource** is directly associated with an Inventory.
- The Inventory model has the following new fields that are blank by default but are set accordingly for Smart Inventories:
  - **kind** is set to **smart** for Smart Inventories
  - **host_filter** is set AND kind is set to smart for Smart Inventories.

# Smart Host Filter.

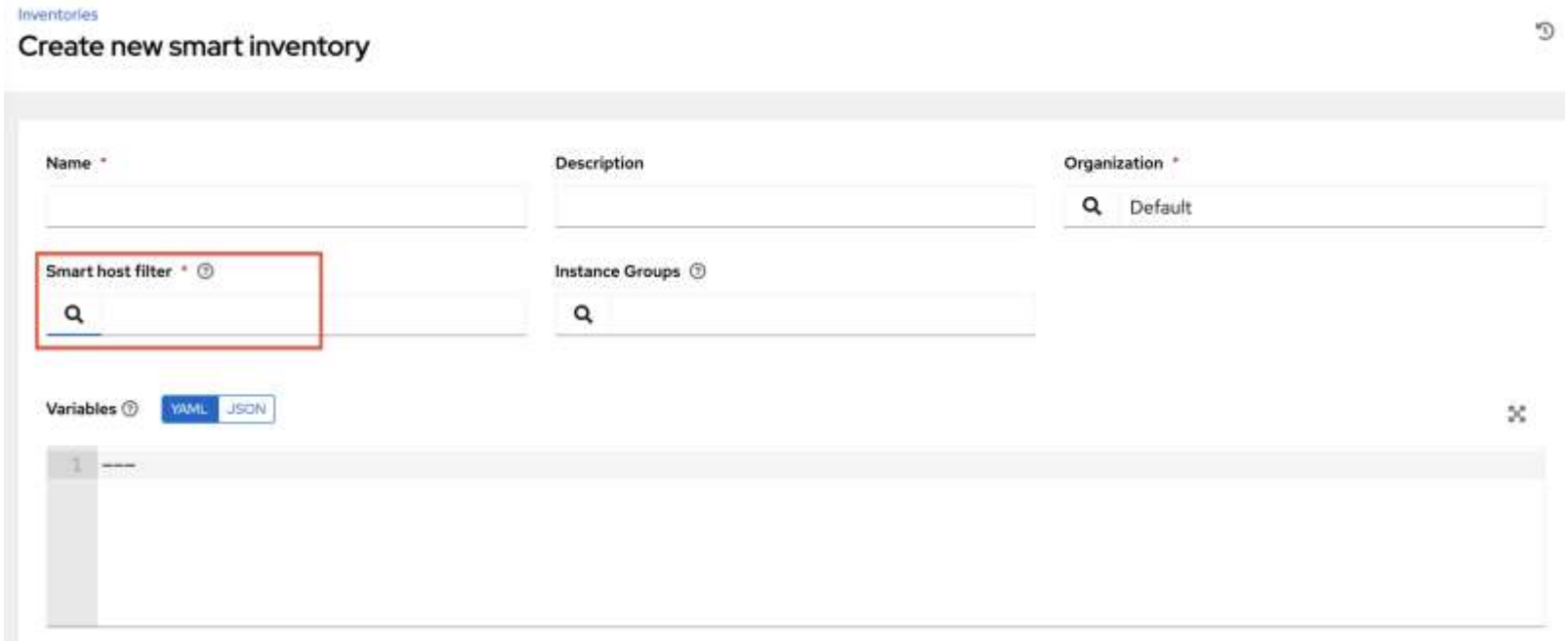


- Facts generated by an Ansible playbook during a Job Template run are stored by AWX| into the database whenever use_fact_cache=True is set per-Job Template.
- The **host_filter** parameter allows for:
  - grouping via ()
  - use of the boolean and operator:
    - __ to reference related fields in relational fields
    - __ is used on ansible_facts to separate keys in a JSON key path
    - **[]** is used to denote a json array in the path specification
    - "" can be used in the value when spaces are wanted in the value
  - **“classic”** Django queries may be embedded in the **host_filter**

# Define host filter with ansible_facts.



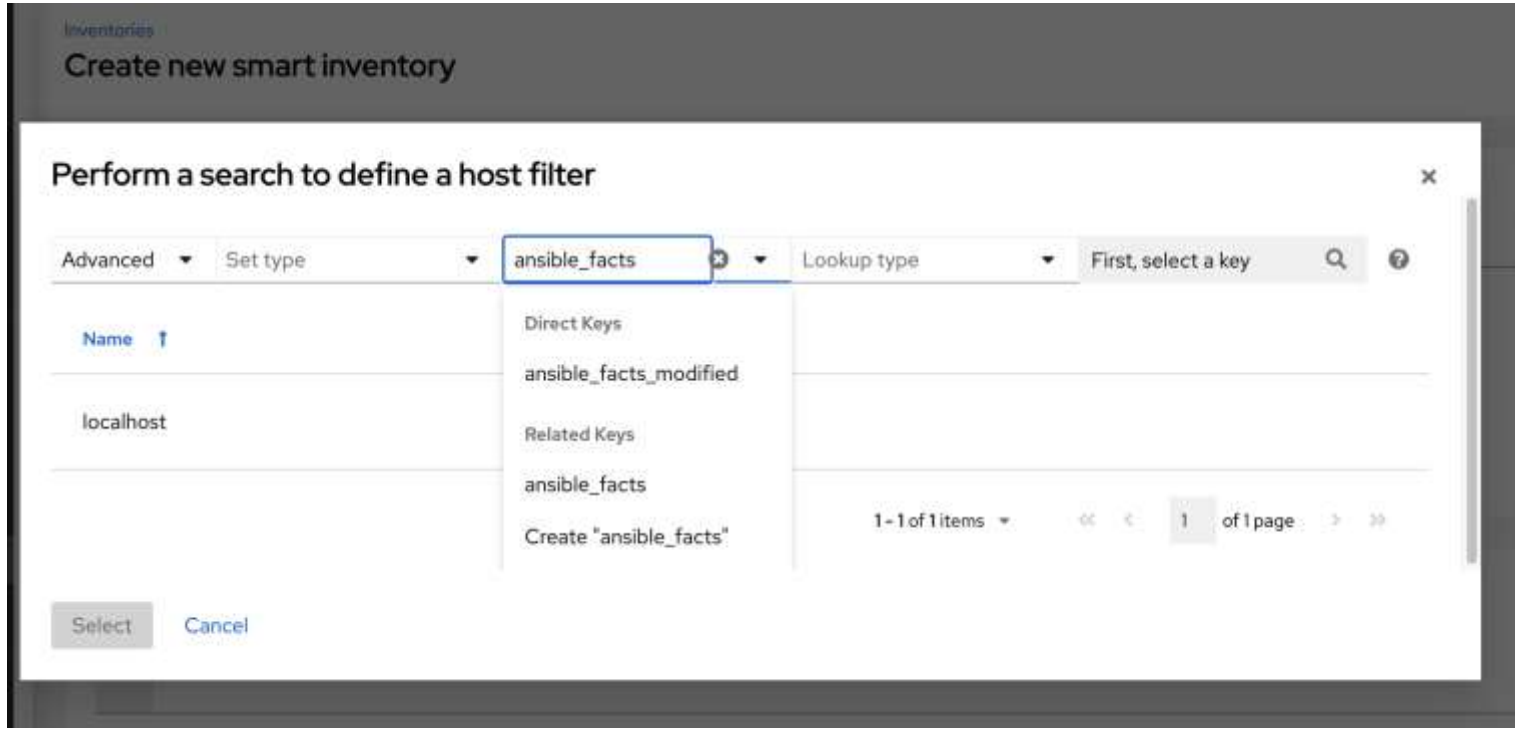
- In the **Create new smart inventory** screen, click the  button next to the **Smart host filter** field to open a pop-up window to filter hosts for this inventory.



# Define host filter with ansible_facts..



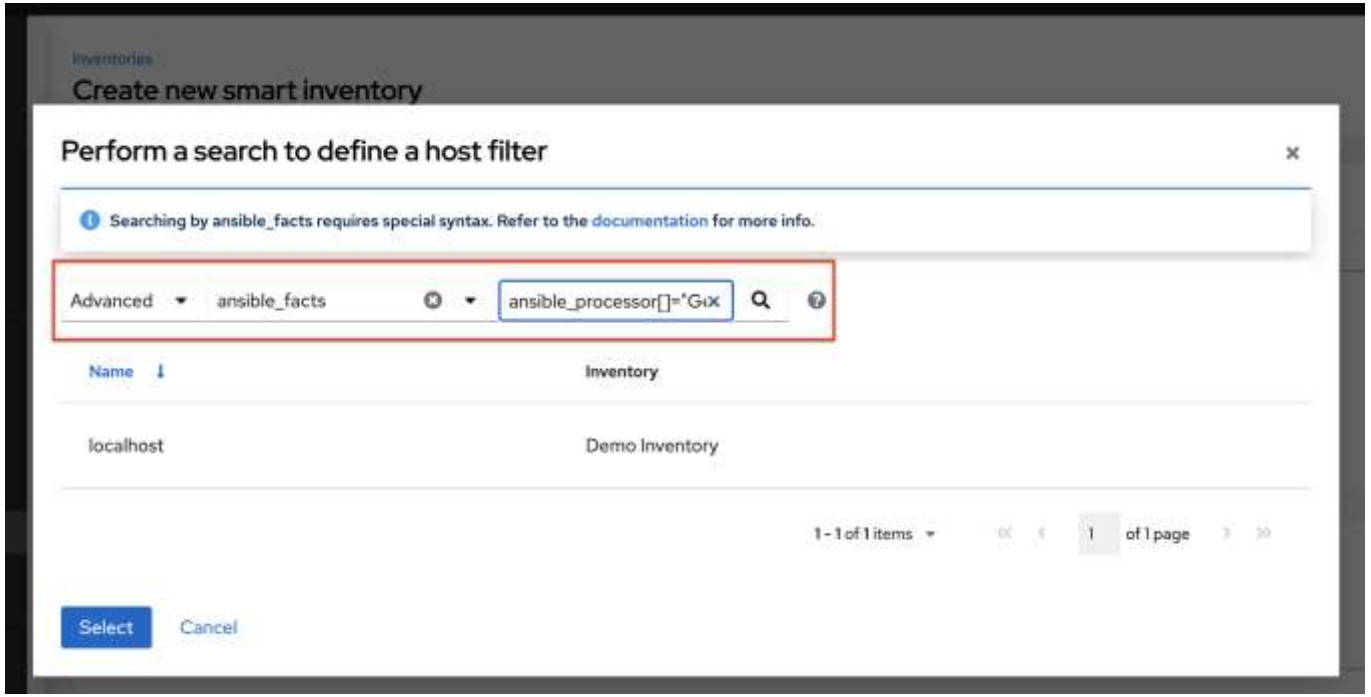
- In the search pop-up window, change the search criteria from **Name** to **Advanced** and select **ansible_facts** from the **Key** field.



# Define host filter with ansible_facts…



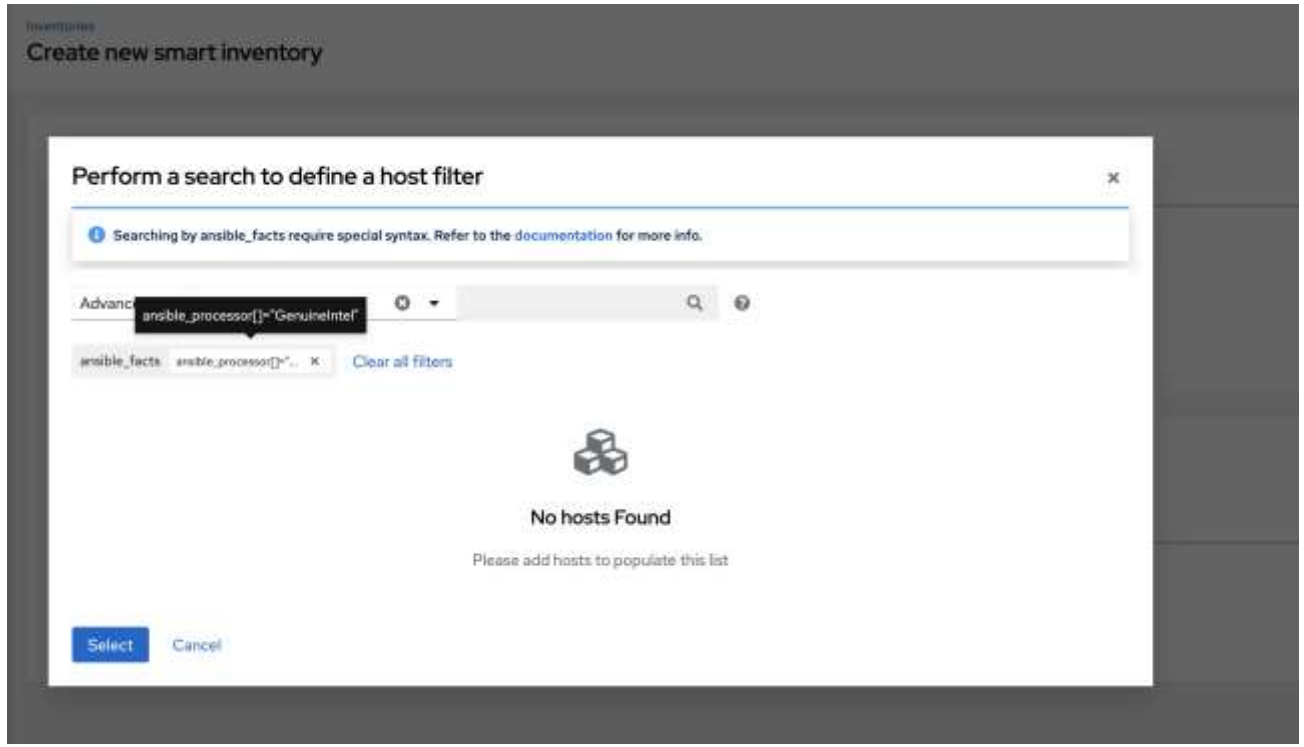
- In the search field, enter ansible_processor[]="GenuineIntel" (no extra spaces or __ before the value) and press **[Enter]**.



# Define host filter with ansible_facts….



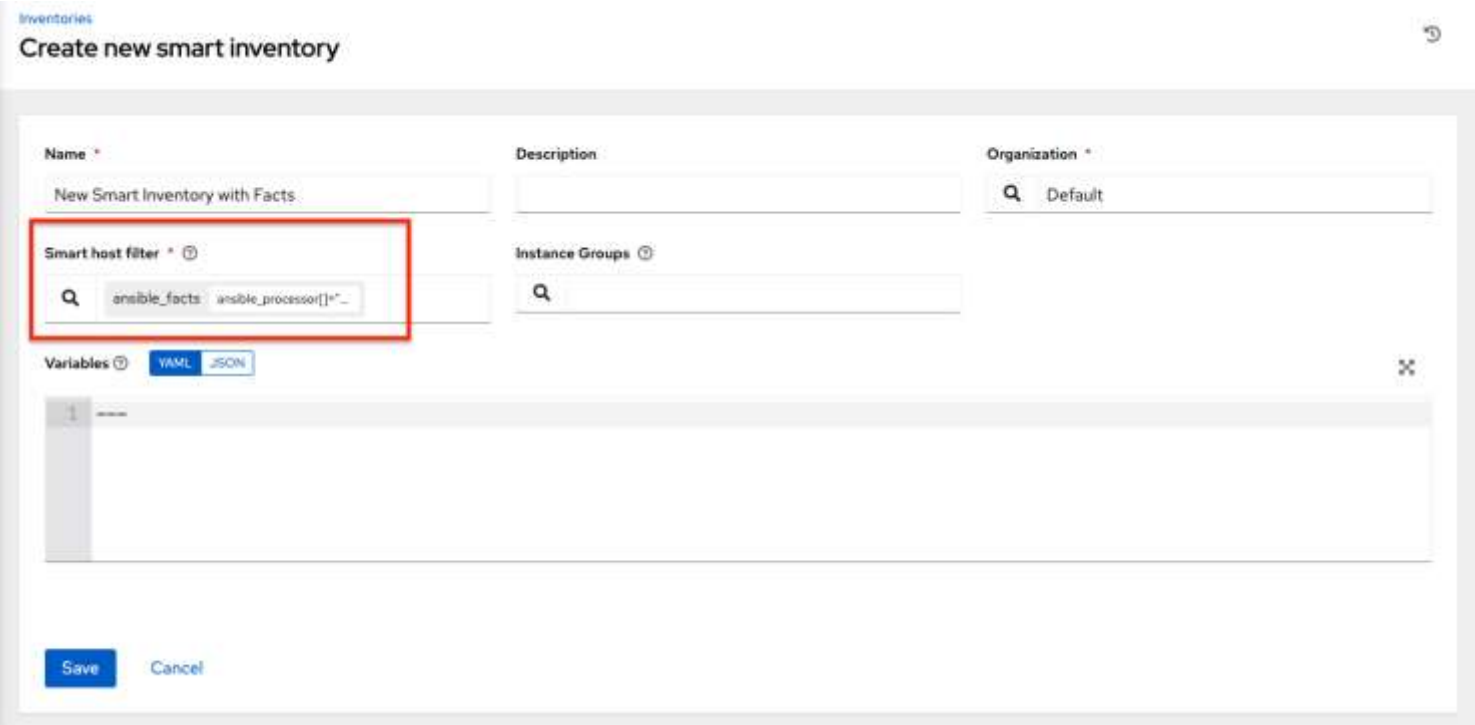
- The resulting search criteria for the specified ansible fact populates in the lower part of the window.



# Define host filter with ansible_facts…



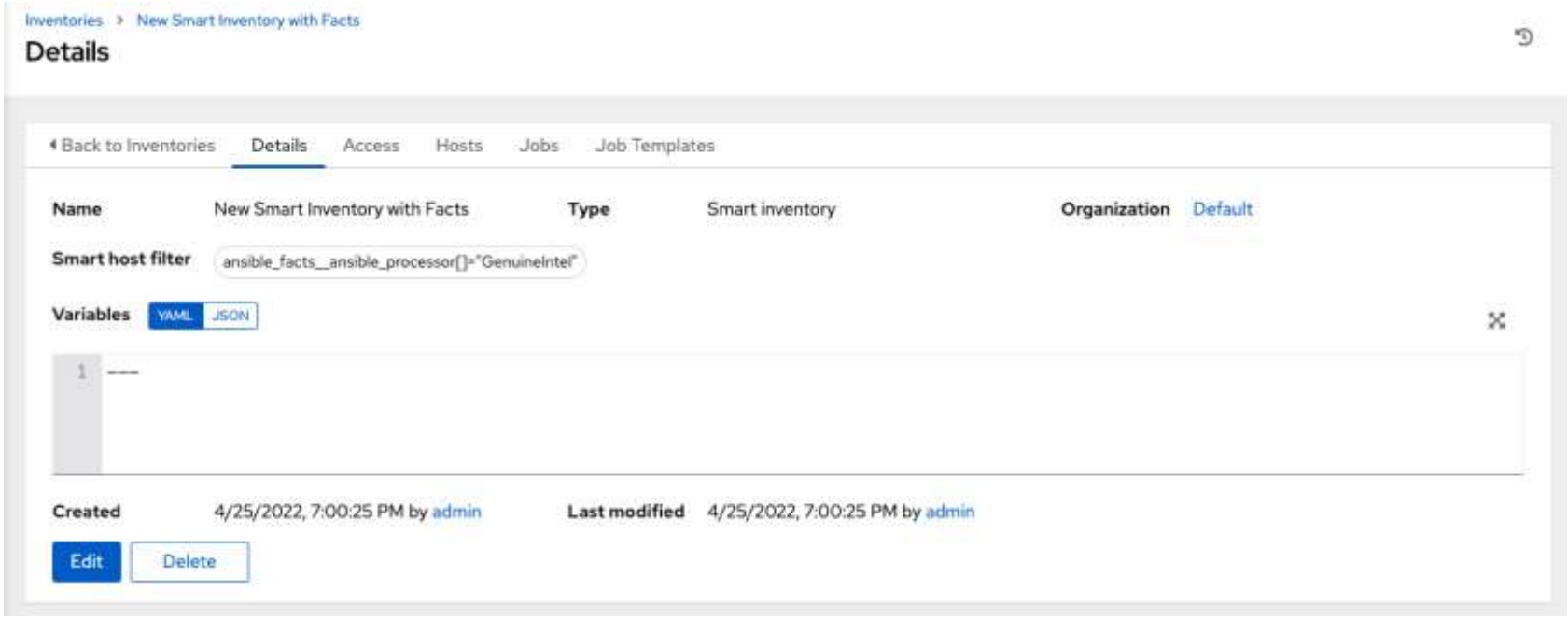
- Click **Select** to add it to the **Smart host filter** field.



# Define host filter with ansible_facts…



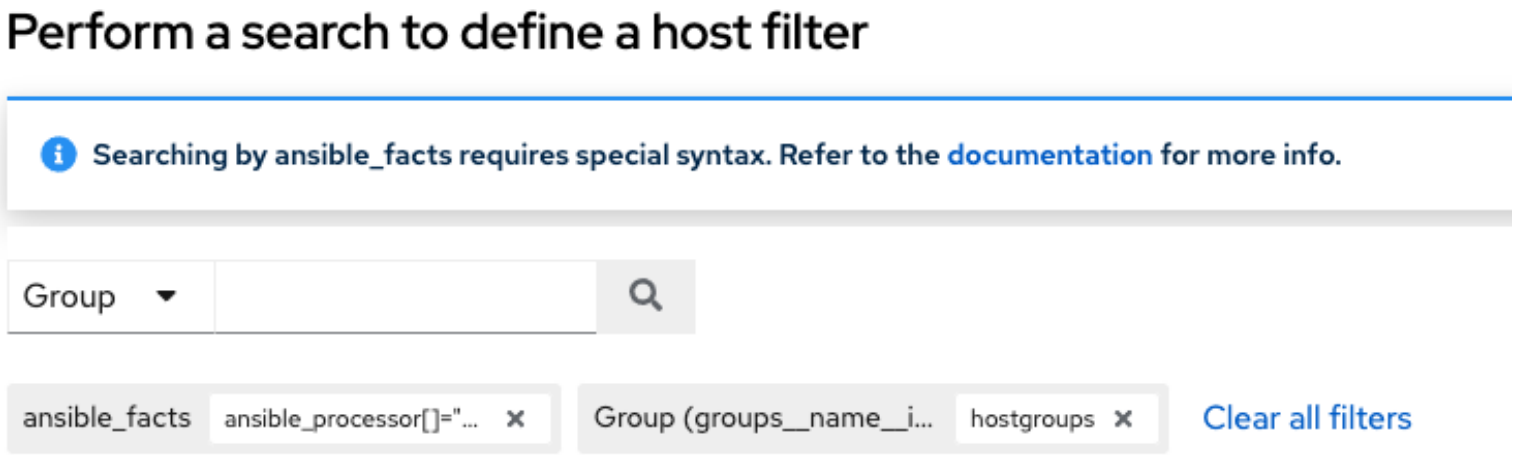
- Click **Save** to save the new Smart Inventory.
- The Details tab of the new Smart Inventory opens and displays the specified ansible facts in the **Smart host filter** field.



# Define host filter with ansible_facts…



- From the Details view, you can edit the **Smart host filter** field by clicking **Edit** and delete existing filter(s), clear all existing filters, or add new ones.



# Constructed Inventories



- As a platform user, this feature allows creation of a new inventory (called a constructed inventory) from a list of input inventories.
- The constructed inventory contains copies of hosts and groups in its input inventories, allowing jobs to target groups of servers across multiple inventories.
- Groups and **hostvars** can be added to the inventory content, and hosts can be filtered to limit the size of the constructed inventory.
- The key factors that distinguish a constructed inventory from a Smart Inventory are:
  - the normal Ansible hostvars namespace is available
  - they provide groups

# Inventory Plugins



- Inventory updates use dynamically-generated YAML files which are parsed by their respective inventory plugin.
- Users can provide the new style inventory plugin config directly to AWX via the inventory source **source_vars** for all the following inventory sources:
  - Amazon Web Services EC2
  - Google Compute Engine
  - Microsoft Azure Resource Manager
  - VMware vCenter
  - Red Hat Satellite 6
  - Red Hat Insights
  - OpenStack
  - Red Hat Virtualization
  - Red Hat Ansible Automation Platform

# Add a new inventory.

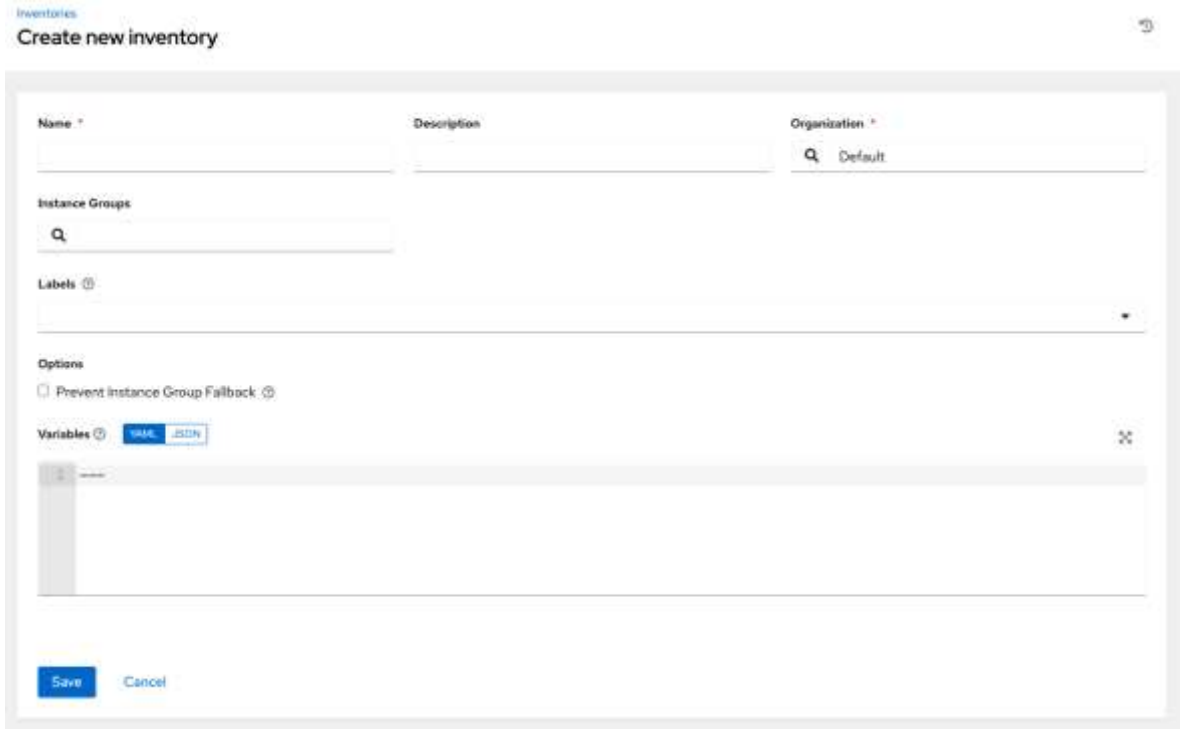


- Adding a new inventory involves several components:
- Add permissions
- Add groups
- Add hosts
- Add source
- View completed jobs

# Add a new inventory..



- To create a new standard inventory, Smart inventory, or constructed inventory:
- Click the **Add** button, and select the type of inventory to create.



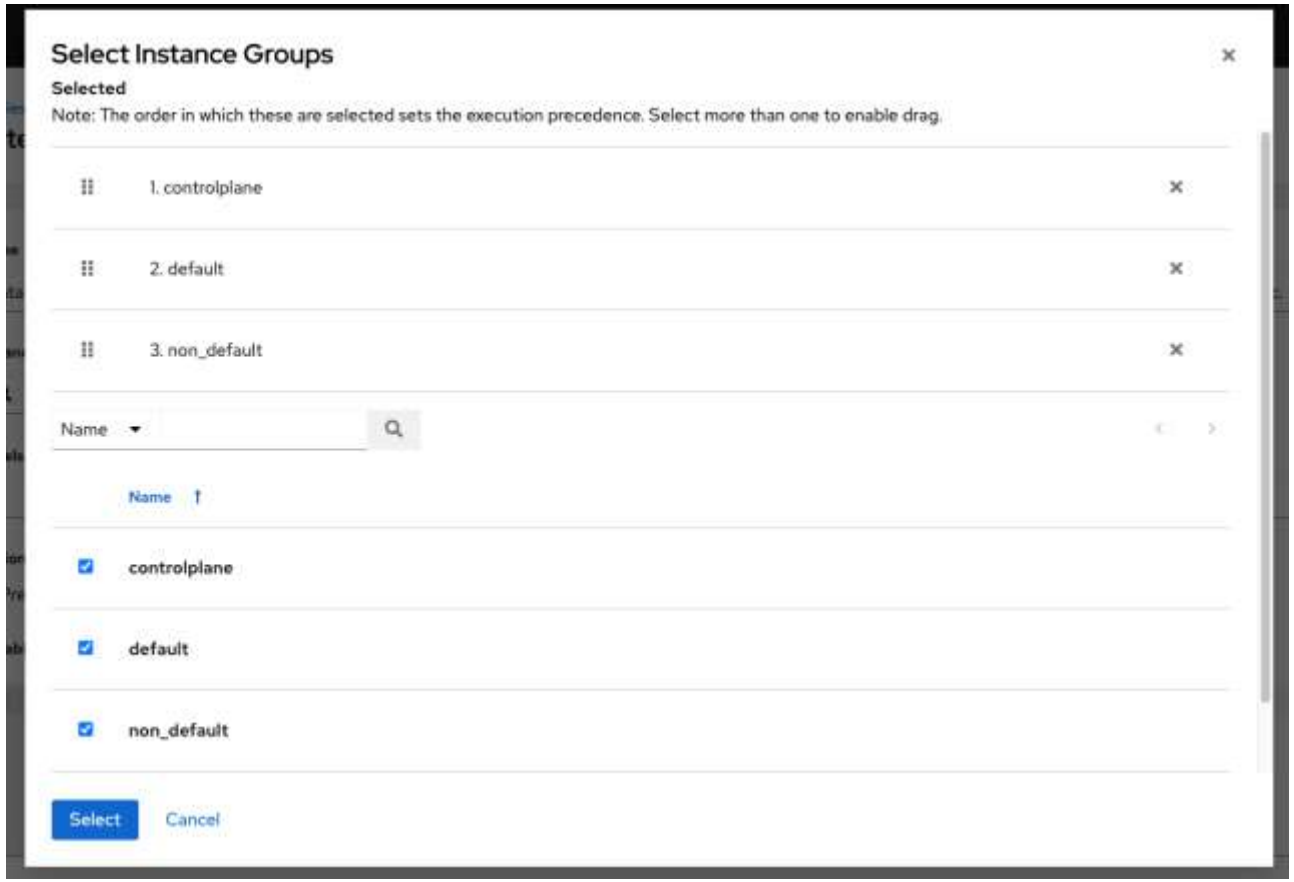
# Add a new inventory…



- Enter the appropriate details into the following fields:
  - **Name**
  - **Description**
  - **Organization**
  - **Smart Host Filter**
  - **Instance Groups**
  - **Labels**

# Add a new inventory…





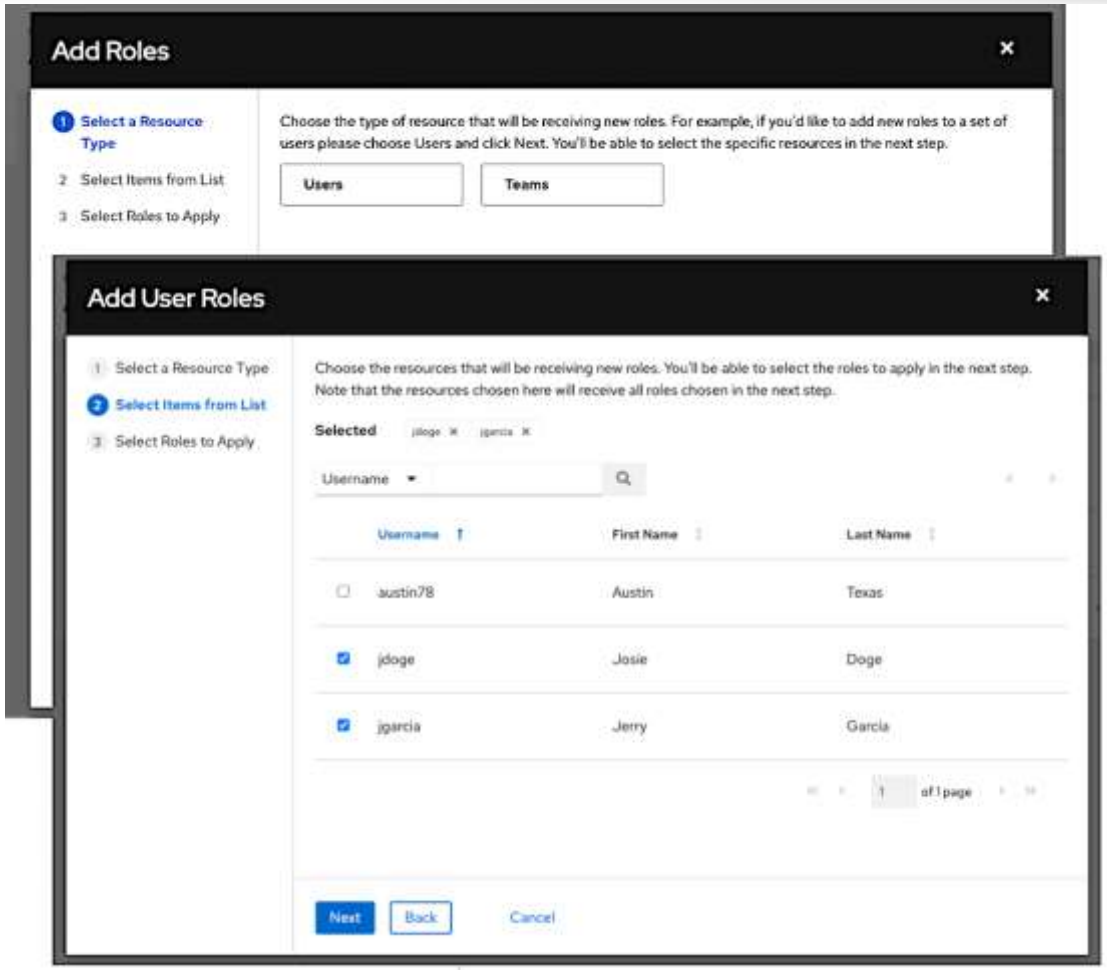
# Add permissions.



- In the **Access** tab, click the **Add** button.
- Select a user or team to add and click **Next**
- Select one or more users or teams from the list by clicking the check box(es) next to the name(s) to add them as members and click **Next**.

# Add permissions..

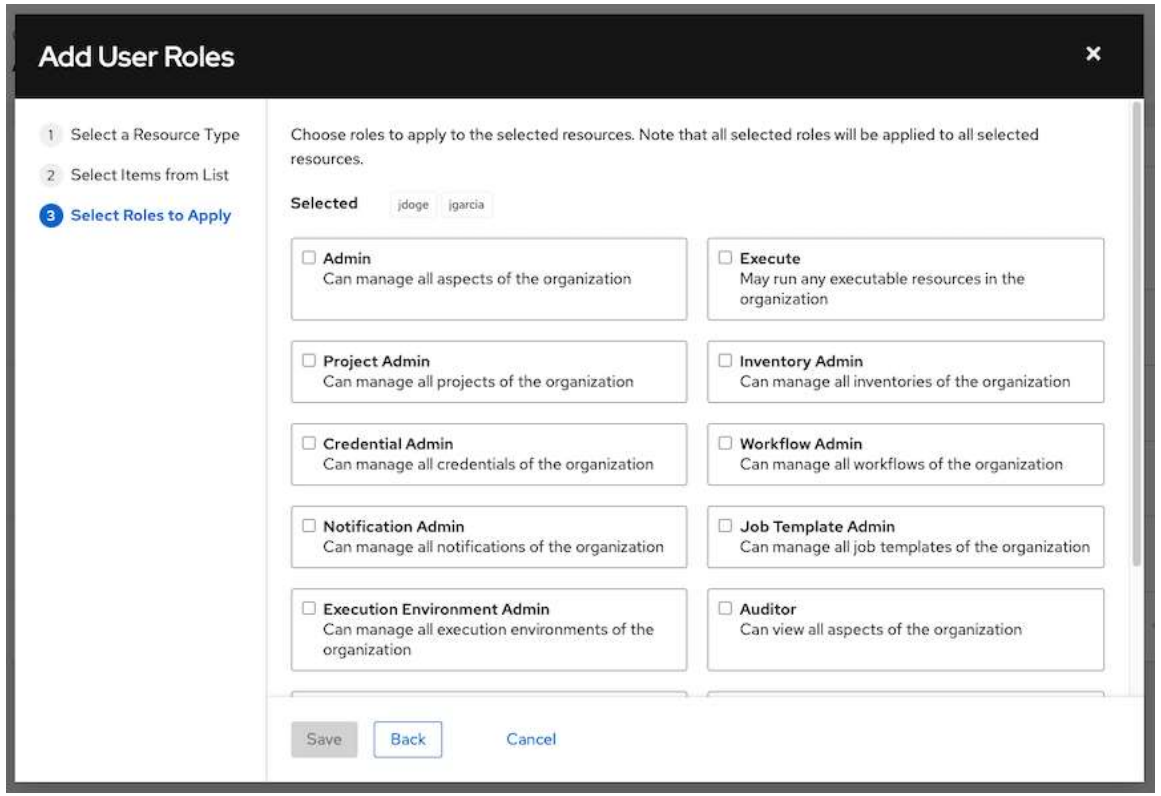




# Add permissions…



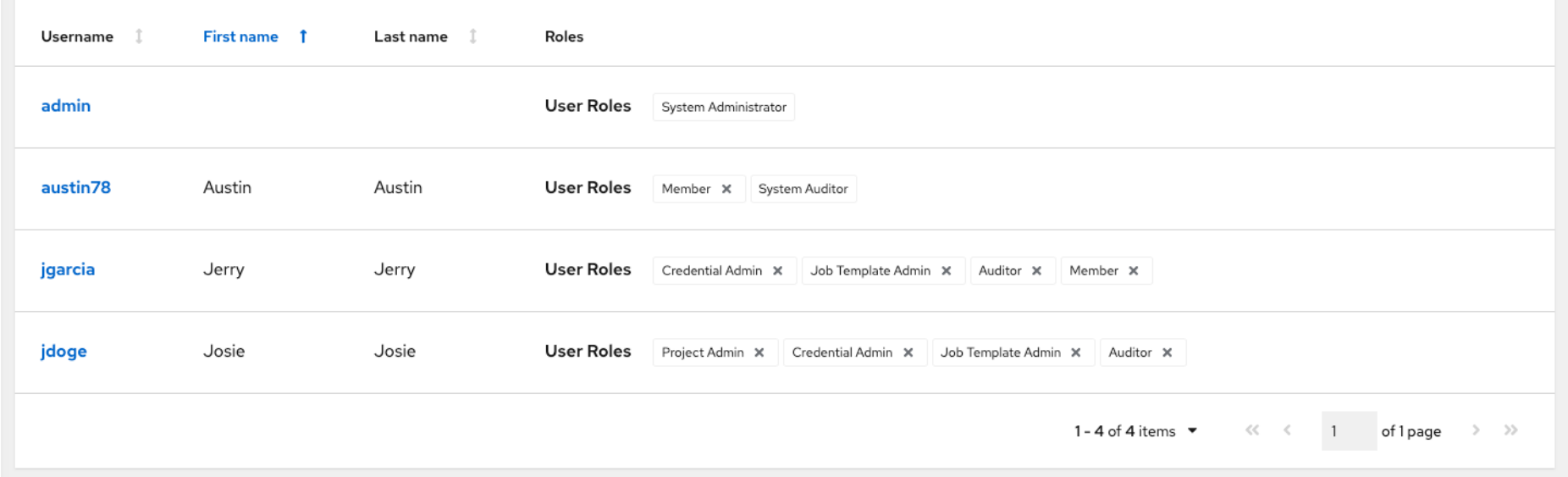
- Select the role(s) you want the selected user(s) or team(s) to have. Be sure to scroll down for a complete list of roles. Different resources have different options available.



# Add permissions…



- Click the **Save** button to apply the roles to the selected user(s) or team(s) and to add them as members.



# Add groups.

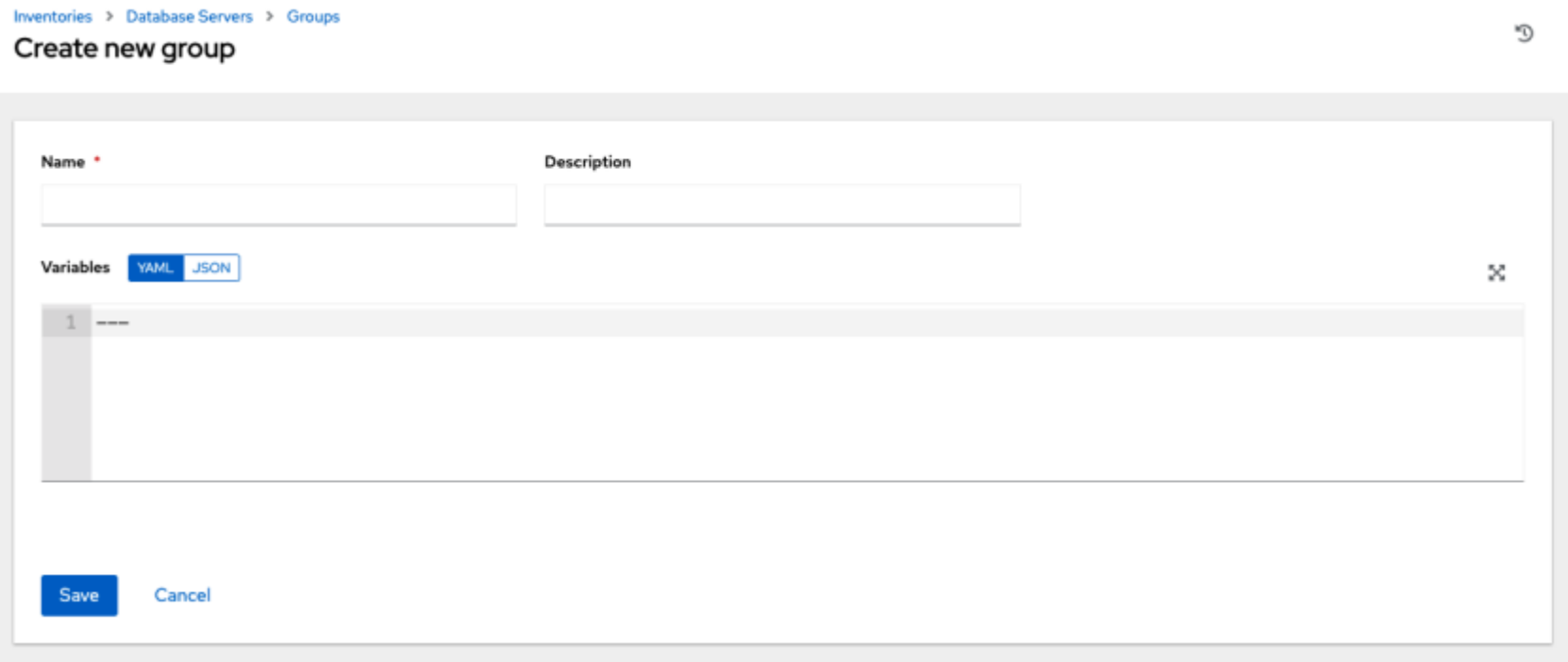


- Inventories are divided into groups, which may contain hosts and other groups, and hosts.
- Groups are only applicable to standard inventories and is not a configurable directly through a Smart Inventory.
  - Create a new Group
  - Create a new Host
  - Run a command on the selected Inventory
  - Edit Inventory properties
  - View activity streams for Groups and Hosts
  - Obtain help building your Inventory

# Add groups..



- Click the **Add** button to open the **Create Group** window.



# Add groups within groups.

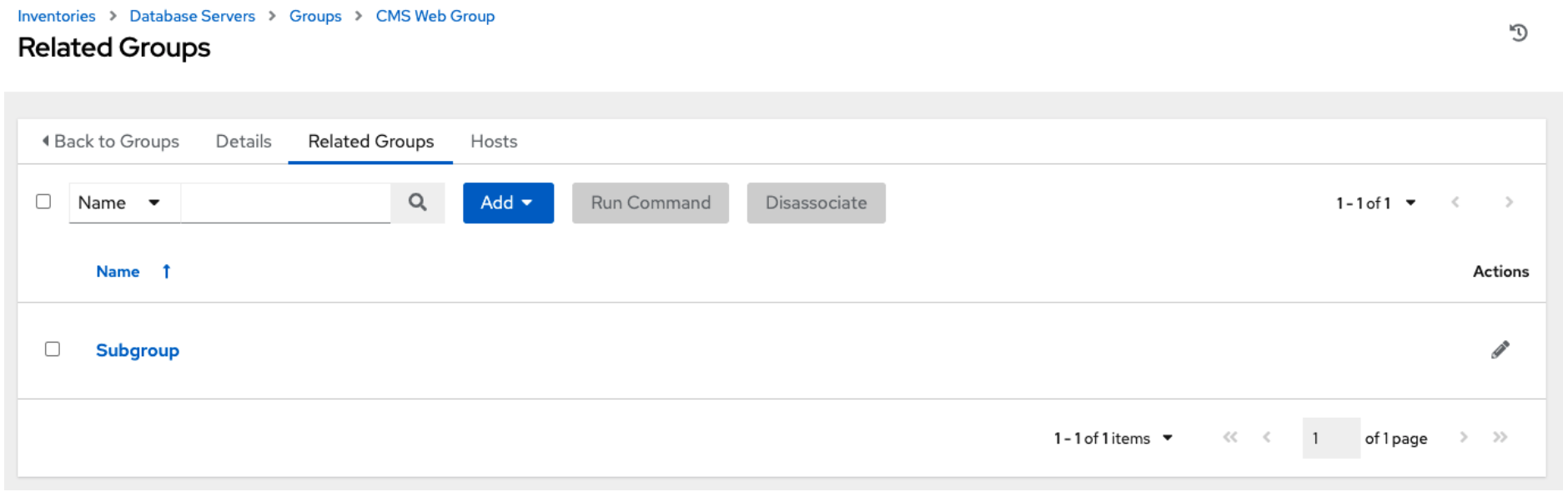


- Click the **Related Groups** tab.
- Click the **Add** button, and select whether to add a group that already exists in your configuration or create a new group.
- If creating a new group, enter the appropriate details into the required and optional fields:
  - **Name**
  - **Description**
  - **Variables**
- When done, click **Save**.

# Add groups within groups..



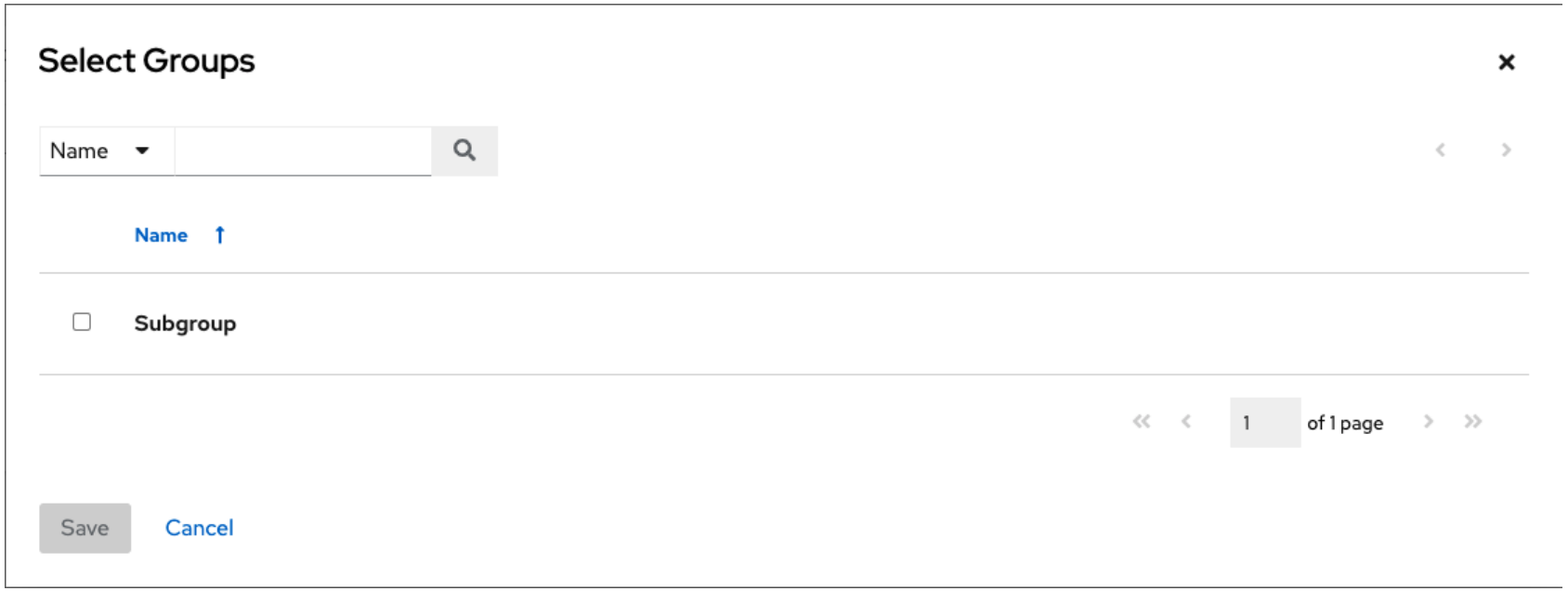
- The **Create Group** window closes and the newly created group displays as an entry in the list of groups associated with the group



# Add groups within groups…



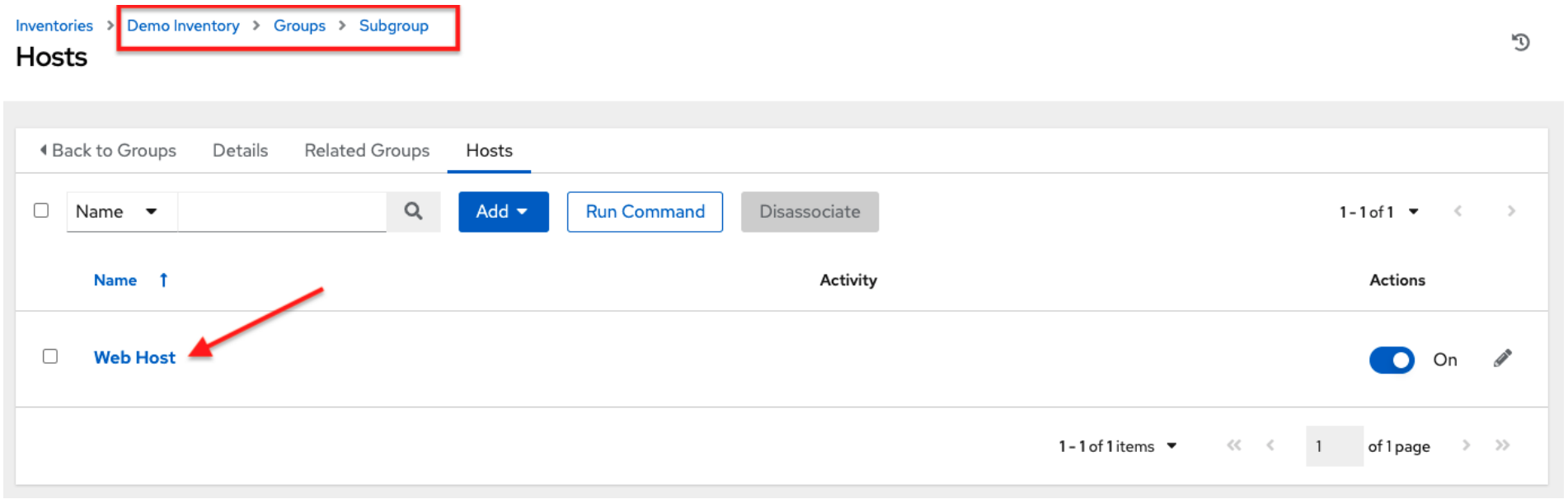
- If you chose to add an existing group, available groups will appear in a separate selection window.



# Add hosts.



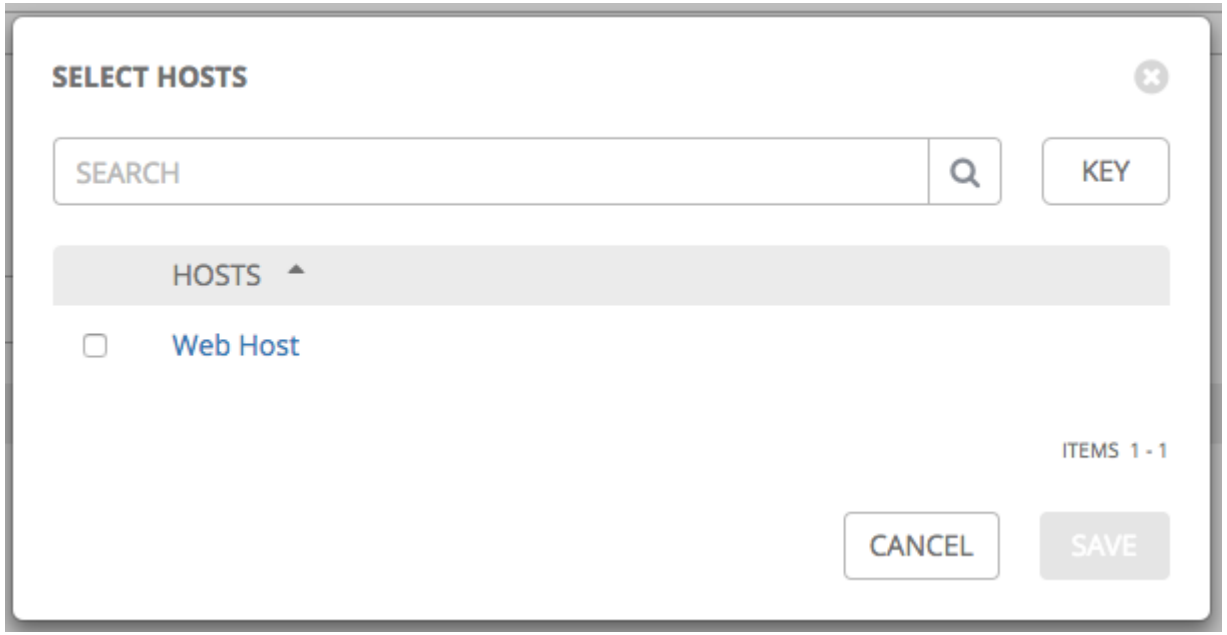
- The **Create Host** window closes and the newly created host displays as an entry in the list of hosts associated with the group



# Add hosts..



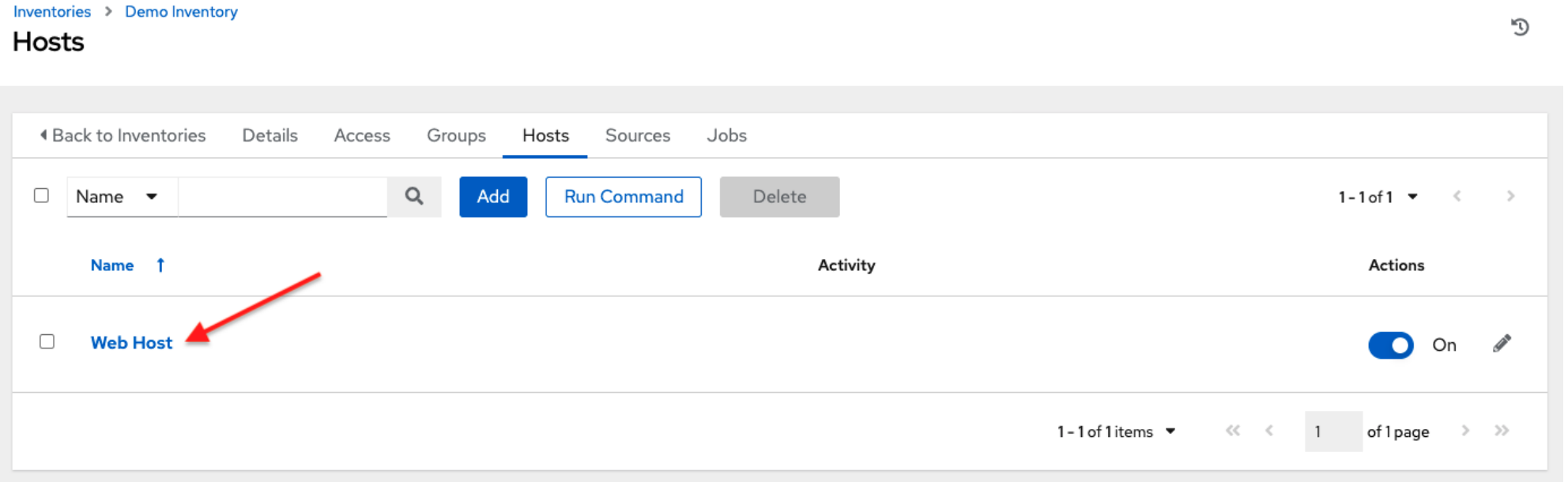
- If you chose to add an existing host, available hosts will appear in a separate selection window.



# Add hosts...



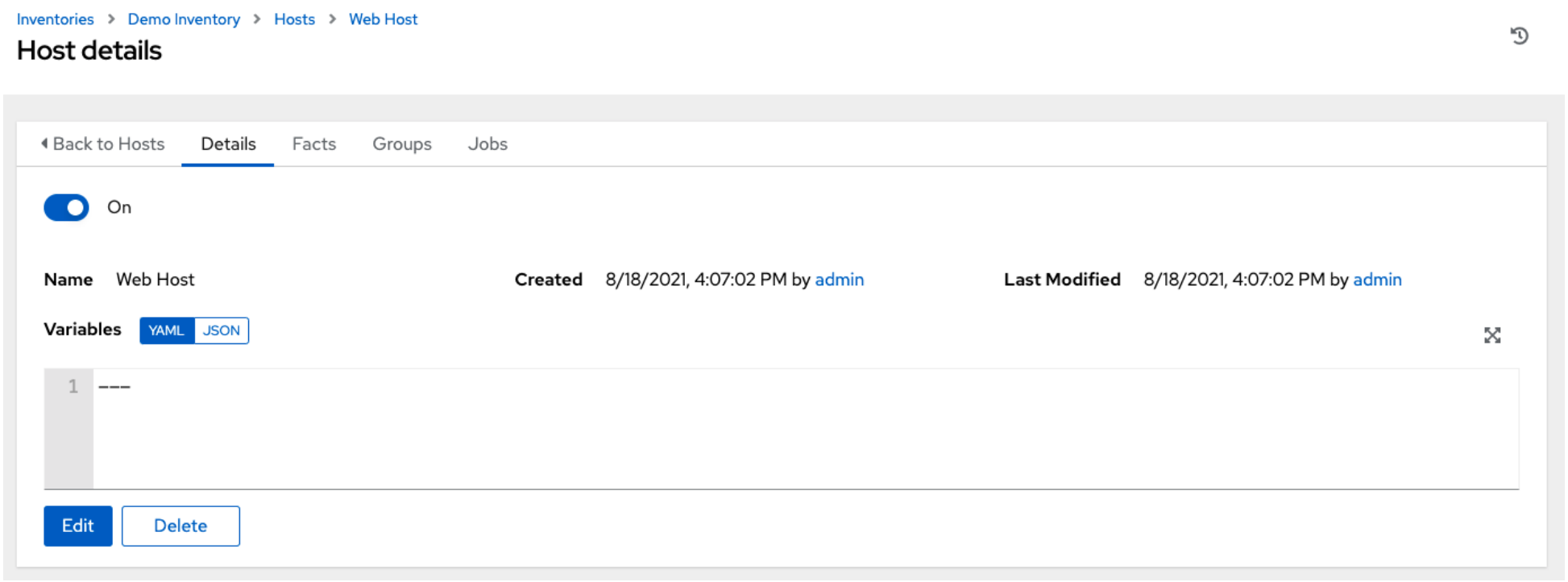
- To configure additional groups for the host, click on the name of the host from the list of hosts.



# Add hosts…



- This opens the Details tab of the selected host.



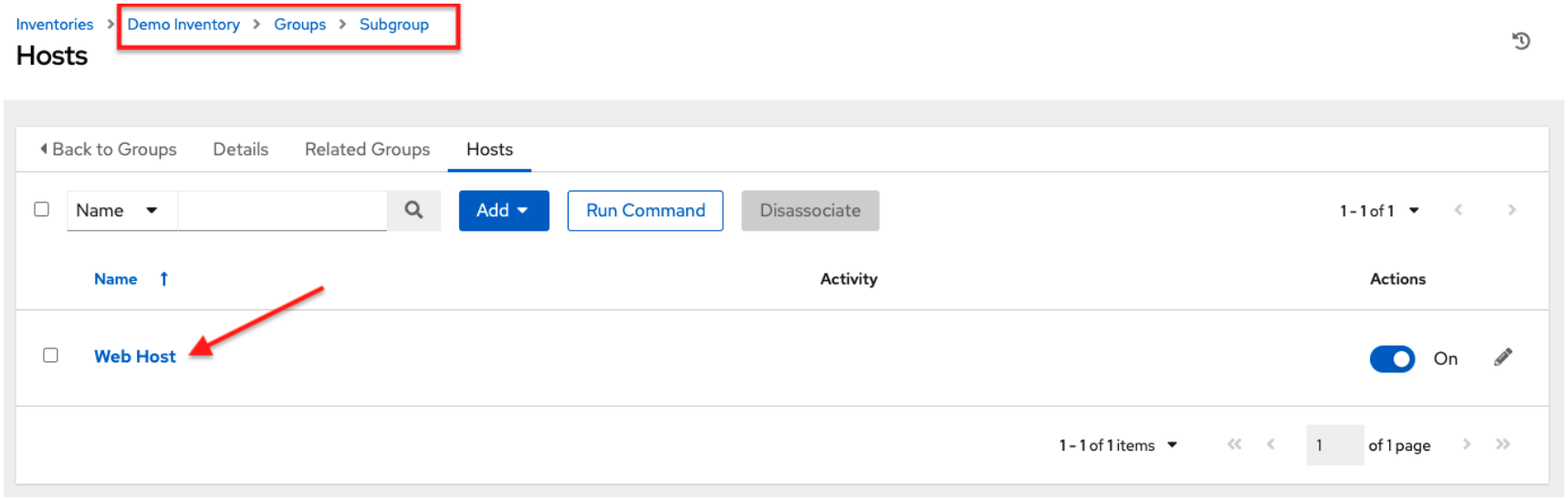
# Running Ad Hoc Commands

I

# Running Ad Hoc Commands.



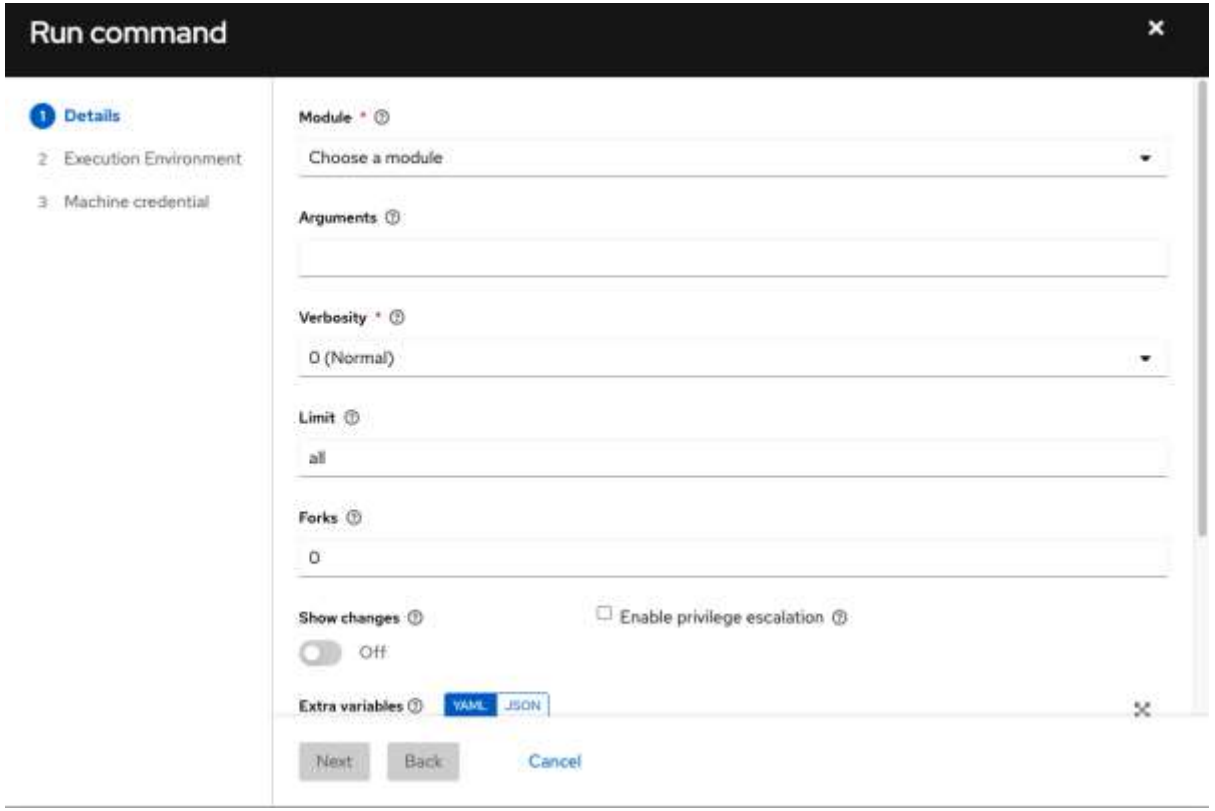
- Select an inventory source from the list of hosts or groups. The inventory source can be a single group or host, a selection of multiple hosts, or a selection of multiple groups.



# Running Ad Hoc Commands..



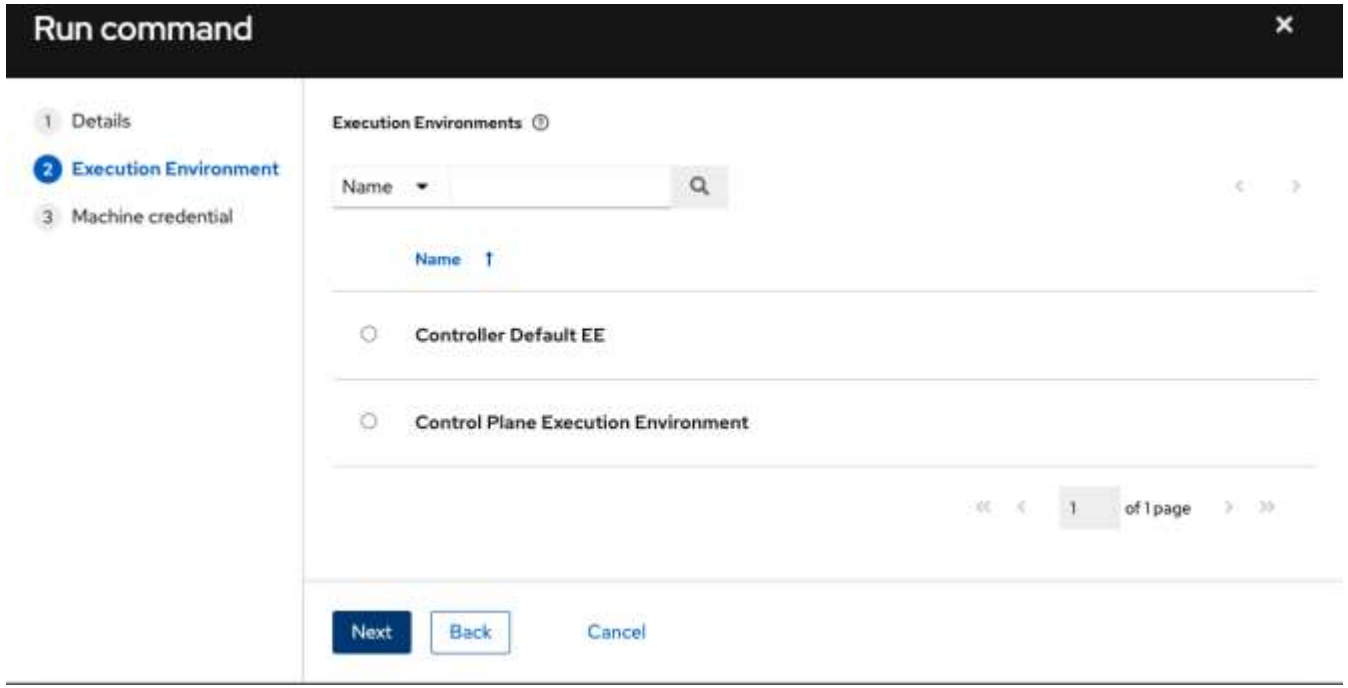
- Click the **Run Command** button.



# Running Ad Hoc Commands…



- Click **Next** to choose the execution environment you want the ad-hoc command to be run against.



# Running Ad Hoc Commands…



- Click **Next** to choose the credential you want to use and click the **Launch** button.