

Hadoop Configuration : Advanced Configuration Flow

- User specifies cluster name
- User selects Hadoop provider/plugin
- Controller invokes `get_versions()`
- User selects version and proceeds to subsequent configuration screen
- user selects a provider specific template
 - o controller reads file
- controller invokes `hadoop_plugin.populate_node_groups(cluster)`
 - o cluster object already has the following attributes populated:
 - name
 - plugin_name
 - plugin_info (formerly extra) – a reference to the plugin’s proprietary template
 - o `hadoop_plugin` detects the `plugin_info` attribute and parses `plugin_template`
 - o `hadoop_plugin` sets node groups on the cluster object, each node group providing the following information:
 - name : string
 - description : string
 - components[] : set of components ['NN', 'JT', ...] (modified from “node_processes”)
 - node_configs – any configuration settings provided in the plugin template
 - vm_requirements { } : map of n/v pairs
 - memory = 5G: required memory
 - disk = 10G : required disk space
 - cardinality = 1+ : legal instance counts
 - default_count = 10 : the default instance count
- controller populates UI based on node_groups that were populated into Cluster object
 - o for each node group, the controller creates a new node group row
 - o each row contains
 - Node Group : this is populated from `node_group.name`
 - Description : this is populated from `node_group.description`
 - VM Requirements : this is populated from `node_group.vm_requirements`
 - VM Flavor : Only flavors that satisfy vm requirements are populated
 - VM Count : Initially populated from `node_group.vm_requirements['default_count']`
 - for case such where cardinality is specified exactly such as '1'
 - o populate with value and don't allow edits
 - Configuration : This will be disabled for advanced configuration at this time.
 - In a standard configuration flow, this column will have a capability for the user to create or leverage a configuration template
- user updates VM Flavor and VM Count values
- UI/controller may do validation against VM Requirements values
- user clicks 'Launch Controller' button
 - o controller invokes `hadoop_plugin.validate(cluster)`
 - o controller invokes `hadoop_plugin.get_infra(cluster)`
 - o controller provisions VM's
 - o controller invokes `hadoop_plugin.configure_cluster(cluster)`
 - cluster will contain node_group/vm association
 - o controller invokes `hadoop_plugin.start_cluster(cluster)`

SPI changes

`populate_node_groups(cluster)`

- populates node groups in the passed in cluster object
- In a standard flow, this call may not be required since the user has selected a cluster template defining the node groups for the cluster. Therefore, the controller may have already set the node groups in the cluster.

convert(cluster, file)

- removed. Now handled by populate_node_groups

Object Model changes

node_group

- added/modified fields
- description : description of node group
- components : list of components ['JT', 'NN', ...]
- vm_requirements : map of vm requirement n/v pairs
 - memory : required memory
 - disk : required disk
 - cardinality : legal vm counts
 - default_count : default vm count
 - ...