

infrastructure.cluster.x-k8s.io APIs

[DCSCluster \[infrastructure.clus](#) [DCSIpHostnamePool \[infrastru](#) [DCSMachin](#)

[DCSMachineTemplate \[infrastructure.cluster.x-k8s.io/v1beta1\]](#)

DCSCluster [infrastructure.cluster.x-k8s.io/v1beta1]

infrastructure.cluster.x-k8s.io group

DCSCluster is the Schema for the dcsclusters API.

v1beta1 version

▼ spec object

DCSClusterSpec defines the desired state of DCSCluster.

▼ controlPlaneEndpoint object

ControlPlaneEndpoint represents the endpoint used to communicate with the control plane.

▼ host string required

host is the hostname on which the API server is serving.

▼ port integer required

port is the port on which the API server is serving.

▼ controlPlaneLoadBalancer object required

▼ host string required

▼ port integer required

▼ type string required

▼ vrid integer

▼ credentialSecretRef object required

CredentialSecretRef

▼ name string

Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. More info:

<https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names>



▼ networkType string required

▼ site string

▼ status object

DCSClusterStatus defines the observed state of DCSCluster.

▼ conditions `[]object`

Condition defines an observation of a Cluster API resource operational state.

▼ lastTransitionTime `string` required

lastTransitionTime is the last time the condition transitioned from one status to another. This should be when the underlying condition changed. If that is not known, then using the time when the API field changed is acceptable.

▼ message `string`

message is a human readable message indicating details about the transition. This field may be empty.

▼ reason `string`

reason is the reason for the condition's last transition in CamelCase. The specific API may choose whether or not this field is considered a guaranteed API. This field may be empty.

▼ severity `string`

severity provides an explicit classification of Reason code, so the users or machines can immediately understand the current situation and act accordingly. The Severity field MUST be set only when Status=False.

▼ status `string` required

status of the condition, one of True, False, Unknown.

▼ type `string` required

type of condition in CamelCase or in foo.example.com/CamelCase. Many .condition.type values are consistent across resources like Available, but because arbitrary conditions can be useful (see .node.status.conditions), the ability to deconflict is important.

▼ **failureDomains** **object**

FailureDomains is a slice of FailureDomains.

▼ **ready** **boolean**

DCSIPHostnamePool

[infrastructure.cluster.x-k8s.io/v1beta1]

infrastructure.cluster.x-k8s.io group

DCSIPHostnamePool is the Schema for the dcsiphostnamepools API.

v1beta1 version

▼ spec object

DCSIPHostnamePoolSpec defines the desired state of DCSIPHostnamePool.

▼ pool []object required

▼ dns string

▼ gateway string

▼ hostname string

▼ ip string

▼ ipv6 string

▼ **machineName** `string`

▼ **mask** `string`

▼ **status** `object`

DCSIPhostnamePoolStatus defines the observed state of DCSIPhostnamePool.

DCSMachine [infrastructure.cluster.x-k8s.io/v1beta1]

infrastructure.cluster.x-k8s.io group

DCSMachine is the Schema for the dcsmachines API.

v1beta1 version

▼ spec object

DCSMachineSpec defines the desired state of DCSMachine.

▼ ipHostPoolRef object required

LocalObjectReference contains enough information to let you locate the referenced object inside the same namespace.

▼ name string

Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. More info:

<https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names>



▼ location object

▼ name string

▼ type string required

▼ urn string

▼ providerID string

▼ resource object

▼ name string

▼ type string required

▼ urn string

▼ vmConfig object

▼ dcsMachineCpuSpec object required

▼ quantity integer required

▼ dcsMachineDiskSpec `[]object`**▼ datastoreClusterName** `string`**▼ datastoreName** `string`**▼ datastoreUrn** `string`**▼ format** `string`**▼ mountOptions** `[]string`**▼ options** `[]string`**▼ path** `string`**▼ quantity** `integer` `required`**▼ systemVolume** `boolean`**▼ wipeFilesystem** `boolean`

▼ **dcsmachineMemorySpec** **object** required

▼ **quantity** **integer** required

▼ **dvSwitchName** **string**

▼ **portGroupName** **string**

▼ **portGroupUrn** **string**

▼ **vmTemplateName** **string** required

▼ **vmUrn** **string**

▼ **status** **object**

DCSMachineStatus defines the observed state of DCSMachine.

▼ **addresses** **[]object**

MachineAddress contains information for the node's address.

▼ **address** **string** required

address is the machine address.

▼ **type** `string` required

type is the machine address type, one of Hostname, ExternalIP, InternalIP, ExternalDNS or InternalDNS.

▼ **cdRomFile** `string`

▼ **conditions** `[]object`

Condition defines an observation of a Cluster API resource operational state.

▼ **lastTransitionTime** `string` required

lastTransitionTime is the last time the condition transitioned from one status to another. This should be when the underlying condition changed. If that is not known, then using the time when the API field changed is acceptable.

▼ **message** `string`

message is a human readable message indicating details about the transition. This field may be empty.

▼ **reason** `string`

reason is the reason for the condition's last transition in CamelCase. The specific API may choose whether or not this field is considered a guaranteed API. This field may be empty.

▼ **severity** `string`

severity provides an explicit classification of Reason code, so the users or machines can immediately understand the current situation and act accordingly. The Severity field MUST be set only when Status=False.

▼ **status** `string` required

status of the condition, one of True, False, Unknown.

▼ **type** `string` required

type of condition in CamelCase or in foo.example.com/CamelCase. Many .condition.type values are consistent across resources like Available, but because arbitrary conditions can be useful (see .node.status.conditions), the ability to deconflict is important.

▼ **failureDomains** `object`

FailureDomains is a slice of FailureDomains.

▼ **networkConfig** `object`

▼ **dns** `string`

▼ **gateway** `string`

▼ **hostname** `string`

▼ **ip** `string`

▼ ipv6 `string`

▼ machineName `string`

▼ mask `string`

▼ ready `boolean`

DCSMachineTemplate

[infrastructure.cluster.x-k8s.io/v1beta1]

infrastructure.cluster.x-k8s.io group

DCSMachineTemplate is the Schema for the dcsmachinetemplates API.

v1beta1 version

▼ spec object

DCSMachineTemplateSpec defines the desired state of DCSMachineTemplate.

▼ template object required

▼ metadata object

ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create. This is a copy of customizable fields from `metav1.ObjectMeta`.

ObjectMeta is embedded in `Machine.Spec`, `MachineDeployment.Template` and `MachineSet.Template`, which are not top-level Kubernetes objects. Given that `metav1.ObjectMeta` has lots of special cases and read-only fields which end up in the generated CRD validation, having it as a subset simplifies the API and some issues that can impact user experience.

During the [upgrade to controller-tools@v2](#) for v1alpha2, we noticed a failure would occur running Cluster API test suite against the new CRDs, specifically `spec.metadata.creationTimestamp in body must be of type string: "null"`. The investigation showed that `controller-tools@v2` behaves differently than its previous version when handling types from `metav1` package.

In more details, we found that embedded (non-top level) types that embedded `metav1.ObjectMeta` had validation properties, including for `creationTimestamp` (`metav1.Time`). The `metav1.Time` type specifies a custom json marshaller that, when `IsZero()` is true, returns `null` which breaks validation because the field isn't marked as nullable.

In future versions, `controller-tools@v2` might allow overriding the type and validation for embedded types. When that happens, this hack should be revisited.

▼ annotations `object`

`annotations` is an unstructured key value map stored with a resource that may be set by external tools to store and retrieve arbitrary metadata. They are not queryable and should be preserved when modifying objects. More info: <http://kubernetes.io/docs/user-guide/annotations>

▼ labels `object`

`labels` is a map of string keys and values that can be used to organize and categorize (scope and select) objects. May match selectors of replication controllers and services. More info: <http://kubernetes.io/docs/user-guide/labels>

▼ spec `object` required

`DCSMachineSpec` defines the desired state of `DCSMachine`.

▼ ipHostPoolRef `object` required

`LocalObjectReference` contains enough information to let you locate the referenced object inside the same namespace.

▼ name `string`

Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this

type with an empty value here are almost certainly wrong. More info: <https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names> ↗

▼ location **object**

▼ name **string**

▼ type **string** required

▼ urn **string**

▼ providerID **string**

▼ resource **object**

▼ name **string**

▼ type **string** required

▼ urn **string**

▼ vmConfig **object**▼ dcsMachineCpuSpec **object** required▼ quantity **integer** required▼ dcsMachineDiskSpec **[]object**▼ datastoreClusterName **string**▼ datastoreName **string**▼ datastoreUrn **string**▼ format **string**▼ mountOptions **[]string**▼ options **[]string**▼ path **string**

▼ **quantity** `integer` `required`

▼ **systemVolume** `boolean`

▼ **wipeFilesystem** `boolean`

▼ **dcsmachineMemorySpec** `object` `required`

▼ **quantity** `integer` `required`

▼ **dvSwitchName** `string`

▼ **portGroupName** `string`

▼ **portGroupUrn** `string`

▼ **vmTemplateName** `string` `required`

▼ **vmUrn** `string`

▼ status **object**

DCSMachineTemplateStatus defines the observed state of DCSMachineTemplate.