Operator APIs

AmlCluster [amlclusters.aml.dev/v1alpha1]

AmlCluster [amlclusters.aml.dev/v1alpha1]

类型

`object`

规格

属性	类型	描述
`apiVersion`	`string`	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources ↗
`kind`	`string`	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗

属性	类型	描述
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	
`status`	`object`	

.spec

类型

`object`

必填

`values`

属性	类型	描述
`amlExperimental`	`object`	
`components`	`object`	
`values`	`object`	

.spec.amlExperimental

类型

`object`

.spec.components

类型

`object`

属性	类型	描述
`knativeServing`	`object`	Knative-Serving component configuration. Requires KServeless operator to be installed. Requires Service Mesh (istio).
`kserve`	`object`	Kserve component configuration. Requires KServeless operator to be installed.

.spec.components.knativeServing

描述

Knative-Serving component configuration. Requires KServeless operator to be installed. Requires Service Mesh (istio).

类型

`object`

属性	类型	描述
`ingressGateway`	`object`	IngressGateway allows to customize some parameters for the Istio Ingress Gateway that is bound to KNative-Serving.
`managementState`	`string`	
`namespace`	`string`	
`values`	`object`	

.spec.components.knativeServing.ingressGateway

描述

IngressGateway allows to customize some parameters for the Istio Ingress Gateway that is bound to KNative-Serving.

类型

`object`

必填

`domain`

属性	类型	描述
`certificate`	`object`	Certificate specifies configuration of the TLS certificate securing communication for the gateway.
`domain`	`string`	Domain specifies the host name for intercepting incoming requests. Most likely, you will want to use a wildcard name, like *.example.com. If you choose to generate a certificate, this is the domain used for the certificate request.

.spec.components.knativeServing.ingressGateway.certificate

描述

Certificate specifies configuration of the TLS certificate securing communication for the gateway.

类型

`object`

属性	类型	描述
`secretName`	`string`	SecretName specifies the name of the Kubernetes Secret resource that contains a TLS certificate secure HTTP communications for the KNative network.
`type`	`string`	Type specifies if the TLS certificate should be generated automatically, or if the certificate is provided by the user. Allowed values are: • SelfSigned: A certificate is going to be generated using an own private key. • Provided: Pre-existence of the TLS Secret (see SecretName) with a valid certificate is assumed. • ACPDefaultIngress: Default ingress certificate configured for ACP (ALB)

.spec.components.knativeServing.values

类型

`object`

.spec.components.kserve

描述

Kserve component configuration. Requires KServeless operator to be installed.

类型

`object`

属性	类型	描述
`managementState`	`string`	
`namespace`	`string`	
`values`	`object`	

.spec.components.kserve.values

类型

`object`

.spec.values

类型

`object`

属性	类型	描述
`amlApiServer`	`object`	
`amlServer`	`object`	
`amlService`	`object`	
`buildkitd`	`object`	
`experimentalFeatures`	`object`	
`global`	`object`	
`mysql`	`object`	

.spec.values.amlApiServer

类型

`object`

属性	类型	描述
`tolerations`	`array`	

.spec.values.amlApiServer.tolerations

类型

`array`

.spec.values.amlApiServer.tolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.

属性	类型	描述
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.spec.values.amlServer

类型

`object`

属性	类型	描述
`nodeSelector`	`array`	
`tolerations`	`array`	

.spec.values.amlServer.nodeSelector

类型

`array`

.spec.values.amlServer.nodeSelector[]

类型

`string`

.spec.values.amlServer.tolerations

类型

`array`

.spec.values.amlServer.tolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.

属性	类型	描述
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.spec.values.amlService

类型

`object`

属性	类型	描述
`appsBaseImage`	`string`	应用镜像构建的基础镜像
`appsGradioVersions`	`string`	应用镜像构建可选的 Gradio 的版本列表
`appsPipMirror`	`string`	应用镜像构建使用的 pip 镜像源
`appsStreamlitVersions`	`string`	应用镜像构建可选的 Streamlit 的版本列表
`gitDatasetUpdateCallbackUrl`	`string`	gitlab 回调 amlservice 的访问地址，用于在数据集更新之后，触发自动更新 preview 数据。
`imageBuildPvcWorkspaceSize`	`string`	镜像构建任务用于存放工作区文件的空间大小，比如存放模型文件
`imageBuildUsedPvcName`	`string`	镜像构建服务使用的pvc名称（非NFS），需提前创建好，建议申请大一点的空间用于缓存构建镜像，比如：150G
`nodeSelector`	`array`	
`notebookStorageClass`	`string`	自动创建的 notebook 使用的存储类

属性	类型	描述
`tolerations`	`array`	
`trainingPvcShareInNamespace`	`string`	训练使用的 PVC 是否在命名空间内共享，即一个命名空间仅使用一个通用的 PVC 缓存训练时的模型、数据集
`trainingPvcSize`	`string`	训练 PVC 大小。如果是命名空间共享的缓存 PVC，建议使用较大的空间，足够命名空间所有训练任务使用
`trainingPvcStorageClass`	`string`	训练 PVC 使用的存储类

.spec.values.amlService.nodeSelector

类型

`array`

.spec.values.amlService.nodeSelector[]

类型

`string`

.spec.values.amlService.tolerations

类型

`array`

.spec.values.amlService.tolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.

属性	类型	描述
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.spec.values.buildkitd

类型

`object`

属性	类型	描述
`storage`	`object`	
`tolerations`	`array`	

.spec.values.buildkitd.storage

类型

`object`

属性	类型	描述
`emptyDir`	`object`	当 type 为 emptyDir 时，需要提供以下配置
`pvc`	`object`	当 type 为 pvc 时，需要提供以下配置
`type`	`string`	存储类型，支持：emptyDir,pvc

.spec.values.buildkitd.storage.emptyDir

描述

当 type 为 emptyDir 时，需要提供以下配置

类型

`object`

属性	类型	描述
`sizeLimit`	`integer`	

.spec.values.buildkitd.storage.pvc

描述

当 type 为 pvc 时，需要提供以下配置

类型

`object`

属性	类型	描述
`accessMode`	`string`	
`storageClassName`	`string`	
`storageSize`	`string`	

.spec.values.buildkitd.tolerations

类型

`array`

.spec.values.buildkitd.tolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.

属性	类型	描述
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.spec.values.experimentalFeatures

类型

`object`

属性	类型	描述
`agents`	`boolean`	
`datasets`	`boolean`	
`imageBuilder`	`boolean`	
`tuneModels`	`boolean`	

.spec.values.global

类型

`object`

必填

`gitlabAdminTokenSecretRef` `gitlabBaseUrl`

属性	类型	描述
`defaultNamespace`	`string`	默认纳管的命名空间

属性	类型	描述
`defaultNodeSelector`	`array`	
`defaultTolerations`	`array`	
`deployFlavor`	`string`	DeployFlavor for single node or HA-cluster
`gitUserEmail`	`string`	后台 git push 使用的 email，仅在 git log 存储
`gitlabAdminTokenSecretRef`	`object`	Gitlab Admin 密钥地址
`gitlabBaseUrl`	`string`	Gitlab 访问地址
`imageRegistryPrefix`	`string`	harbor 项目名字
`images`	`object`	
`labelBaseDomain`	`string`	
`mysql`	`object`	MySQL configuration.
`registry`	`object`	

.spec.values.global.defaultNodeSelector

类型

`array`

.spec.values.global.defaultNodeSelector[]

类型

`string`

.spec.values.global.defaultTolerations

类型

`array`

.spec.values.global.defaultTolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.

属性	类型	描述
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.spec.values.global.gitlabAdminTokenSecretRef

描述

Gitlab Admin 密钥地址

类型

`object`

必填

`name` `namespace`

属性	类型	描述
`name`	`string`	Name is the name of resource being referenced
`namespace`	`string`	Namespace is the namespace of resource being referenced

.spec.values.global.images

类型

`object`

.spec.values.global.mysql

描述

MySQL configuration.

类型

`object`

必填

`database` `host` `user`

属性	类型	描述
`database`	`string`	Database to use.
`host`	`string`	Host where the database server is located.

属性	类型	描述
`passwordSecretRef`	`object`	Password reference to a secret containing the password.
`port`	`integer`	MySQL port to use, default is usually OK. (default: 3306)
`user`	`string`	User to log in as.

.spec.values.global.mysql.passwordSecretRef

描述

Password reference to a secret containing the password.

类型

`object`

必填

`name` `namespace`

属性	类型	描述
`name`	`string`	Name is the name of resource being referenced
`namespace`	`string`	Namespace is the namespace of resource being referenced

.spec.values.global.registry

类型

`object`

属性	类型	描述
`address`	`string`	

.spec.values.mysql

类型

`object`

属性	类型	描述
`pvc`	`object`	
`tolerations`	`array`	

.spec.values.mysql.pvc

类型

`object`

属性	类型	描述
`storageClassName`	`string`	
`storageSize`	`integer`	

.spec.values.mysql.tolerations

类型

`array`

.spec.values.mysql.tolerations[]

描述

The pod this Toleration is attached to tolerates any taint that matches the triple <key,value,effect> using the matching operator <operator>.

类型

`object`

属性	类型	描述
`effect`	`string`	Effect indicates the taint effect to match. Empty means match all taint effects. When specified, allowed values are NoSchedule, PreferNoSchedule and NoExecute.
`key`	`string`	Key is the taint key that the toleration applies to. Empty means match all taint keys. If the key is empty, operator must be Exists; this combination means to match all values and all keys.
`operator`	`string`	Operator represents a key's relationship to the value. Valid operators are Exists and Equal. Defaults to Equal. Exists is equivalent to wildcard for value, so that a pod can tolerate all taints of a particular category.
`tolerationSeconds`	`integer`	TolerationSeconds represents the period of time the toleration (which must be of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default, it is not set, which means tolerate the taint

属性	类型	描述
		forever (do not evict). Zero and negative values will be treated as 0 (evict immediately) by the system.
`value`	`string`	Value is the taint value the toleration matches to. If the operator is Exists, the value should be empty, otherwise just a regular string.

.status

类型

`object`

属性	类型	描述
`components`	`object`	Expose component's specific status
`conditions`	`array`	
`lastHandledReconcileAt`	`string`	LastHandledReconcileAt holds the value of the most recent reconcile request value, so a change of the annotation value can be detected.
`observedGeneration`	`integer`	The observed generation of the object. When this matches the object's metadata.generation, it indicates the status is up-to-date.

.status.components

描述

Expose component's specific status

类型

`object`

属性	类型	描述
`aml`	`object`	
`kServe`	`object`	
`knativeServing`	`object`	

.status.components.aml

类型

`object`

属性	类型	描述
`conditions`	`array`	Conditions is a set of Condition instances.
`deployedRelease`	`object`	
`managementState`	`string`	

.status.components.aml.conditions

描述

Conditions is a set of Condition instances.

类型

`array`

.status.components.aml.conditions[]

描述

Condition represents an observation of an object's state. Conditions are an extension mechanism intended to be used when the details of an observation are not a priori known or would not apply to all instances of a given Kind. Conditions should be added to explicitly convey properties that users and components care about rather than requiring those properties to be inferred from other observations. Once defined, the meaning of a Condition can not be changed arbitrarily - it becomes part of the API, and has the same backwards- and forwards-compatibility concerns of any other part of the API.

类型

`object`

必填

`status` `type`

属性	类型	描述
`lastTransitionTime`	`string`	
`message`	`string`	
`reason`	`string`	ConditionReason is intended to be a one-word, CamelCase representation of the category of cause of the current status. It is intended to be used in concise output, such as one-line kubectl get output, and in summarizing occurrences of causes.
`status`	`string`	
`type`	`string`	ConditionType is the type of the condition and is typically a CamelCased word or short phrase.

属性	类型	描述
		Condition types should indicate state in the "abnormal-true" polarity. For example, if the condition indicates when a policy is invalid, the "is valid" case is probably the norm, so the condition should be called "Invalid".

.status.components.aml.deployedRelease

类型

`object`

属性	类型	描述
`name`	`string`	

.status.components.kServe

类型

`object`

属性	类型	描述
`conditions`	`array`	Conditions is a set of Condition instances.
`deployedRelease`	`object`	
`managementState`	`string`	

.status.components.kServe.conditions

描述

Conditions is a set of Condition instances.

类型

`array`

.status.components.kServe.conditions[]

描述

Condition represents an observation of an object's state. Conditions are an extension mechanism intended to be used when the details of an observation are not a priori known or would not apply to all instances of a given Kind. Conditions should be added to explicitly convey properties that users and components care about rather than requiring those properties to be inferred from other observations. Once defined, the meaning of a Condition can not be changed arbitrarily - it becomes part of the API, and has the same backwards- and forwards-compatibility concerns of any other part of the API.

类型

`object`

必填

`status` `type`

属性	类型	描述
`lastTransitionTime`	`string`	
`message`	`string`	
`reason`	`string`	ConditionReason is intended to be a one-word, CamelCase representation of the category of cause of the current status. It is intended to be used in concise output, such as one-line kubectl get output, and in summarizing occurrences of causes.
`status`	`string`	

属性	类型	描述
`type`	`string`	ConditionType is the type of the condition and is typically a CamelCased word or short phrase. Condition types should indicate state in the "abnormal-true" polarity. For example, if the condition indicates when a policy is invalid, the "is valid" case is probably the norm, so the condition should be called "Invalid".

.status.components.kServe.deployedRelease

类型

`object`

属性	类型	描述
`name`	`string`	

.status.components.knativeServing

类型

`object`

属性	类型	描述
`conditions`	`array`	Conditions is a set of Condition instances.
`deployedRelease`	`object`	
`managementState`	`string`	

.status.components.knativeServing.conditions

描述

Conditions is a set of Condition instances.

类型

`array`

.status.components.knativeServing.conditions[]

描述

Condition represents an observation of an object's state. Conditions are an extension mechanism intended to be used when the details of an observation are not a priori known or would not apply to all instances of a given Kind. Conditions should be added to explicitly convey properties that users and components care about rather than requiring those properties to be inferred from other observations. Once defined, the meaning of a Condition can not be changed arbitrarily - it becomes part of the API, and has the same backwards- and forwards-compatibility concerns of any other part of the API.

类型

`object`

必填

`status` `type`

属性	类型	描述
`lastTransitionTime`	`string`	
`message`	`string`	

属性	类型	描述
`reason`	`string`	ConditionReason is intended to be a one-word, CamelCase representation of the category of cause of the current status. It is intended to be used in concise output, such as one-line kubectl get output, and in summarizing occurrences of causes.
`status`	`string`	
`type`	`string`	ConditionType is the type of the condition and is typically a CamelCased word or short phrase. Condition types should indicate state in the "abnormal-true" polarity. For example, if the condition indicates when a policy is invalid, the "is valid" case is probably the norm, so the condition should be called "Invalid".

.status.components.knativeServing.deployedRelease

类型

`object`

属性	类型	描述
`name`	`string`	

.status.conditions

类型

`array`

.status.conditions[]

描述

Condition contains details for one aspect of the current state of this API Resource. --- This struct is intended for direct use as an array at the field path .status.conditions. For example, type FooStatus struct{ // Represents the observations of a foo's current state. // Known .status.conditions.type are: "Available", "Progressing", and "Degraded" // +patchMergeKey=type // +patchStrategy=merge // +listType=map // +listMapKey=type Conditions []metav1.Condition `json:"conditions,omitempty" patchStrategy:"merge" patchMergeKey:"type" protobuf:"bytes,1,rep,name=conditions"` // other fields }

类型

`object`

必填

`lastTransitionTime` `message` `reason` `status` `type`

属性	类型	描述
`lastTransitionTime`	`string`	lastTransitionTime is the last time the condition transitio from one status to another. This should be when the und condition changed. If that is not known, then using the ti when the API field changed is acceptable.
`message`	`string`	message is a human readable message indicating detai the transition. This may be an empty string.

属性	类型	描述
`observedGeneration`	`integer`	observedGeneration represents the .metadata.generatio the condition was set based upon. For instance, if .metadata.generation is currently 12, but the .status.conditions[x].observedGeneration is 9, the condi out of date with respect to the current state of the instan
`reason`	`string`	reason contains a programmatic identifier indicating the for the condition's last transition. Producers of specific c types may define expected values and meanings for this and whether the values are considered a guaranteed AP value should be a CamelCase string. This field may not empty.
`status`	`string`	status of the condition, one of True, False, Unknown.
`type`	`string`	# type of condition in CamelCase or in foo.example.com/CamelCa Many .condition.type values are consistent across resou like Available, but because arbitrary conditions can be u (see .node.status.conditions), the ability to deconflict is important. The regex it matches is (dns1123Subdomain (qualifiedNameFmt)

API 端点

可用的 API 端点如下：

- `/apis/amlclusters.aml.dev/v1alpha1/amlclusters`
  - `DELETE`: delete collection of AmlCluster
  - `GET`: list objects of kind AmlCluster
  - `POST`: create a new AmlCluster
- `/apis/amlclusters.aml.dev/v1alpha1/amlclusters/{name}`
  - `DELETE`: delete the specified AmlCluster
  - `GET`: read the specified AmlCluster
  - `PATCH`: partially update the specified AmlCluster
  - `PUT`: replace the specified AmlCluster
- `/apis/amlclusters.aml.dev/v1alpha1/amlclusters/{name}/status`
  - `GET`: read status of the specified AmlCluster
  - `PATCH`: partially update status of the specified AmlCluster
  - `PUT`: replace status of the specified AmlCluster

/apis/amlclusters.aml.dev/v1alpha1/amlclusters

HTTP 方法

`DELETE`

描述

delete collection of AmlCluster

HTTP 响应

HTTP 状态码	响应体
200 - OK	`Status↗` schema

HTTP 状态码	响应体
401 - Unauthorized	Empty

HTTP 方法

`GET`

描述

list objects of kind AmlCluster

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlClusterList` schema
401 - Unauthorized	Empty

HTTP 方法

`POST`

描述

create a new AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a

参数	类型	描述
		warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

请求体参数

参数	类型	描述
`body`	`AmlCluster` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
201 - Created	`AmlCluster` schema
202 - Accepted	`AmlCluster` schema
401 - Unauthorized	Empty

/apis/amlclusters.aml.dev/v1alpha1/amlclusters/{name}

HTTP 方法

`DELETE`

描述

delete the specified AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP 响应

HTTP 状态码	响应体
200 - OK	`Status↗` schema
202 - Accepted	`Status↗` schema
401 - Unauthorized	Empty

HTTP 方法

`GET`

描述

read the specified AmlCluster

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
401 - Unauthorized	Empty

HTTP 方法

`PATCH`

描述

partially update the specified AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace the specified AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

请求体参数

参数	类型	描述
`body`	`AmlCluster` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema

HTTP 状态码	响应体
201 - Created	`AmlCluster` schema
401 - Unauthorized	Empty

/apis/amlclusters.aml.dev/v1alpha1/amlclusters/{name}/status

HTTP 方法

`GET`

描述

read status of the specified AmlCluster

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
401 - Unauthorized	Empty

HTTP 方法

`PATCH`

描述

partially update status of the specified AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

参数	类型	描述
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace status of the specified AmlCluster

查询参数

参数	类型	描述
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun

参数	类型	描述
		directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
`fieldValidation`	`string`	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

请求体参数

参数	类型	描述
`body`	`AmlCluster` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`AmlCluster` schema
201 - Created	`AmlCluster` schema
401 - Unauthorized	Empty