Platform Base APIs

ClusterModule [cluster.alauda.i CommonConfig [operator.alaud ProductBas

ProductConfig [alauda.io/v1alp ProductUpgrade [product.alauda.io/v1alpha1]

ClusterModule [cluster.alauda.io/v1alpha1]

描述

ClusterModule is the Schema for the clustermodules API

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 ↗

属性	类型	描述
`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 ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	ClusterModuleSpec defines the desired state of ClusterModule
`status`	`object`	ClusterModuleStatus defines the observed state of ClusterModule

.spec

描述

ClusterModuleSpec defines the desired state of ClusterModule

类型

`object`

属性	类型	描述
`parallelUpgrade`	`boolean`	ParallelUpgrade stands for if the upgrading could execute parallelly.
`upgradeModulesInfos`	`array`	UpgradeModuleInfos stands for the information to upgrade the modules
`valuesOverride`	`object`	ValuesOverride ...
`version`	`string`	

.spec.upgradeModulesInfos

描述

UpgradeModuleInfos stands for the information to upgrade the modules

类型

`array`

.spec.upgradeModulesInfos[]

描述

UpgradeModuleInfo ...

类型

`object`

必填

`name` `version`

属性	类型	描述
`config`	`object`	
`name`	`string`	
`version`	`string`	

.spec.upgradeModulesInfos[].config

类型

`object`

.spec.valuesOverride

描述

ValuesOverride ...

类型

`object`

.status

描述

ClusterModuleStatus defines the observed state of ClusterModule

类型

`object`

必填

`editable` `upgradable`

属性	类型	描述
`appReleases`	`object`	

属性	类型	描述
`base`	`object`	ModuleStatus ...
`editable`	`boolean`	
`needUpgrade`	`boolean`	
`sync`	`string`	SyncMethod ...
`unableUpgradeReason`	`string`	
`upgradable`	`boolean`	

.status.appReleases

类型

`object`

.status.base

描述

ModuleStatus ...

类型

`object`

必填

`deployStatus` `runningStatus`

属性	类型	描述
`appReleases`	`array`	

属性	类型	描述
`deployStatus`	`string`	DeployStatus ...
`deployTime`	`string`	
`lastProbeTime`	`string`	
`lastTransitionTime`	`string`	
`logPath`	`string`	
`message`	`string`	
`reason`	`string`	
`runningStatus`	`string`	RunningStatus ...
`version`	`string`	

.status.base.appReleases

类型

`array`

.status.base.appReleases[]

类型

`string`

API 端点

可用的 API 端点如下：

- `/apis/cluster.alauda.io/v1alpha2/clustermodules`
  - `DELETE`: delete collection of ClusterModule
  - `GET`: list objects of kind ClusterModule
  - `POST`: create a new ClusterModule
- `/apis/cluster.alauda.io/v1alpha2/clustermodules/{name}`
  - `DELETE`: delete the specified ClusterModule
  - `GET`: read the specified ClusterModule
  - `PATCH`: partially update the specified ClusterModule
  - `PUT`: replace the specified ClusterModule
- `/apis/cluster.alauda.io/v1alpha2/clustermodules/{name}/status`
  - `GET`: read status of the specified ClusterModule
  - `PATCH`: partially update status of the specified ClusterModule
  - `PUT`: replace status of the specified ClusterModule

/apis/cluster.alauda.io/v1alpha2/clustermodules

HTTP 方法

`DELETE`

描述

delete collection of ClusterModule

HTTP 响应

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

HTTP 方法

`GET`

描述

list objects of kind ClusterModule

HTTP 响应

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

HTTP 方法

`POST`

描述

create a new ClusterModule

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

参数	类型	描述
		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`	`ClusterModule` schema	`application/json` formatted

HTTP 响应

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

/apis/cluster.alauda.io/v1alpha2/clustermodules/{name}

HTTP 方法

`DELETE`

描述

delete the specified ClusterModule

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

参数	类型	描述
		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 ClusterModule

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update the specified ClusterModule

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.

HTTP 响应

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

HTTP 方法

`PUT`

描述

replace the specified ClusterModule

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`	`ClusterModule` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`ClusterModule` schema
201 - Created	`ClusterModule` schema

HTTP 状态码	响应体
401 - Unauthorized	Empty

/apis/cluster.alauda.io/v1alpha2/clustermodules/{name}/status

HTTP 方法

`GET`

描述

read status of the specified ClusterModule

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update status of the specified ClusterModule

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

参数	类型	描述
		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	`ClusterModule` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace status of the specified ClusterModule

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`	`ClusterModule` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`ClusterModule` schema
201 - Created	`ClusterModule` schema

HTTP 状态码	响应体
401 - Unauthorized	Empty

CommonConfig [operator.alauda.io/v1alpha1]

描述

CommonConfig is the Schema for the commonconfigs API

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:

属性	类型	描述
		https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	CommonConfigSpec defines the desired state of CommonConfig
`status`	`object`	CommonConfigStatus defines the observed state of CommonConfig

.spec

描述

CommonConfigSpec defines the desired state of CommonConfig

类型

`object`

必填

`etcd` `global` `oidc` `release` `repositories`

属性	类型	描述
`elasticsearch`	`object`	Elasticsearch Config

属性	类型	描述
`etcd`	`object`	Etcd Config
`ext`	`object`	Ext Config
`global`	`object`	Global configuration
`kafka`	`object`	Kafka Config
`oidc`	`object`	OIDC Config
`release`	`string`	Release verison
`repositories`	`array`	Repositories Config

.spec.elasticsearch

描述

Elasticsearch Config

类型

`object`

属性	类型	描述
`host`	`string`	Server address
`nodes`	`array`	Log nodes
`password`	`string`	Password
`username`	`string`	Uername

.spec.elasticsearch.nodes

描述

Log nodes

类型

`array`

.spec.elasticsearch.nodes[]

类型

`string`

.spec.etcd

描述

Etcd Config

类型

`object`

必填

`ectdCaSecret` `ectdPeerSecret` `servers`

属性	类型	描述
`ectdCaSecret`	`object`	Etcd Ca Secret
`ectdPeerSecret`	`object`	Etcd Peer secret
`servers`	`array`	Etcd servers

.spec.etcd.ectdCaSecret

描述

Etcd Ca Secret

类型

`object`

必填

`secretSource`

属性	类型	描述
`secretSource`	`object`	

.spec.etcd.ectdCaSecret.secretSource

类型

`object`

必填

`tls.crt` `tls.key`

属性	类型	描述
`tls.crt`	`string`	
`tls.key`	`string`	

.spec.etcd.ectdPeerSecret

描述

Etcd Peer secret

类型

`object`

必填

`secretSource`

属性	类型	描述
`secretSource`	`object`	

.spec.etcd.ectdPeerSecret.secretSource

类型

`object`

必填

`tls.crt` `tls.key`

属性	类型	描述
`tls.crt`	`string`	
`tls.key`	`string`	

.spec.etcd.servers

描述

Etcd servers

类型

`array`

.spec.etcd.servers[]

类型

`string`

.spec.ext

描述

Ext Config

类型

`object`

.spec.global

描述

Global configuration

类型

`object`

必填

`apiAddress` `defaultAdmin` `erebusApiAddress` `host` `labelBaseDomain` `replicas` `tlsSecret`

属性	类型	描述
`apiAddress`	`string`	Platform api address
`defaultAdmin`	`string`	The platform deploys a management account by default, and the default email admin@cpaas.io ↗
`erebusApiAddress`	`string`	Platform api gateway address
`host`	`string`	Platform access host address, support domain name or IP
`isOCP`	`boolean`	Whether to deploy on openshift cluster
`labelBaseDomain`	`string`	Platform resource instance label field, the default is cpaas.io
`namespace`	`string`	The namespace deployed by the platform, default cpaas-system
`replicas`	`integer`	The number of instances of platform common deployment components, the default is 2
`scheme`	`string`	Platform access protocol, support http or https

属性	类型	描述
`tlsSecret`	`object`	Tls Secret

.spec.global.tlsSecret

描述

Tls Secret

类型

`object`

必填

`secretSource`

属性	类型	描述
`secretSource`	`object`	

.spec.global.tlsSecret.secretSource

类型

`object`

必填

`tls.crt` `tls.key`

属性	类型	描述
`tls.crt`	`string`	
`tls.key`	`string`	

.spec.kafka

描述

Kafka Config

类型

`object`

必填

`auth` `host` `kafka_password` `kafka_user` `zk_acl_password` `zk_password` `zk_user`

属性	类型	描述
`auth`	`boolean`	Whether to enable authentication
`host`	`string`	Server address
`kafka_password`	`string`	
`kafka_user`	`string`	
`zk_acl_password`	`string`	
`zk_password`	`string`	
`zk_user`	`string`	

.spec.oidc

描述

OIDC Config

类型

`object`

必填

`clientID` `clientSecret` `issuer` `responseType` `scopes`

属性	类型	描述
`clientID`	`string`	OIDC Client ID
`clientSecret`	`string`	OIDC Client Secret
`issuer`	`string`	OIDC server address
`responseType`	`string`	OIDC Response Type
`scopes`	`string`	OIDC Scopes

.spec.repositories

描述

Repositories Config

类型

`array`

.spec.repositories[]

描述

Repository Config

类型

`object`

必填

`type` `url`

属性	类型	描述
`type`	`string`	Repository type
`url`	`string`	Repository url

.status

描述

CommonConfigStatus defines the observed state of CommonConfig

类型

`object`

API 端点

可用的 API 端点如下：

- `/apis/operator.alauda.io/v1alpha1/commonconfigs`
  - `DELETE`: delete collection of CommonConfig
  - `GET`: list objects of kind CommonConfig
  - `POST`: create a new CommonConfig
- `/apis/operator.alauda.io/v1alpha1/commonconfigs/{name}`
  - `DELETE`: delete the specified CommonConfig
  - `GET`: read the specified CommonConfig
  - `PATCH`: partially update the specified CommonConfig

- `PUT`: replace the specified CommonConfig

/apis/operator.alauda.io/v1alpha1/commonconfigs

HTTP 方法

`DELETE`

描述

delete collection of CommonConfig

HTTP 响应

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

HTTP 方法

`GET`

描述

list objects of kind CommonConfig

HTTP 响应

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

HTTP 方法

`POST`

描述

create a new CommonConfig

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`	`CommonConfig` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`CommonConfig` schema
201 - Created	`CommonConfig` schema

HTTP 状态码	响应体
202 - Accepted	`CommonConfig` schema
401 - Unauthorized	Empty

/apis/operator.alauda.io/v1alpha1/commonconfigs/{name}

HTTP 方法

`DELETE`

描述

delete the specified CommonConfig

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 CommonConfig

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update the specified CommonConfig

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	`CommonConfig` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace the specified CommonConfig

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`	`CommonConfig` schema	`application/json` formatted

HTTP 响应

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

ProductBase [product.alauda.io/v1alpha1]

描述

ProductBase is the Schema for the productbases API

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 ↗

属性	类型	描述
`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 ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	ProductBaseSpec defines the desired state of ProductBase
`status`	`object`	ProductBaseStatus defines the observed state of ProductBase

.spec

描述

ProductBaseSpec defines the desired state of ProductBase

类型

`object`

必填

`defaultAdmin` `http` `platformURL` `registry` `version`

属性	类型	描述
`alternativeURLs`	`array`	AlternativeURLs is the platform's alternative urls
`defaultAdmin`	`object`	DefaultAdmin stands for the admin's config
`defaultRedirectPath`	`string`	DefaultRedirectPath stands for the default redirect path
`globalConfig`	`object`	GlobalConfig stands for the global config
`http`	`object`	HTTP stands for the http config
`ingress`	`object`	Ingress stands for the ingress configs
`platformPublicURL`	`string`	PlatformPublicURL stands the platform's public url
`platformURL`	`string`	PlatformURL is the platform's access url
`productBrand`	`string`	ProductBrand values: TKE、ACP、Common
`registry`	`object`	Registry is Docker Registry's Config

属性	类型	描述
`replicas`	`integer`	Replicas stands for the deploy replicas
`version`	`string`	Version stands for this product's version to deploy

.spec.alternativeURLs

描述

AlternativeURLs is the platform's alternative urls

类型

`array`

.spec.alternativeURLs[]

类型

`string`

.spec.defaultAdmin

描述

DefaultAdmin stands for the admin's config

类型

`object`

必填

`account` `email`

属性	类型	描述
`account`	`string`	Account stands for admin's account
`email`	`string`	Email stands for admin's email
`hash`	`string`	Hash stands for admin's password hash

.spec.globalConfig

描述

GlobalConfig stands for the global config

类型

`object`

属性	类型	描述
`labels`	`object`	

.spec.globalConfig.labels

类型

`object`

.spec.http

描述

HTTP stands for the http config

类型

`object`

必填

`forceRedirectHttps`

属性	类型	描述
`forceRedirectHttps`	`boolean`	ForceRedirectHTTPS stands for if force redirect to https
`httpPort`	`integer`	HTTPPort is the port of http
`httpsPort`	`integer`	HTTPPort is the port of https
`tls`	`object`	TLS statnds for the server's TLS Certificate

.spec.http.tls

描述

TLS statnds for the server's TLS Certificate

类型

`object`

属性	类型	描述
`secretRef`	`object`	SecretRef is the reference for tls secret

.spec.http.tls.secretRef

描述

SecretRef is the reference for tls secret

类型

`object`

必填

`name`

属性	类型	描述
`name`	`string`	Name is the name of secret

.spec.ingress

描述

Ingress stands for the ingress configs

类型

`object`

必填

`controller` `host` `ingressClassName`

属性	类型	描述
`annotations`	`object`	
`controller`	`object`	Controller stands for the ingress controller configs
`host`	`string`	Host stands for the host config in ingress

属性	类型	描述
`ingressClassName`	`string`	IngressClassName stands for the ingressClassName of the ingress

.spec.ingress.annotations

类型

`object`

.spec.ingress.controller

描述

Controller stands for the ingress controller configs

类型

`object`

必填

`install`

属性	类型	描述
`install`	`boolean`	Install stands for if install ingress controller
`nodes`	`array`	Nodes stands for the ingress controller's running node

.spec.ingress.controller.nodes

描述

Nodes stands for the ingress controller's running node

类型

`array`

.spec.ingress.controller.nodes[]

类型

`string`

.spec.registry

描述

Registry is Docker Registry's Config

类型

`object`

必填

`address` `preferPlatformURL`

属性	类型	描述
`address`	`string`	Address is Docker Registry's address
`external`	`boolean`	External stands for a external registry
`preferPlatformURL`	`boolean`	PreferPlatformURL stands for business cluster prefer platform URL as registry address
`secretRef`	`object`	SecretRef is the reference for pullImageSecret

.spec.registry.secretRef

描述

SecretRef is the reference for pullImageSecret

类型

`object`

必填

`name`

属性	类型	描述
`name`	`string`	Name is the name of secret

.status

描述

ProductBaseStatus defines the observed state of ProductBase

类型

`object`

必填

`phase` `version`

属性	类型	描述
`artifacts`	`array`	Artifacts stands for the product's artifacts
`conditions`	`array`	Conditions contains this product's conditions

属性	类型	描述
`globalID`	`string`	GlobalID stands for the global Id
`phase`	`string`	Phase stands for this product's phase
`replicas`	`integer`	Replicas stands for the product workload replicas
`version`	`string`	Version statnds for this product's current version

.status.artifacts

描述

Artifacts stands for the product's artifacts

类型

`array`

.status.artifacts[]

描述

Artifact stands for the artifact's info

类型

`object`

必填

`channels` `name` `packageType`

属性	类型	描述
`channels`	`array`	Channels stands for the artifact's channels
`name`	`string`	Name stands for the artifact's name
`packageType`	`string`	PackageType stands for the artifact's package type

.status.artifacts[].channels

描述

Channels stands for the artifact's channels

类型

`array`

.status.artifacts[].channels[]

描述

Channel stands for the artifact's channel

类型

`object`

必填

`artifactStatus` `channel` `default` `displayName` `repository` `tag`

属性	类型	描述
`artifactStatus`	`string`	ArtifactStatus stands for the channel's artifact status
`brief`	`object`	Brief stands for the artifact's brief
`catalogSource`	`string`	CatalogSource stands for the catalog source to get the operator bundle
`categories`	`array`	Categories stands for the artifact's categories, such as: AI, Database, etc.
`channel`	`string`	Channel stands for the channel's name
`default`	`boolean`	Default stands for the artifact's default channel
`description`	`object`	Description stands for the channel's description
`digest`	`string`	Digest stands for the channel's digest

属性	类型	描述
`displayName`	`object`	DisplayName stands for the artifact's display name
`hidden`	`boolean`	Hidden stands for the artifact is hidden on the Marketplace's page
`packIgnore`	`boolean`	PackIgnore indicates whether to skip packaging
`provider`	`object`	Provider stands for the artifact's provider
`providerType`	`string`	ProviderType stands for the artifact's provider type, such as: platform, certified, custom, community
`repository`	`string`	Repository stands for the channel's repository
`supportedArchitectures`	`array`	SupportedArchitectures stands for the supported architectures, such as: amd64, arm64, etc.

属性	类型	描述
`supportedPlatformVersions`	`array`	SupportedPlatformVersions stands for the supported platform versions, such as: v4.0, v4.1, etc.
`supportedProtocolStacks`	`array`	SupportedProtocolStacks stands for the supported protocol stacks, such as: IPv4, IPv6, DualStack, etc.
`tag`	`string`	Tag stands for the channel's tag

.status.artifacts[].channels[].brief

描述

Brief stands for the artifact's brief

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.status.artifacts[].channels[].categories

描述

Categories stands for the artifact's categories, such as: AI, Database, etc.

类型

`array`

.status.artifacts[].channels[].categories[]

类型

`string`

.status.artifacts[].channels[].description

描述

Description stands for the channel's description

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.status.artifacts[].channels[].displayName

描述

DisplayName stands for the artifact's display name

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.status.artifacts[].channels[].provider

描述

Provider stands for the artifact's provider

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.status.artifacts[].channels[].supportedArchitectures

描述

SupportedArchitectures stands for the supported architectures, such as: amd64, arm64, etc.

类型

`array`

.status.artifacts[].channels[].supportedArchitectures[]

类型

`string`

.status.artifacts[].channels[].supportedPlatformVersions

描述

SupportedPlatformVersions stands for the supported platform versions, such as: v4.0, v4.1, etc.

类型

`array`

.status.artifacts[].channels[].supportedPlatformVersions[]

类型

`string`

.status.artifacts[].channels[].supportedProtocolStacks

描述

SupportedProtocolStacks stands for the supported protocol stacks, such as: IPv4, IPv6, DualStack, etc.

类型

`array`

.status.artifacts[].channels[].supportedProtocolStacks[]

类型

`string`

.status.conditions

描述

Conditions contains this product's conditions

类型

`array`

.status.conditions[]

描述

ProductBaseCondition ...

类型

`object`

必填

`status` `type`

属性	类型	描述
`lastHeartbeatTime`	`string`	LastHeartbeatTime stands for the timestamp to check Condition
`lastTransitionTime`	`string`	LastTransitionTime stands for the timestamp of Condition's status changed
`message`	`string`	Message stands for Condition's message
`reason`	`string`	Reason stands for Condition's reason
`status`	`string`	Status stands for Condition's status

属性	类型	描述
`type`	`string`	Type stands for Condition's type

API 端点

可用的 API 端点如下：

- `/apis/product.alauda.io/v1alpha2/productbases`
  - `DELETE`: delete collection of ProductBase
  - `GET`: list objects of kind ProductBase
  - `POST`: create a new ProductBase
- `/apis/product.alauda.io/v1alpha2/productbases/{name}`
  - `DELETE`: delete the specified ProductBase
  - `GET`: read the specified ProductBase
  - `PATCH`: partially update the specified ProductBase
  - `PUT`: replace the specified ProductBase
- `/apis/product.alauda.io/v1alpha2/productbases/{name}/status`
  - `GET`: read status of the specified ProductBase
  - `PATCH`: partially update status of the specified ProductBase
  - `PUT`: replace status of the specified ProductBase

/apis/product.alauda.io/v1alpha2/productbases

HTTP 方法

`DELETE`

描述

delete collection of ProductBase

HTTP 响应

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

HTTP 方法

`GET`

描述

list objects of kind ProductBase

HTTP 响应

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

HTTP 方法

`POST`

描述

create a new ProductBase

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

参数	类型	描述
		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`	`ProductBase` schema	`application/json` formatted

HTTP 响应

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

/apis/product.alauda.io/v1alpha2/productbases/{name}

HTTP 方法

`DELETE`

描述

delete the specified ProductBase

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 ProductBase

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update the specified ProductBase

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	`ProductBase` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace the specified ProductBase

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`	`ProductBase` schema	`application/json` formatted

HTTP 响应

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

/apis/product.alauda.io/v1alpha2/productbases/{name}/status

HTTP 方法

`GET`

描述

read status of the specified ProductBase

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update status of the specified ProductBase

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.

HTTP 响应

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

HTTP 方法

`PUT`

描述

replace status of the specified ProductBase

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`	`ProductBase` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`ProductBase` schema
201 - Created	`ProductBase` schema

HTTP 状态码	响应体
401 - Unauthorized	Empty

ProductConfig [alauda.io/v1alpha1]

描述

ProductConfig is the Schema for the productconfigs API

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 ↗

属性	类型	描述
`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 ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	ProductConfigSpec defines the desired state of ProductConfig
`status`	`object`	ProductConfigStatus defines the observed state of ProductConfig

.spec

描述

ProductConfigSpec defines the desired state of ProductConfig

类型

`object`

必填

`availableVersion` `brief` `description` `displayName` `displayUIConfig` `installUIConfig` `logo` `platformVersionMatch` `productResource` `type`

属性	类型	描述
`availableVersion`	`string`	AvailableVersion stands for the available version to deploy
`brief`	`object`	Brief stands for the product's brief
`chartVersions`	`array`	ChartVersions stands for the charts when deploying, they are optional and will be added to the Prouct CR
`commonConfigRef`	`array`	CommonConfigRef stands for the reference from CommonConfig to Product CR
`description`	`object`	Description stands for the product's description
`displayName`	`object`	DisplayName stands for the product's display name
`displayUIConfig`	`object`	DisplayUIConfig stands for the web UI type when showing product's detail information
`installUIConfig`	`object`	InstallUIConfig stands for the web UI config when deploying product

属性	类型	描述
`isDisplay`	`boolean`	IsDisplay stands for if the product displaying on the products list
`logo`	`string`	Logo stands for the product logo with base64 encoded data uri
`platformVersionMatch`	`string`	PlatformVersionMatch stands for the product need match the platform version
`productResource`	`object`	ProductResource defines the target product CRD type
`singleApp`	`object`	SingleAppConfig stands for the config for SignleApp type
`subApp`	`object`	SubApp stands for the config for SubApp type
`type`	`string`	Type stands for the product's type
`upgradeUIConfig`	`array`	UpgradeUIConfig stands for the web UI config when upgrading product

属性	类型	描述
`upgradeableVersions`	`array`	UpgradeableVersions stands for the versions can upgrade from

.spec.brief

描述

Brief stands for the product's brief

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.spec.chartVersions

描述

ChartVersions stands for the charts when deploying, they are optional and will be added to the Prouct CR

类型

`array`

.spec.chartVersions[]

描述

ChartVersion stands for chart version

类型

`object`

必填

`name` `version`

属性	类型	描述
`name`	`string`	Name stands for the chart's name
`version`	`string`	Version stands for the chart's version

.spec.commonConfigRef

描述

CommonConfigRef stands for the reference from CommonConfig to Product CR

类型

`array`

.spec.commonConfigRef[]

描述

KeysRef stands for key references with specified version

类型

`object`

必填

`keys` `versionMatch`

属性	类型	描述
`keys`	`array`	Keys stands for a set of keys
`versionMatch`	`string`	VersionMatch stands for the target version

.spec.commonConfigRef[].keys

描述

Keys stands for a set of keys

类型

`array`

.spec.commonConfigRef[].keys[]

描述

KeyRef stands for key reference

类型

`object`

必填

`refKey` `targetKey`

属性	类型	描述
`refKey`	`string`	RefKey stands for source key
`targetKey`	`string`	TargetKey stands for target key

.spec.description

描述

Description stands for the product's description

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.spec.displayName

描述

DisplayName stands for the product's display name

类型

`object`

属性	类型	描述
`en`	`string`	EN stands for text with English
`zh`	`string`	ZH stands for text with Simple Chinese

.spec.displayUIConfig

描述

DisplayUIConfig stands for the web UI type when showing product's detail information

类型

`object`

必填

`type`

属性	类型	描述
`scheme`	`object`	Scheme defines the dynamic form, used when Type is DynamicForm
`type`	`string`	Type defines the web UI type, [InUnderlord is DEPRECATED, use SubApp instead]

.spec.displayUIConfig.scheme

描述

Scheme defines the dynamic form, used when Type is DynamicForm

类型

`object`

.spec.installUIConfig

描述

InstallUIConfig stands for the web UI config when deploying product

类型

`object`

必填

`type`

属性	类型	描述
`scheme`	`object`	Scheme defines the dynamic form, used when Type is DynamicForm
`type`	`string`	Type defines the web UI type, [InUnderlord is DEPRECATED, use SubApp instead]

.spec.installUIConfig.scheme

描述

Scheme defines the dynamic form, used when Type is DynamicForm

类型

`object`

.spec.productResource

描述

ProductResource defines the target product CRD type

类型

`object`

必填

`group` `names` `version`

属性	类型	描述
`group`	`string`	Group stands for the target product's CRD Group

属性	类型	描述
`names`	`object`	Names stands for the target product's CRD's kind name
`version`	`string`	Version stands for the target product's CRD version

.spec.productResource.names

描述

Names stands for the target product's CRD's kind name

类型

`object`

必填

`kind` `plural`

属性	类型	描述
`categories`	`array`	categories is a list of grouped resources this custom resource belongs to (e.g. 'all'). This is published in API discovery documents, and used by clients to support invocations like `kubectl get all`.
`kind`	`string`	kind is the serialized kind of the resource. It is normally CamelCase and singular. Custom resource instances will use this value as the `kind` attribute in API calls.

属性	类型	描述
`listKind`	`string`	listKind is the serialized kind of the list for this resource. Defaults to "`kind`List".
`plural`	`string`	plural is the plural name of the resource to serve. The custom resources are served under `/apis/<group>/<version>/.../<plural>`. Must match the name of the CustomResourceDefinition (in the form `<names.plural>.<group>`). Must be all lowercase.
`shortNames`	`array`	shortNames are short names for the resource, exposed in API discovery documents, and used by clients to support invocations like `kubectl get <shortname>`. It must be all lowercase.
`singular`	`string`	singular is the singular name of the resource. It must be all lowercase. Defaults to lowercased `kind`.

.spec.productResource.names.categories

描述

categories is a list of grouped resources this custom resource belongs to (e.g. 'all'). This is published in API discovery documents, and used by clients to support invocations like `kubectl get all`.

类型

`array`

.spec.productResource.names.categories[]

类型

`string`

.spec.productResource.names.shortNames

描述

shortNames are short names for the resource, exposed in API discovery documents, and used by clients to support invocations like `kubectl get <shortname>`. It must be all lowercase.

类型

`array`

.spec.productResource.names.shortNames[]

类型

`string`

.spec.singleApp

描述

SingleAppConfig stands for the config for SignleApp type

类型

`object`

必填

`entrypoint`

属性	类型	描述
`entrypoint`	`string`	Entrypoint stands for the entrypoint of standalone app

.spec.subApp

描述

SubApp stands for the config for SubApp type

类型

`object`

必填

`entrypoint` `name`

属性	类型	描述
`entrypoint`	`string`	Entrypoint stands for the entrypoint of sub app
`name`	`string`	Name stands for the name of sub app

.spec.upgradeUIConfig

描述

UpgradeUIConfig stands for the web UI config when upgrading product

类型

`array`

.spec.upgradeUIConfig[]

描述

VersionedUIConfig stands for the web UI type with special version

类型

`object`

必填

`type` `versionMatch`

属性	类型	描述
`scheme`	`object`	Scheme defines the dynamic form, used when Type is DynamicForm
`type`	`string`	Type defines the web UI type, [InUnderlord is DEPRECATED, use SubApp instead]
`versionMatch`	`string`	VersionMatch stands for the target version

.spec.upgradeUIConfig[].scheme

描述

Scheme defines the dynamic form, used when Type is DynamicForm

类型

`object`

.spec.upgradeableVersions

描述

UpgradeableVersions stands for the versions can upgrade from

类型

`array`

.spec.upgradeableVersions[]

类型

`string`

.status

描述

ProductConfigStatus defines the observed state of ProductConfig

类型

`object`

必填

`message` `phase` `reason`

属性	类型	描述
`message`	`string`	Message contains the detail messages about the status
`phase`	`string`	Phase stands for the ProductConfig's status
`reason`	`string`	Reason explain why the ProductCongfig in this status

API 端点

可用的 API 端点如下：

- `/apis/alauda.io/v1alpha1/productconfigs`
  - `DELETE`: delete collection of ProductConfig
  - `GET`: list objects of kind ProductConfig
  - `POST`: create a new ProductConfig
- `/apis/alauda.io/v1alpha1/productconfigs/{name}`

- `DELETE` : delete the specified ProductConfig
- `GET` : read the specified ProductConfig
- `PATCH` : partially update the specified ProductConfig
- `PUT` : replace the specified ProductConfig

- `/apis/alauda.io/v1alpha1/productconfigs/{name}/status`
  - `GET` : read status of the specified ProductConfig
  - `PATCH` : partially update status of the specified ProductConfig
  - `PUT` : replace status of the specified ProductConfig

/apis/alauda.io/v1alpha1/productconfigs

HTTP 方法

`DELETE`

描述

delete collection of ProductConfig

HTTP 响应

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

HTTP 方法

`GET`

描述

list objects of kind ProductConfig

HTTP 响应

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

HTTP 状态码	响应体
401 - Unauthorized	Empty

HTTP 方法

`POST`

描述

create a new ProductConfig

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`	`ProductConfig` schema	`application/json` formatted

HTTP 响应

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

/apis/alauda.io/v1alpha1/productconfigs/{name}

HTTP 方法

`DELETE`

描述

delete the specified ProductConfig

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

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

HTTP 方法

`GET`

描述

read the specified ProductConfig

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update the specified ProductConfig

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

参数	类型	描述
		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	`ProductConfig` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace the specified ProductConfig

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

参数	类型	描述
		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`	`ProductConfig` schema	`application/json` formatted

HTTP 响应

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

/apis/alauda.io/v1alpha1/productconfigs/{name}/status

HTTP 方法

`GET`

描述

read status of the specified ProductConfig

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update status of the specified ProductConfig

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

参数	类型	描述
		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	`ProductConfig` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace status of the specified ProductConfig

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.

参数	类型	描述
		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`	`ProductConfig` schema	`application/json` formatted

HTTP 响应

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

ProductUpgrade [product.alauda.io/v1alpha1]

描述

ProductUpgrade is the Schema for the productupgrades API

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:

属性	类型	描述
		https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	ProductUpgradeSpec defines the desired state of ProductUpgrade
`status`	`object`	ProductUpgradeStatus defines the observed state of ProductUpgrade

.spec

描述

ProductUpgradeSpec defines the desired state of ProductUpgrade

类型

`object`

必填

`onlineUpgrade`

属性	类型	描述
`onlineUpgrade`	`object`	OnlineUpgrade is the online upgrade configuration

属性	类型	描述
`pause`	`boolean`	Pause is a flag to pause the upgrade
`targetArch`	`string`	TargetArch is the architecture of the product to upgrade to
`targetVersion`	`string`	TargetVersion is the version of the product to upgrade to
`type`	`string`	Type is the type of product upgrade
`upgrade`	`boolean`	Upgrade is a flag of to do the upgrade

.spec.onlineUpgrade

描述

OnlineUpgrade is the online upgrade configuration

类型

`object`

必填

`checkFrequency` `enable` `enableMsgNotify`

属性	类型	描述
`checkFrequency`	`integer`	CheckFrequency is the frequency to check for new version

属性	类型	描述
`enable`	`boolean`	Enable is a flag to enable online upgrade
`enableMsgNotify`	`boolean`	EnableMsgNotify is a flag to enable message notification

.status

描述

ProductUpgradeStatus defines the observed state of ProductUpgrade

类型

`object`

必填

`phase` `source` `status`

属性	类型	描述
`lastCheckTimestamp`	`string`	LastCheckTimestamp is the last time the product upgrade was checked
`message`	`string`	Message is the message of the product upgrade
`phase`	`string`	Phase is the phase of the product upgrade

属性	类型	描述
`preheatVersions`	`array`	PreHeatVersions is the pre-heat histories of the product upgrade
`source`	`object`	OnlineUpgradeSource is the source of online upgrade
`status`	`string`	Status is the status of the product upgrade
`upgradable`	`boolean`	Upgradable is a flag to indicate whether the product is upgradable
`upgradeBlockReason`	`string`	UpgradeBlockReason is the reason why the product is not upgradable
`upgradePaths`	`array`	UpgradePaths is the upgrade paths of the product upgrade

.status.preheatVersions

描述

PreHeatVersions is the pre-heat histories of the product upgrade

类型

`array`

.status.preheatVersions[]

描述

PreheatHistory defines the pre-heat history

类型

`object`

必填

`arch` `status` `version`

属性	类型	描述
`arch`	`string`	Arch is the architecture of the new version
`artifactsAll`	`integer`	ArtifactsAll is the number of artifacts
`artifactsSynced`	`integer`	ArtifactsSynced is the number of artifacts synced
`baseOperatorVer`	`string`	BaseOperatorVer is the version of the base operator
`preHeatEndTime`	`string`	PreHeatEndTime is the end time of pre-heat
`preHeatStartTime`	`string`	PreHeatStartTime is the start time of pre-heat
`progress`	`integer`	Progress is the progress of pre-heat

属性	类型	描述
`status`	`string`	PreHeatStatus is the pre-heat status
`version`	`string`	Version is the version of the product to upgrade to

.status.source

描述

OnlineUpgradeSource is the source of online upgrade

类型

`object`

必填

`registry`

属性	类型	描述
`envId`	`integer`	EnvId is the environment id
`insecure`	`boolean`	Insecure is a flag to enable insecure registry
`plainHttp`	`boolean`	PlainHttp is a flag to enable plain http
`registry`	`string`	Registry is the cloud registry

属性	类型	描述
`tenantId`	`integer`	TenantId is the tenant id

.status.upgradePaths

描述

UpgradePaths is the upgrade paths of the product upgrade

类型

`array`

.status.upgradePaths[]

描述

UpgradePath defines the upgrade path

类型

`object`

必填

`arches` `release` `sizes` `version`

属性	类型	描述
`arches`	`array`	Arches is the architectures of the new version
`release`	`string`	Release is the release version of the product to upgrade to
`sizes`	`array`	Sizes is the architectures of the new version

属性	类型	描述
`tags`	`array`	Tags is the tags of the new version
`version`	`string`	Version is the version of the product to upgrade to

.status.upgradePaths[].arches

描述

Arches is the architectures of the new version

类型

`array`

.status.upgradePaths[].arches[]

类型

`string`

.status.upgradePaths[].sizes

描述

Sizes is the architectures of the new version

类型

`array`

.status.upgradePaths[].sizes[]

类型

`integer`

.status.upgradePaths[].tags

描述

Tags is the tags of the new version

类型

`array`

.status.upgradePaths[].tags[]

类型

`string`

API 端点

可用的 API 端点如下：

- `/apis/product.alauda.io/v1alpha1/productupgrades`
  - `DELETE`: delete collection of ProductUpgrade
  - `GET`: list objects of kind ProductUpgrade
  - `POST`: create a new ProductUpgrade
- `/apis/product.alauda.io/v1alpha1/productupgrades/{name}`
  - `DELETE`: delete the specified ProductUpgrade
  - `GET`: read the specified ProductUpgrade
  - `PATCH`: partially update the specified ProductUpgrade
  - `PUT`: replace the specified ProductUpgrade
- `/apis/product.alauda.io/v1alpha1/productupgrades/{name}/status`
  - `GET`: read status of the specified ProductUpgrade
  - `PATCH`: partially update status of the specified ProductUpgrade
  - `PUT`: replace status of the specified ProductUpgrade

/apis/product.alauda.io/v1alpha1/productupgrades

HTTP 方法

`DELETE`

描述

delete collection of ProductUpgrade

HTTP 响应

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

HTTP 方法

`GET`

描述

list objects of kind ProductUpgrade

HTTP 响应

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

HTTP 方法

`POST`

描述

create a new ProductUpgrade

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`	`ProductUpgrade` schema	`application/json` formatted

HTTP 响应

HTTP 状态码	响应体
200 - OK	`ProductUpgrade` schema
201 - Created	`ProductUpgrade` schema

HTTP 状态码	响应体
202 - Accepted	`ProductUpgrade` schema
401 - Unauthorized	Empty

/apis/product.alauda.io/v1alpha1/productupgrades/{name}

HTTP 方法

`DELETE`

描述

delete the specified ProductUpgrade

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 ProductUpgrade

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update the specified ProductUpgrade

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	`ProductUpgrade` schema
401 - Unauthorized	Empty

HTTP 方法

`PUT`

描述

replace the specified ProductUpgrade

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`	`ProductUpgrade` schema	`application/json` formatted

HTTP 响应

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

/apis/product.alauda.io/v1alpha1/productupgrades/{name}/status

HTTP 方法

`GET`

描述

read status of the specified ProductUpgrade

HTTP 响应

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

HTTP 方法

`PATCH`

描述

partially update status of the specified ProductUpgrade

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.

HTTP 响应

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

HTTP 方法

`PUT`

描述

replace status of the specified ProductUpgrade

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

参数	类型	描述
		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`	`ProductUpgrade` schema	`application/json` formatted

HTTP 响应

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