

# Monitoring APIs

[Indicators \[monitoring.alauda.io/v1/indicators\]](#) [Metrics \[monitoring.alauda.io/v1/metrics\]](#) [Variables \[monitoring.alauda.io/v1/variables\]](#)

# Indicators [monitoring.alauda.io/v1beta1]

## /platform/monitoring.alauda.io/v1beta1/clusters/{cluster}/indicators

### **get** Get metrics indicators of a cluster

Retrieve metrics indicators for a specified cluster.

#### Parameters

- `cluster` (*in path*): `string` **required**  
Name of the cluster
- `kind` (*in query*): `string`  
Filter by indicator type
- `alertEnabled` (*in query*): `boolean`  
Filter indicators that support alerts
- `multipleEnabled` (*in query*): `boolean`  
Filter indicators that support multiple alert objects
- `dashboardEnabled` (*in query*): `boolean`  
Filter indicators that support custom dashboards

#### Response

- `200` `IndicatorList`: Successful response with list of metric indicators

# /platform/monitoring.alauda.io/v1beta1/projects/{project}/indicators

## get Get metrics indicators of a project

Retrieve metrics indicators for a specified project.

### Parameters

- `project` (*in path*): `string` `required`  
Name of the project
- `kind` (*in query*): `string`  
Filter by indicator type
- `alertEnabled` (*in query*): `boolean`  
Filter indicators that support alerts
- `multipleEnabled` (*in query*): `boolean`  
Filter indicators that support multiple alert objects
- `dashboardEnabled` (*in query*): `boolean`  
Filter indicators that support custom dashboards

### Response

- `200` `IndicatorList`: Successful response with list of metric indicators

# /platform/monitoring.alauda.io/v1beta1/projects/{project}/clusters/{cluster}/namespaces/{namespace}/indicators

## get Get metrics indicators of a namespace

Retrieve metrics indicators for a specified namespace.

## Parameters

- `project` (*in path*): `string` `required`  
Name of the project
- `cluster` (*in path*): `string` `required`  
Name of the cluster
- `namespace` (*in path*): `string` `required`  
Name of the namespace
- `kind` (*in query*): `string`  
Filter by indicator type
- `alertEnabled` (*in query*): `boolean`  
Filter indicators that support alerts
- `multipleEnabled` (*in query*): `boolean`  
Filter indicators that support multiple alert objects
- `dashboardEnabled` (*in query*): `boolean`  
Filter indicators that support custom dashboards

## Response

- `200` `IndicatorList`: Successful response with list of metric indicators

## IndicatorList

- `alertEnabled` : `boolean`  
Whether the indicator supports alerts
- `annotations` :
- `kind` : `string`  
Kind of the indicator
- `name` : `string`  
Name of the indicator
- `query` : `string`  
Query string for the indicator
- `type` : `string`

## Type of the indicator

- `unit` : `string`

## Unit of the indicator

- `legend` : `string`

## Legend of the indicator

- `variables` : `[]string`

## Variables used in the query

# Metrics [monitoring.alauda.io/v1beta1]

## /platform/monitoring.alauda.io/v1beta1/clusters/{cluster}/metrics

### **get** Get monitoring metrics of a cluster

Retrieves monitoring metrics for a specified cluster.

#### Parameters

- cluster** (in path): `string` **required**  
Name of the cluster
- start** (in query): `integer`  
Start time as a 10-digit timestamp
- end** (in query): `integer`  
End time as a 10-digit timestamp
- step** (in query): `integer`  
Step interval
- time** (in query): `integer`  
Timestamp, defaults to current time. Mutually exclusive with start/end/step.
- timeout** (in query): `integer`  
Timeout in seconds, default is 10s
- indicators** (in query): `string`  
Metrics to query, multiple indicators separated by '|'
- expr** (in query): `string`

Native query expression, mutually exclusive with indicators

- `variables` (*in query*): `string`  
Variables for the query, formatted as 'A=v1,B=v2,C=v3'

## Response

- `200` `MetricList`: Successful response with list of monitoring metrics
- `400` `Status`: Monitoring plugin not deployed
- `503` `Status`: Monitoring plugin unavailable

# /platform/monitoring.alauda.io/v1beta1/projects/{project}/metrics

## `get` Get monitoring metrics of a project

Retrieves monitoring metrics for a specified cluster.

## Parameters

- `project` (*in path*): `string` `required`  
Name of the project
- `start` (*in query*): `integer`  
Start time as a 10-digit timestamp
- `end` (*in query*): `integer`  
End time as a 10-digit timestamp
- `step` (*in query*): `integer`  
Step interval
- `time` (*in query*): `integer`  
Timestamp, defaults to current time. Mutually exclusive with start/end/step.
- `timeout` (*in query*): `integer`  
Timeout in seconds, default is 10s
- `indicators` (*in query*): `string`

Metrics to query, multiple indicators separated by '|'

- `expr` (*in query*): `string`  
Native query expression, mutually exclusive with indicators
- `variables` (*in query*): `string`  
Variables for the query, formatted as 'A=v1,B=v2,C=v3'

## Response

- `200` **MetricList**: Successful response with list of monitoring metrics
- `400` **Status**: Monitoring plugin not deployed
- `503` **Status**: Monitoring plugin unavailable

# /platform/monitoring.alauda.io/v1beta1/projects/{project}/clusters/{cluster}/namespaces/{namespace}/metrics

## `get` Get monitoring metrics of a namespace

Retrieves monitoring metrics for a specified cluster.

## Parameters

- `project` (*in path*): `string` **required**  
Name of the project
- `cluster` (*in path*): `string` **required**  
Name of the cluster
- `namespace` (*in path*): `string` **required**  
Name of the namespace
- `start` (*in query*): `integer`  
Start time as a 10-digit timestamp
- `end` (*in query*): `integer`  
End time as a 10-digit timestamp

- `step` (*in query*): `integer`  
Step interval
- `time` (*in query*): `integer`  
Timestamp, defaults to current time. Mutually exclusive with start/end/step.
- `timeout` (*in query*): `integer`  
Timeout in seconds, default is 10s
- `indicators` (*in query*): `string`  
Metrics to query, multiple indicators separated by '|'
- `expr` (*in query*): `string`  
Native query expression, mutually exclusive with indicators
- `variables` (*in query*): `string`  
Variables for the query, formatted as 'A=v1,B=v2,C=v3'

## Response

- `200` `MetricList`: Successful response with list of monitoring metrics
- `400` `Status`: Monitoring plugin not deployed
- `503` `Status`: Monitoring plugin unavailable

## MetricList

- `indicator` : `string`  
The indicator name
- `query` : `string`  
The native query expression
- `result` : `[]MetricResult`

## MetricResult

- `metric` : `map[string]string`
- `values` : `[]string`

# Status

- `kind` : `string`
- `apiVersion` : `string`
- `metadata` : `object`
- `status` : `string`
- `message` : `string`
- `reason` : `string`
- `code` : `integer`

# Variables [monitoring.alauda.io/v1beta1]

## /platform/monitoring.alauda.io/v1beta1/clusters/{cluster}/variables

### **get** Get variable candidates for monitoring panels in a cluster

Fetches the variable candidates for monitoring panels in the specified cluster. The API permissions are the same as for metrics.

#### Parameters

- `cluster` (in path): `string` **required**  
The name of the cluster.
- `expr` (in query): `string`  
The query expression to fetch variable values.
- `regex` (in query): `string`  
The regular expression to extract variable values.
- `start` (in query): `integer`  
Start time as a 10-digit timestamp
- `end` (in query): `integer`  
End time as a 10-digit timestamp

#### Response

- `200` **Option**: Successful response with variable candidates.

- `400` **Status:** Invalid expr
- `503` **Status:** Monitoring plugin unavailable

## /platform/monitoring.alauda.io/v1beta1/projects/{project}/variables

### `get` Get variable candidates for monitoring panels in a project

Fetches the variable candidates for monitoring panels in the specified cluster. The API permissions are the same as for metrics.

#### Parameters

- `project` (*in path*): `string` **required**  
The name of the project.
- `expr` (*in query*): `string`  
The query expression to fetch variable values.
- `regex` (*in query*): `string`  
The regular expression to extract variable values.
- `start` (*in query*): `integer`  
Start time as a 10-digit timestamp
- `end` (*in query*): `integer`  
End time as a 10-digit timestamp

#### Response

- `200` **Option:** Successful response with variable candidates.
- `400` **Status:** Invalid expr
- `503` **Status:** Monitoring plugin unavailable

# /platform/monitoring.alauda.io/v1beta1/projects/{project}/clusters/{cluster}/namespaces/{namespace}/variables

## **get** Get variable candidates for monitoring panels in a namespace

Fetches the variable candidates for monitoring panels in the specified cluster. The API permissions are the same as for metrics.

### Parameters

- `project` (*in path*): `string` **required**  
The name of the project.
- `cluster` (*in path*): `string` **required**  
The name of the cluster.
- `namespace` (*in path*): `string` **required**  
The name of the namespace.
- `expr` (*in query*): `string`  
The query expression to fetch variable values.
- `regex` (*in query*): `string`  
The regular expression to extract variable values.
- `start` (*in query*): `integer`  
Start time as a 10-digit timestamp
- `end` (*in query*): `integer`  
End time as a 10-digit timestamp

### Response

- `200` **Option**: Successful response with variable candidates.
- `400` **Status**: Invalid expr
- `503` **Status**: Monitoring plugin unavailable

## Option

- `options` : `[]string`

## Status

- `kind` : `string`
- `apiVersion` : `string`
- `metadata` : `object`
- `status` : `string`
- `message` : `string`
- `reason` : `string`
- `code` : `integer`