

Kafka Cluster Types in Confluent Cloud

Confluent offers different types of Apache Kafka® clusters in Confluent Cloud. The cluster type you choose determines the features, capabilities, and price of the cluster. Use the information in this topic to find the cluster with the features and capabilities that best meets your needs.

! Ready to try the right Kafka cluster for your workload?

Sign up for Confluent Cloud [↗](#) and get \$400 in free credits to test different cluster types—Basic, Standard, Enterprise, and Freight—based on your performance and networking needs.

Cluster types

Confluent Cloud offers these Kafka cluster types:

- Basic clusters - Used for experimentation, early development and basic use cases.
- Standard clusters - Used for production-ready features and functionality.
- Enterprise clusters - Used for production-ready functionality that requires private networking capabilities.
- Dedicated clusters - Used for critical production workloads with high traffic or private networking requirements.
- Freight clusters - Used for high-throughput, relaxed latency workloads that are less expensive than self-managed open source Kafka.

► Freight cluster considerations

Cluster provisioning and scaling

Confluent uses billing units to provision and scale clusters.

Elastic scaling

Basic, Standard, Enterprise, and Freight clusters are elastic, shrinking and expanding automatically based on load. You don't resize these clusters (unlike Dedicated clusters). When you need more capacity, your cluster expands up to the fixed ceiling. If you're not using capacity above the minimum, you're not paying for it. If you're at zero capacity, you don't pay for anything. For more information, see [Elastic Confluent Unit for Kafka](#) ([../billing/overview.html#e-cku-definition](#)) and [eCKU/CKU comparison](#).

► Freight clusters scaling considerations

Manual scaling

Dedicated clusters are provisioned and billed in terms of Confluent Unit for Kafka (CKU). CKUs are a unit of horizontal scalability in Confluent Cloud that provide a preallocated amount of resources. How much you can ingest and stream per CKU depends on a variety of factors including client application design and partitioning strategy. For more information, see [Monitor Dedicated Clusters in Confluent Cloud](#) ([../monitoring/cluster-load-metric.html#monitor-dedicated-clusters](#)) and [Dedicated Cluster Performance and Expansion in Confluent Cloud](#) ([../monitoring/monitor-performance.html#cloud-cluster-monitor-performance](#)).

Features

All clusters have the following features:

- [Kafka ACLs](#) ([../security/access-control/acls/overview.html#acl-manage](#))
- [Fully-managed replica placement](#) ([resilience.html#confluent-cloud-resilience](#))
- [User interface to manage consumer lag](#) ([../monitoring/monitor-lag.html#cloud-monitoring-lag](#))
- [Topic management](#) ([../topics/overview.html#cloud-topics-manage](#))
- [Fully-Managed Connectors](#) ([../connectors/overview.html#kafka-connect-cloud](#))
- [View and consume Connect logs](#) ([../connectors/logging-cloud-connectors.html#ccloud-connector-logging](#))
- [Stream Governance](#) ([../stream-governance/index.html#cloud-dg](#))
- [Stream Catalog](#) ([../stream-governance/stream-catalog.html#cloud-stream-catalog](#))
- [Stream Lineage](#) ([../stream-governance/stream-lineage.html#cloud-stream-lineage](#))

- Encryption-at-rest [↗](#)
- TLS for data in transit ([../security/encrypt/tls.html#manage-data-in-transit-with-tls](#))
- Role-based Access Control (RBAC) ([../security/access-control/rbac/overview.html#cloud-rbac](#))
(Basic clusters do not support RBAC roles for resources within the Kafka cluster)

Feature comparison table

The tables below offer comparisons of the features supported by only some Kafka cluster types.

Feature	<u>Basic</u>	<u>Standard</u>	<u>Enterprise</u>	<u>Dedicated</u>	<u>Freight</u>
Exactly Once Semantics (/platform/current/streams/concepts.html#streams-concepts-processing-guarantees)	Yes	Yes	Yes	Yes	No
Key based compacted storage (/platform/current/kafka/design.html#log-compaction)	Yes	Yes	Yes	Yes	No
Custom Connectors (../connectors/bring-your-connector/overview.html#cc-bring-your-connector)	Yes	Yes	No	Yes	No
Flink (../flink/overview.html#ccloud-flink)	Yes	Yes	Yes	Yes	No
ksqlDB (../ksqldb/overview.html#cloud-ksqldb-create-stream-processing-apps)	Yes	Yes	No	Yes	No
Public networking (../networking/overview.html#cloud-networking-support-public)	Yes	Yes	No	Yes	No
Private networking (../networking/overview.html#cloud-networking-support-public)	No	No	Yes	Yes	Yes
OAuth (../security/authenticate/workload-identities/identity-providers/oauth/overview.html#oauth-overview)	No	Yes	Yes	Yes	Yes
Audit logs (../monitoring/audit-logging/cloud-audit-log-concepts.html#cloud-audit-logs)	No	Yes	Yes	Yes	Yes

Feature	Basic	Standard	Enterprise	Dedicated	Freight
Self-managed encryption keys (../security/encrypt/byok/overview.html#byok-encrypted-clusters)	No	No	Yes	Yes	No
Automatic Elastic scaling (../billing/overview.html#e-cku-definition)	Yes	Yes	Yes	No	Yes
Stream Sharing (../stream-sharing/index.html#cloud-data-sharing)	Yes	Yes	No	Yes but all private networking options are not supported	No
Client Quotas (client-quotas.html#client-quotas)	No	No	No	Yes	No
Access Transparency (../monitoring/audit-logging/access-transparency-overview.html#access-transparency-overview)	No	No	No	Yes	No


Cluster linking capabilities

The table below offers a comparison of cluster linking capabilities by cluster type.

Cluster type	Basic	Standard	Enterprise	Dedicated	Freight
Supports source clusters	Yes	Yes	Yes (*)	Yes (*)	No
Supports destination clusters	No	No	Yes (*)	Yes (*)	No

* Capability dependent on the networking type and the other cluster involved. To learn more, see [Supported cluster types \(../multi-cloud/cluster-linking/index.html#cloud-cluster-linking-supported-types\)](#) in the Cluster Linking documentation.

Uptime service level agreement options

The table below offers a comparison of uptime service level agreements (SLA) options by cluster type. For more information, see [Confluent Cloud Service Level Agreement](#) .

Considerations:

- To obtain a higher uptime SLA, you can upgrade from Basic to a Standard cluster at any time using the Cloud Console.
- Standard and Enterprise clusters require 2 eCKU minimums for the 99.99% SLA.
- Dedicated clusters require Multi-Zone deployments for the 99.99% SLA.

Cluster type	99.5%	99.9%	99.95%	99.99%
Basic	Yes	No	No	No
Standard	No	Yes	No	Yes (Requires 2 eCKU)
Enterprise	No	Yes	No	Yes (Requires 2 eCKU)
Dedicated	No	No	Yes (SZ)	Yes (MZ)
Freight	No	No	No	Yes

eCKU/CKU comparison

Use the table below to compare limits for a single billing unit for each cluster type. For more information, see Elastic Confluent Unit for Kafka ([../billing/overview.html#e-cku-definition](https://docs.confluent.io/cloud/current/billing/overview.html#e-cku-definition)), Minimum/maximum eCKU requirements and Cluster provisioning and scaling.

Dimension	<u>Basic</u> eCKU	<u>Standard</u> eCKU	<u>Enterprise</u> eCKU	<u>Dedicated</u> CKU	<u>Freight</u> eCKU
Ingress (../_glossary.html#term-ingress) (Mbps)	5	25	60	60	60
Egress (../_glossary.html#term-egress) (Mbps)	15	75	180	180	180

Dimension	<u>Basic</u> eCKU	<u>Standard</u> eCKU	<u>Enterprise</u> eCKU	<u>Dedicated</u> CKU	<u>Freight</u> eCKU
Partitions (pre-replication) (../_glossary.html#term-partitions-pre-replication)	30	250	3,000	4,500	3,000
Number of partitions that you can compact (pre-replication)	30	250	360	4,500	None
Total client connections (../_glossary.html#term-total-client-connections)	20	1000	4,500	18,000	18,000
Connection attempts (../_glossary.html#term-connection-attempts) (per second)	5	80	250	500	500
Requests (../_glossary.html#term-requests) (per second)	100	1,500	7,500	15,000	15,000
Kafka REST Produce v3 - Max throughput (MBps):	N/a	N/a	N/a	50	N/a
Kafka REST Produce v3 - Max connection requests (per second):	N/a	N/a	N/a	300	N/a
Kafka REST Produce v3 - Max streamed requests (per second):	N/a	N/a	N/a	3000	N/a
Kafka REST Admin v3 - Max connection requests (per second):	N/a	N/a	N/a	300	N/a

Minimum/maximum eCKU requirements

The table below lists minimum and maximum requirements for elastic cluster types. Dedicated clusters have CKU limits that depend on billing and other factors. For more information, see CKU purchase limits.

❗ Limited Availability of 32 eCKU maximum Enterprise clusters

Enterprise clusters that scale to 32 eCKU maximum (current limit is 10) are in Limited Availability and available by request. To sign up, contact Confluent [↗](#).

Cluster type SKU	Minimum eCKU	Maximum eCKU
Basic	1	50
Standard	1 (99.9% SLA), 2 (99.99% SLA)	10
Enterprise	1 (99.9% SLA), 2 (99.99% SLA)	10 (current maximum) / 32 (Limited Availability)
Freight	2	152

CKU purchase limits

Dedicated clusters can be purchased in any whole number of CKUs up to the limit.

- For organizations with credit card billing, the upper limit is 4 CKUs per Dedicated cluster. Clusters up to 152 * CKUs are available by request.
- For organizations with integrated cloud provider billing or payment using an invoice, the upper limit is 24 CKUs per Dedicated cluster. Clusters up to 152 * CKUs are available by request.

For clusters that can scale to 152 * CKU, contact Confluent Support [↗](#) to discuss the onboarding process and product considerations.

Single-zone clusters can have 1 or more CKUs, whereas multi-zone clusters, which are spread across three availability zones, require a minimum of 2 CKUs. Zone availability cannot be changed after the cluster is created.

* AWS and Google Cloud support Kafka clusters to 152 CKUs. Azure supports Kafka clusters to 100 CKUs.

Fixed limits and recommended guidelines

CKUs determine the capacity of your cluster. For a Confluent Cloud cluster, the expected performance for any given workload is dependent on a variety of dimensions, such as message size and number of

partitions.

There are two categories of CKU dimensions:

- Dimensions with a fixed limit that cannot be exceeded.
- Dimensions with a more flexible guideline that may be exceeded depending on the overall cluster load.

The recommended guideline for a dimension is calculated for a workload optimized across the dimensions, enabling high levels of CKU utilization as measured by the cluster load metric ([../monitoring/cluster-load-metric.html#cloud-cluster-load-metric](#)). You may exceed the recommended guideline for a dimension, and achieve higher performance for that dimension, usually only if your usage of other dimensions is less than the recommended guideline or fixed limit.

Also note that usage patterns across all dimensions affect the workload and you may not achieve the suggested guideline for a particular dimension. For example, if you reach the partition limit, you will not likely reach the maximum CKU throughput guideline.

You should monitor the cluster load metric ([../monitoring/cluster-load-metric.html#cloud-cluster-load-metric](#)) for your cluster to see how your usage pattern correlates with cluster utilization.

When a cluster's load metric is high, the cluster may delay new connections and/or throttle clients in an attempt to ensure the cluster remains available. This throttling would register as non-zero values for the producer client `produce-throttle-time-max` and `produce-throttle-time-avg` metrics ([../client-apps/monitoring.html#producer-throttling](#)) and consumer client `fetch-throttle-time-max` and `fetch-throttle-time-avg` metrics ([/platform/current/kafka/monitoring.html#fetch-metrics-kafka](#)).

Dimensions with fixed limits

The following dimensions have fixed limits that you cannot exceed:

- Storage (pre-replication)
- Partitions (pre-replication)
- Connection attempts
- Kafka REST Produce v3
- Kafka REST Admin v3

Dimensions with recommended guidelines


These dimensions provide guidelines for capacity planning. The ability to fully utilize these dimensions depend on the workload and utilization of other dimensions. See more about measuring load in cluster load metric ([../monitoring/cluster-load-metric.html#cloud-cluster-load-metric](#)) and for the maximum bandwidth for each cloud provider (AWS, Google Cloud, Azure), are available in [Benchmark Your Dedicated Apache Kafka Cluster on Confluent Cloud](#).

The following dimensions come with recommended guidelines:

- Ingress
- Egress
- Total client connections
- Requests

Cluster limit comparison

Use the table below to compare cluster limits across cluster types.


Limited Availability of 32 eCKU maximum Enterprise clusters

Enterprise clusters that scale to 32 eCKU maximum (current limit is 10) are in Limited Availability and available by request. To sign up, contact Confluent.

For Enterprise clusters, the following table shows the current maximum (10 eCKU). If you’re participating in the 32 eCKU Limited Availability for Enterprise clusters, your cluster limits are higher.

Dimension	Basic	Standard	Enterprise	Dedicated	Freight
-----------	-------	----------	------------	-----------	---------

Dimension	Basic	Standard	Enterprise	Dedicated	Freight
<u>Maximum eCKU/CKU</u>	50	10	10 (current maximum) / 32 (Limited Availability)	152 ⁺	152
Ingress (MBps) ^{* +}	250	250	600	9,120	9,120
Egress (MBps) ^{* +}	750	750	1800	27,360	27,360
Partitions (pre-replication) ^{* +}	1500	2500	30,000	100,000	50,000
Number of partitions you can compact ^{* +}	1500	2500	3,600	100,000	None
Total client connections ^{* +}	1000	10,000	45,000	2,736,000	2,736,000
Connection attempts (per second) ^{* +}	80	800	2500	76,000	76,000
Requests (per second) ^{* +}	15,000	15,000	75,000	2,280,000	2,280,000
Message size (MB)	8	8	20	20	20
Client version (minimum)	0.11.0	0.11.0	0.11.0	0.11.0	0.11.0
Request size (MB)	100	100	100	100	100
Fetch bytes (MB)	55	55	55	55	55
API keys	50	100	500	2,000	500
Partition creation and deletion (per five minute period)	250	500	500	5,000	500
Connector tasks per Kafka cluster	250 ^{* +}	250	250	250	N/A
ACLs	1,000	1,000	4,000	10,000	10,000
Kafka REST Produce v3 - Max throughput (MBps):	10	10	10	7,600 ⁺	10
Kafka REST Produce v3 - Max connection requests (per second):	25	25	25	45,600 ⁺	25
Kafka REST Produce v3 - Max streamed requests (per second):	1000	1000	1000	456,000 ⁺	1000

Dimension	Basic	Standard	Enterprise	Dedicated	Freight
Kafka REST Produce v3 - Max message size for Kafka REST Produce API (MB):	8	8	8	20	8
Kafka REST Admin v3 - Max connection requests (per second):	25	25	25	45,600 [†]	25


* Limit based on Elastic Confluent Unit for Kafka (eCKU). You only pay for the capacity you use up to the limit. For more information, see Elastic Confluent Unit for Kafka ([../billing/overview.html#e-cku-definition](https://docs.confluent.io/cloud/current/billing/overview.html#e-cku-definition)).

† Limit based on a Dedicated Kafka cluster with 152 CKU. For more information, see CKU purchase limits and Confluent Unit for Kafka ([../billing/overview.html#cku-definition](https://docs.confluent.io/cloud/current/billing/overview.html#cku-definition)).

*† Basic clusters are limited to one task per connector. You can deploy 250 connectors to a Basic cluster but each connector can only have one task. If you need more than one task, upgrade your cluster.

The capabilities provided in this topic are for planning purposes, and are not a guarantee of performance, which varies depending on each unique configuration.

Migrate from open source Kafka

If you are currently self-managing Kafka, use the following information to help choose which cluster type best suits your use-cases. For information about migrating from open source Kafka to Confluent Cloud, see the [Migrating from Kafka services to Confluent](#)  PDF.

- Ingress: Use producer outgoing-byte-rate metrics (</platform/current/kafka/monitoring.html#producer-global-request-metrics>) and broker `kafka.server:type=BrokerTopicMetrics,name=BytesInPerSec` metrics (</platform/current/kafka/monitoring.html#kafka-monitoring-metrics-broker>) to understand your throughput.
- Egress: the consumer incoming-byte-rate metrics (</platform/current/kafka/monitoring.html#consumer-global-request-metrics>) and broker `kafka.server:type=BrokerTopicMetrics,name=BytesOutPerSec` (</platform/current/kafka/monitoring.html#kafka-monitoring-metrics-broker>) to understand your throughput.
- Storage (pre-replication): Use the amount of disk space your cluster is using to understand your storage needs.

- Partitions (pre-replication): Use the `kafka.controller:type=KafkaController,name=GlobalPartitionCount` metric to understand your partition usage. Find details in the Broker section (</platform/current/kafka/monitoring.html#kafka-monitoring-metrics-broker>).
- Total client connections: Use the broker `kafka.server:type=socket-server-metrics,listener={listener_name},networkProcessor={#},name=connection-count` metrics (</platform/current/kafka/monitoring.html#kstypesocketservermetrics>) to understand how many connections you are using. This value may not have a 1:1 ratio to connections in Confluent Cloud, depending on the number of brokers, partitions, and applications in your self-managed cluster.
- Connection attempts: Use the rate of change for the `kafka.server:type=socket-server-metrics,listener={listener_name},networkProcessor={#},name=connection-count` (</platform/current/kafka/monitoring.html#zk-metrics>) metric and the Consumer `connection-creation-rate` metric to understand how many new connections you are creating. For details, see Broker Metrics (</platform/current/kafka/monitoring.html#kstypesocketservermetrics>) and Global Connection Metrics (</platform/current/kafka/monitoring.html#consumer-metric-global-connection-metrics>).
- Requests: Use the broker `kafka.network:type=RequestMetrics,name=RequestsPerSec,request={Produce FetchConsumer FetchFollower}` metrics (</platform/current/kafka/monitoring.html#kafka-monitoring-metrics-broker>) and client request-rate metrics (</platform/current/kafka/monitoring.html#kafka-monitoring-metrics-producer>) to understand your request volume.

Partition guidelines

The partition capabilities that follow are based on benchmarking and intended as practical guidelines for planning purposes. Performance per partition will vary depending on your individual configuration, and these benchmarks do not guarantee performance.

Except for Basic clusters, all clusters offer:

- Unlimited storage per partition
- Unlimited storage per partition for compacted topics



Basic clusters offer:

- 5 TB per partition
- 5 TB per partition for compacted topics

Use the table below to compare partition limits across cluster types.

Dimension	Basic	Standard	Enterprise	Dedicated	Freight
Ingress per partition	~5 MBps	~5 MBps	~6 MBps	~4 MBps (aggregate producer throughput)	~6 MBps
Egress per partition	~15 MBps	~15 MBps	~18 MBps	~12 MBps (aggregate producer throughput)	~18 MBps

Related content

- For per-topic settings and limits, see [Manage Kafka Cluster Configuration Settings in Confluent Cloud \(broker-config.html#cloud-broker-config\)](#)
- For quotas that apply to organizations, environments, clusters, and accounts, see [Service Quotas for Confluent Cloud \(../quotas/service-quotas.html#ccloud-resource-limits\)](#)
- For performance monitoring your clusters, see [Metrics API \(../monitoring/metrics-api.html#metrics-api\)](#)
- For costs by dimension, see [Billing dimensions \(../billing/overview.html#billing-dimensions\)](#)
- For cost estimates, see [Confluent Cost Estimator](#) 
- For information about migrating from open source Kafka to Confluent Cloud, see the [Migrating from Kafka services to Confluent](#)  PDF