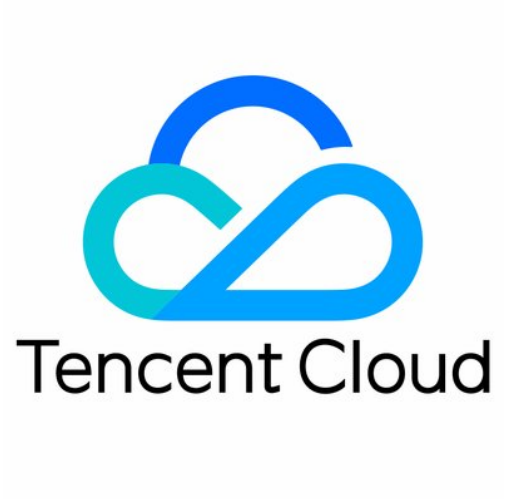
Message Queue CKafka

Product Introduction

Product Documentation



©2013-2022 Tencent Cloud. All rights reserved.

Copyright Notice

©2013-2022 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice

Tencent Cloud

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

©2013-2022 Tencent Cloud. All rights reserved.

Product Introduction
Overview

Last updated：2022-06-02 16:43:57

CKafka Overview

Based on the open-source Apache Kafka message queuing engine, Tencent Cloud Kafka (CKafka) provides high-throughput and highly scalable message queuing services. It is perfectly compatible with the APIs of Apache Kafka v0.9, v0.10, v1.1, v2.4 and v2.8 and has greater advantages in terms of performance, scalability, business security, and Ops, allowing you to enjoy powerful features at low costs while eliminating tedious Ops work.

Features

- **Message Decoupling**
  CKafka effectively decouples the relationship between message producer and consumer, allowing you to independently scale or modify the production/consumption processing procedure as long as they follow the same API constraints.

- **Peak Shifting**
  CKafka can withstand access traffic surges instead of completely crashing due to sudden overwhelming requests, which effectively boosts system robustness.

- **Sequential Read/Write**
  CKafka can guarantee the order of messages in a partition. Just like most message queue services, it can also ensure that data is processed in order, greatly improving disk efficiency.

- **Async Communication**
  In the scenario where the business does not need to process messages immediately, CKafka provides an async message processing mechanism, that is, when the traffic is high, messages will be put into the queue only and processed after the traffic drops, which significantly relieves the system pressure.

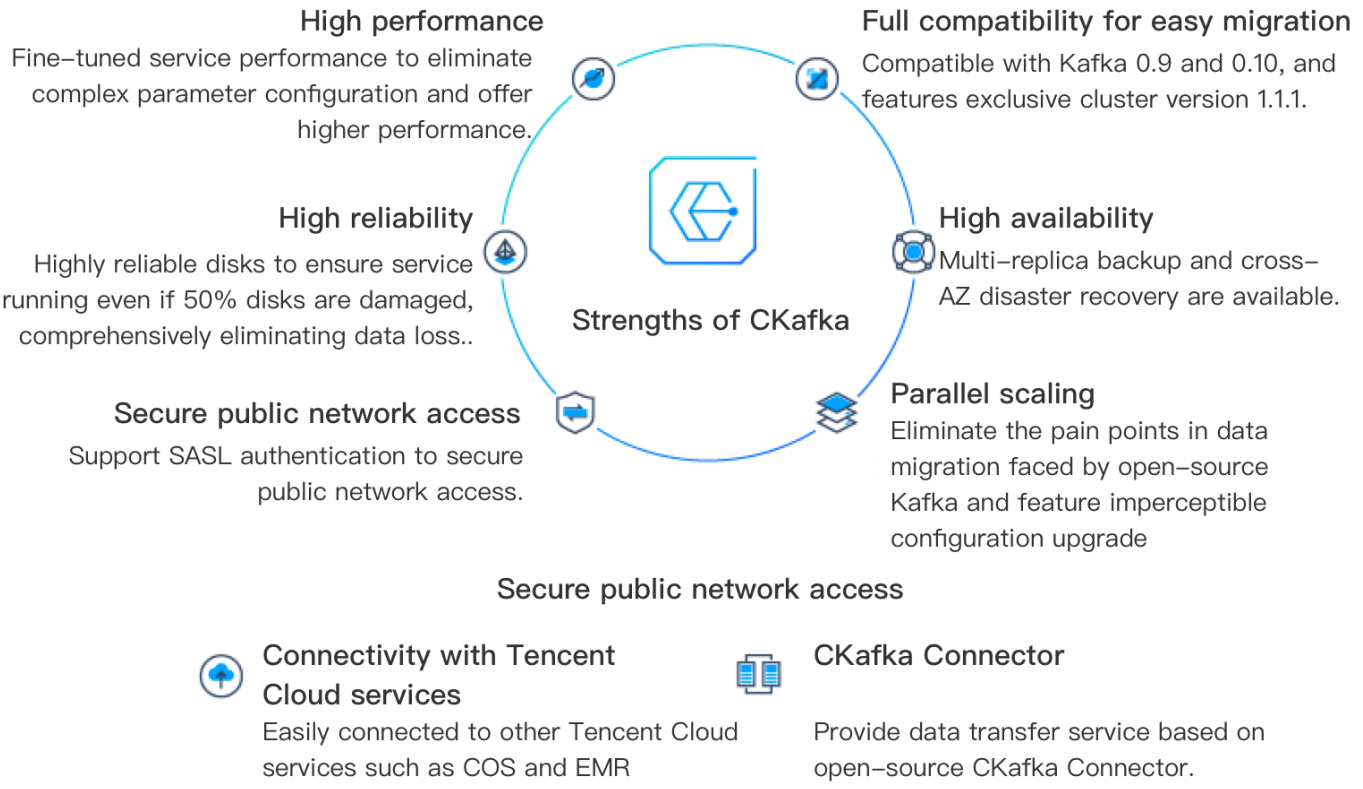
Note：
CKafka supports private deployment. You can submit a ticket for consultation.

©2013-2022 Tencent Cloud. All rights reserved.

Strengths

Last updated：2022-06-02 16:45:07

This document describes the strengths of CKafka compared to Apache Kafka.



Strengths

Full compatibility with Apache Kafka and easy migration

- CKafka is compatible with Kafka 0.9, 0.10, 1.1 and 2.4.
- CKafka is based on the existing code of the open-source Apace Kafka ecosystem. Without any changes to your existing project, you can migrate data to the cloud and enjoy the high-performance message queue services provided by Tencent Cloud Kafka.

High performance

- The message queue team at Tencent Cloud has optimized CKafka to allow you to enjoy higher-performance services without having to go through a complicated process of configuration.

©2013-2022 Tencent Cloud. All rights reserved.

- CKafka supports upgrade and downgrade via UI operations and is backed by a powerful IaaS layer, delivering 10% to 20% higher production capabilities than Apache Kafka.

High availability

- Leveraging Tencent's years of experience in monitoring technologies, CKafka offers comprehensive monitoring on clusters and has a professional Ops team in place that responds to alarms on a 24/7 basis to ensure high availability.
- Custom multi-AZ deployments in the same region is supported to improve disaster recovery ability.

High reliability

- CKafka uses highly reliable disks and can keep its services running even if 50% of the disks are unavailable.
- 2 replicas are created by default, and up to 3 replicas can be used. The more replicas, the higher the reliability.

Parallel scaling

- The backend system automatically scales resources when cluster traffic and disk usage hit the alarm thresholds. The client side is unaffected during the process.
- CKafka solves the problems in data migration with Apache Kafka and features smooth upgrade with no influence on your business.

Data security

CKafka provides security capabilities including authentication, authorization, and root account/sub-account, offering enterprise-level security services.

- VPC: supports access from Tencent Cloud VPC, enhancing network security.
- SASL authentication: allows safer public network access.
- Root account/sub-account: supports CAM features including root account/sub-account and collaborator, enabling authorization between root and sub-accounts as well as across organizational accounts.

Connectivity with Related Services

Integration with cloud services

CKafka can easily integrate with other Tencent Cloud services such as COS and EMR.

Kafka Connector

CKafka supports data transfer based on open-source Kafka Connector, enabling data transfer between Kafka clusters.

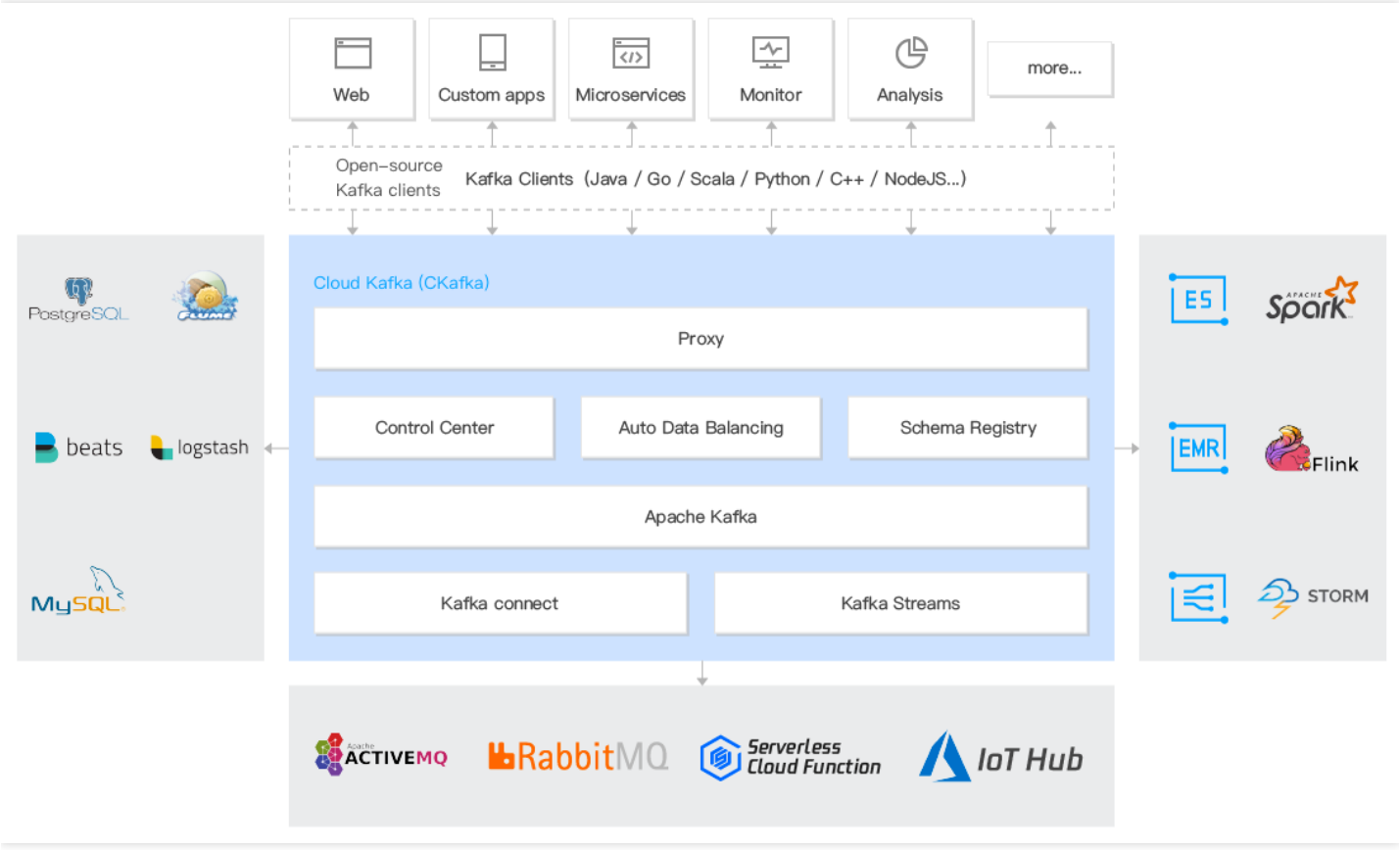
©2013-2022 Tencent Cloud. All rights reserved.

Use Cases

Last updated：2021-03-25 11:34:30

CKafka is widely used in big data scenarios, such as webpage tracking, behavior analysis, log aggregation, monitoring, streaming data processing, and online and offline data analysis.
You can make data integration easier by:

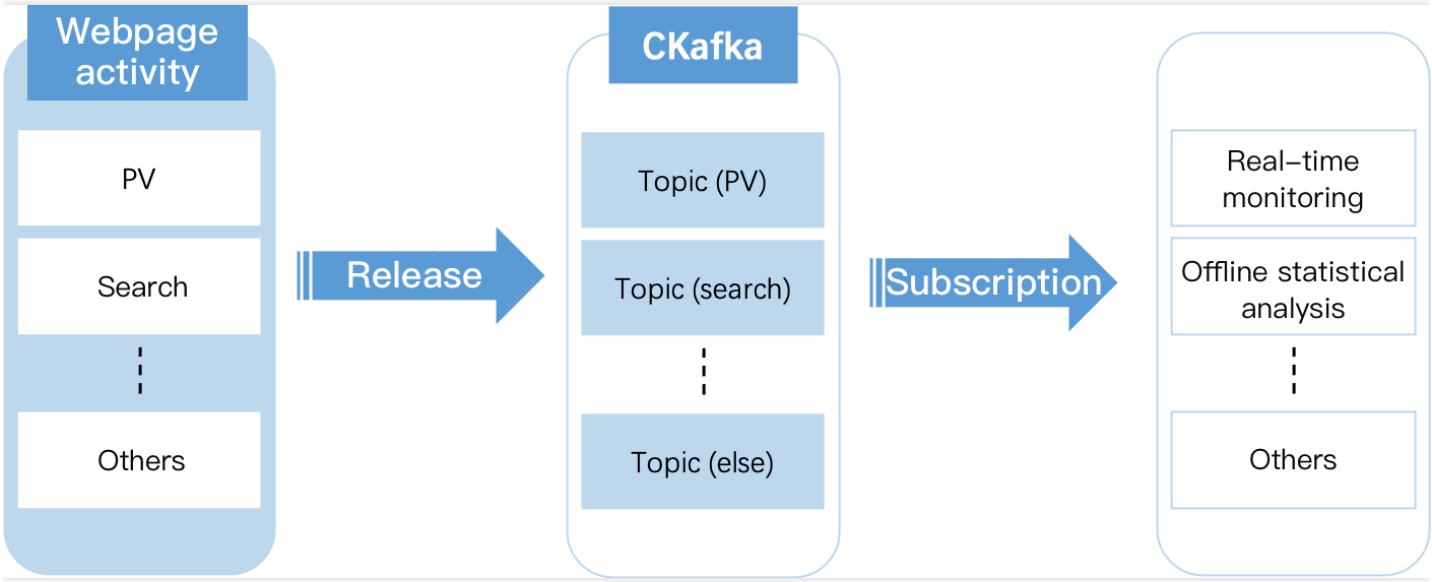
- Importing the messages in CKafka to data warehouses in Tencent Cloud such as COS and Oceanus.
- Connecting to other Tencent Cloud products via SCF triggers.



Webpage Tracking

CKafka processes website activities (such as PV, search, and other user behaviors) in real time and then publishes them to topics by type. These information flows can be used for real-time monitoring or offline statistical analysis. Since a large amount of activity information is generated in each user's page views, website activity tracking requires a very high throughput. CKafka can perfectly meet the requirements of high throughput and offline processing.

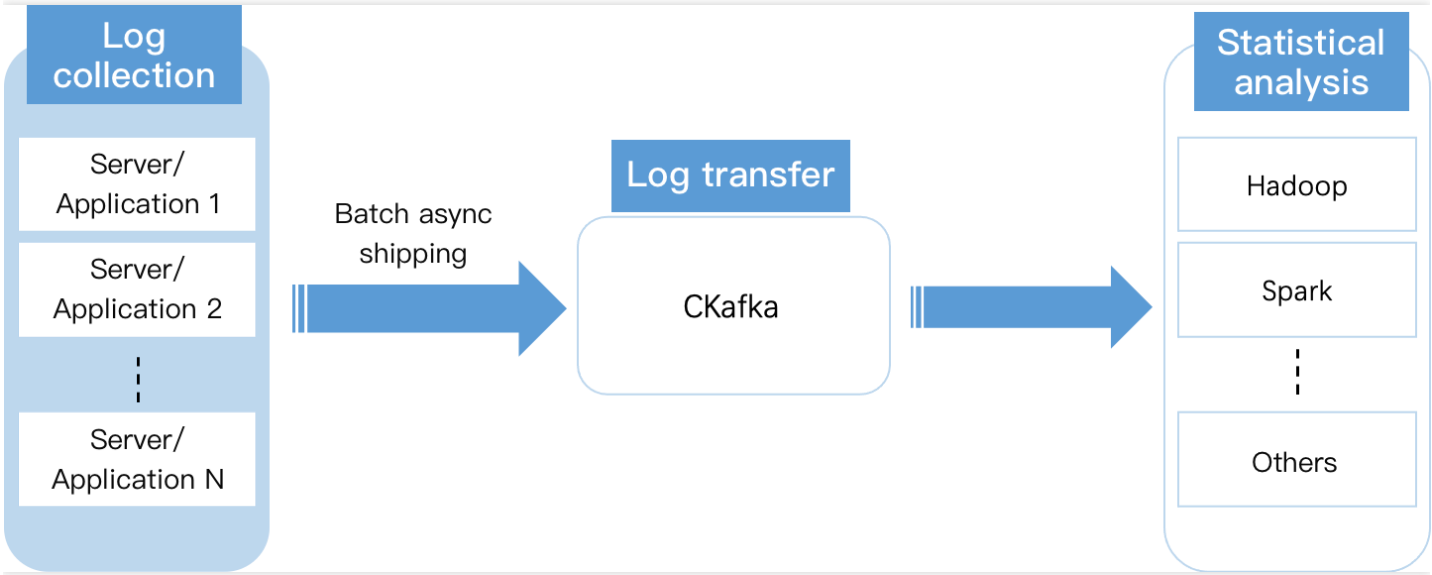
©2013-2022 Tencent Cloud. All rights reserved.



Log Aggregation

The low-latency processing capability of CKafka makes it easier to sustain distributed processing (consumption) of data from multiple sources. Compared to a centralized log aggregation system, CKafka can achieve stronger persistence guarantee and lower end-to-end latency while providing the same performance.
The features of CKafka make it an ideal "log collection center". Multiple servers/applications can send operation logs to a CKafka cluster "in batches" and "asynchronously" with no need to store them locally or in a database. CKafka can submit/compress messages in batches, so that the producer can hardly perceive the performance overhead. In this case, the consumer can use systematic storage and analysis systems such as Hadoop to perform statistical analysis on the pulled logs.

©2013-2022 Tencent Cloud. All rights reserved.



Big Data Scenarios

In some business scenarios involving big data, massive amounts of concurrent data need to be processed and aggregated, so high cluster processing performance and scalability are required. Thanks to its advantages in data distribution mechanism, allocation of disk storage space, processing of message formats, server selection, and data compression, CKafka is suitable for processing high numbers of real-time messages and can aggregate the data generated by distributed applications for easier system OPS.
Specifically, CKafka can process offline data or streaming data effectively and aggregate and analyze data easily.

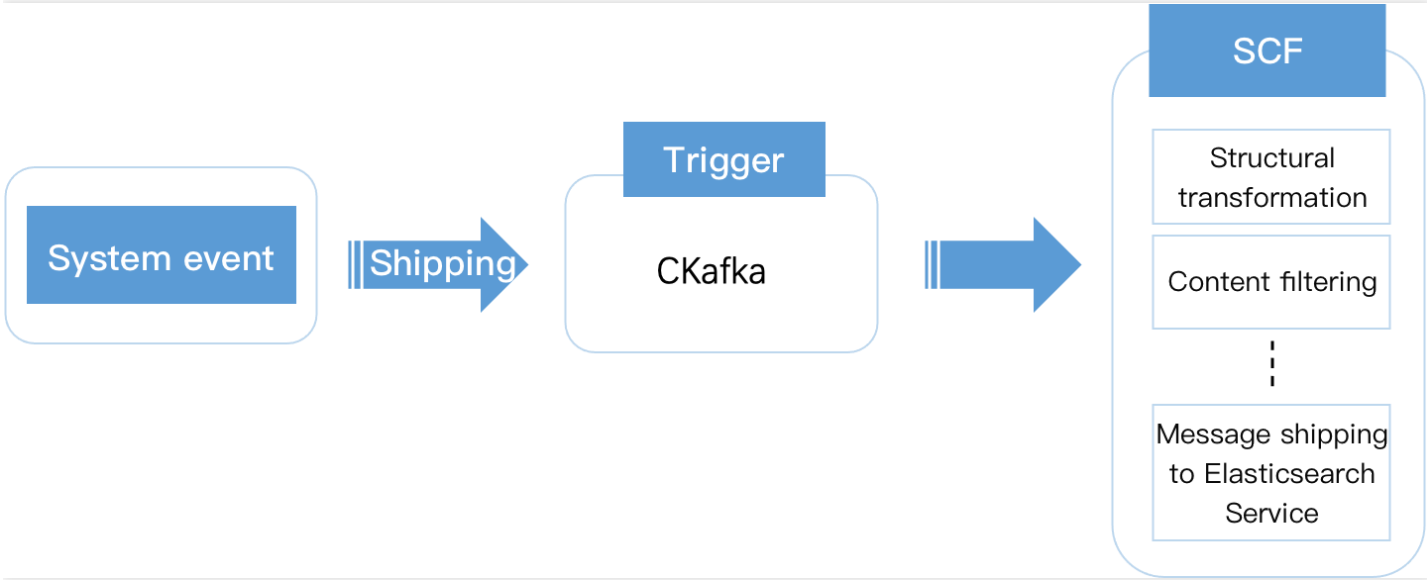
Function Trigger

CKafka can be used as SCF function triggers, and when a message is received, a function can be triggered and the message will be passed to the function as event content. For example, when CKafka triggers a function, the function can transform the message structure, filter the message contents, or deliver the message to Elasticsearch Service (ES).

Note：

For more information on the availability of SCF, see Service Level Agreement for SCF.

©2013-2022 Tencent Cloud. All rights reserved.

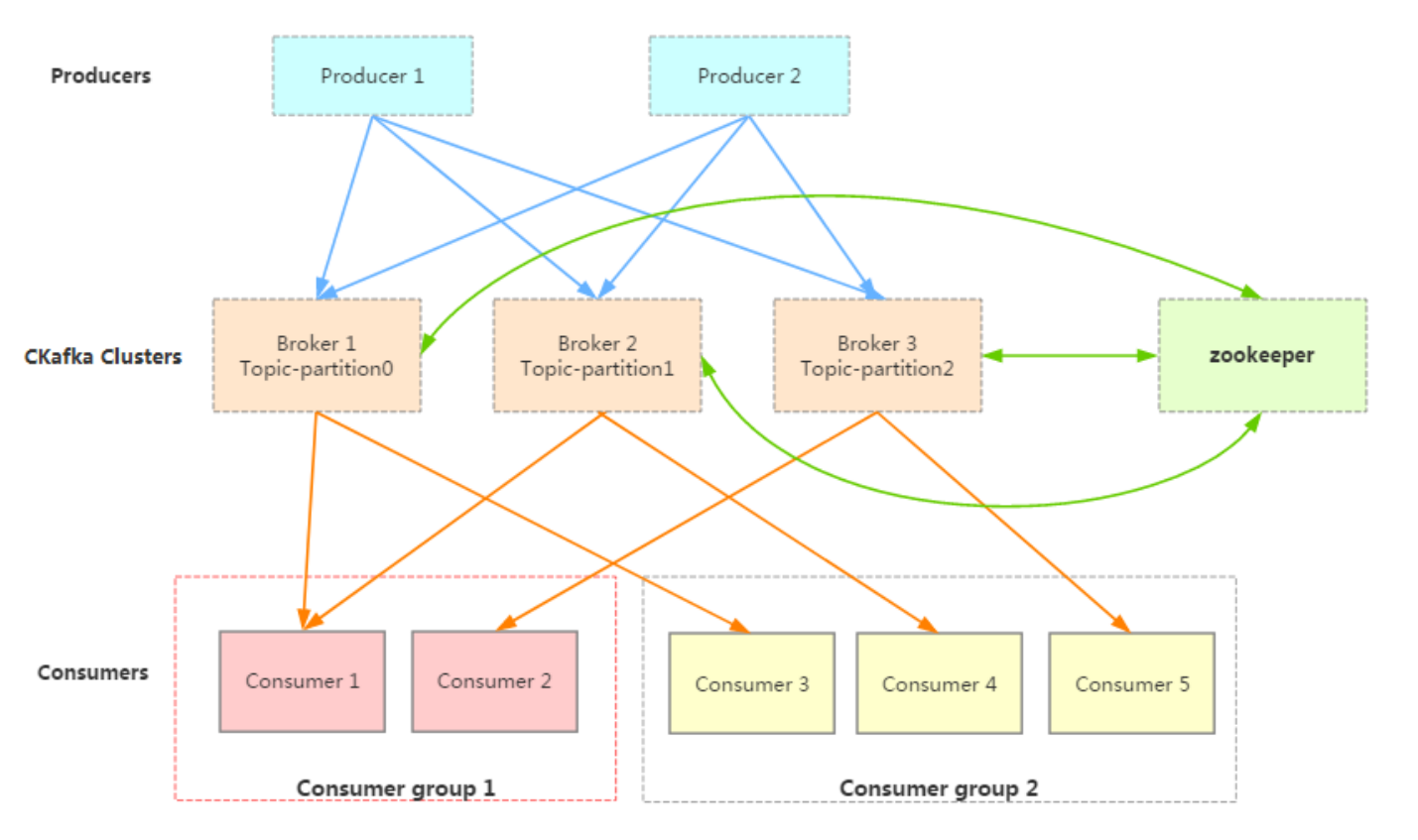


©2013-2022 Tencent Cloud. All rights reserved.

How It Works

Last updated：2022-01-14 10:59:47

The architecture of CKafka is as follows:



- A producer can be information such as messages generated by webpage activities and service logs. It publishes messages to CKafka's broker cluster in push mode.
- ZooKeeper is used to manage the cluster configuration, elect the leader, implement fault tolerance, and do more in the cluster.
- Consumers are divided into several consumer groups. A consumer consumes messages from the broker in pull mode.

For the advantages of CKafka over self-built open-source Apache Kafka, see Strengths.

High Throughput

In CKafka, a huge amount of network data is permanently stored in disks and high numbers of disk files are sent over the network. The performance of this process directly affects Kafka's overall throughput, especially in the following

©2013-2022 Tencent Cloud. All rights reserved.

aspects:

- **Improved disk utilization**: data is read and written sequentially in the disk, which helps increase the disk utilization.
  - Message write: messages are written to the page cache and flushed by the async thread.
  - Message read: messages are transferred directly from the page cache into the socket and then sent out.
  - If the corresponding data is not found in the page cache, disk IO will be caused, and the messages will be loaded from the disk to the page cache and then sent out directly from the socket.
- **Broker's zero copy mechanism**: the sendfile system is called to send data directly from the page cache to the network.
- **Reduced network overheads**
  - Data compression reduces the network load.
  - Batch processing mechanism: the producer writes data to the broker in batch, while the consumer pulls data from the broker in batch.

Data Persistence

Data persistence is mainly implemented in CKafka through the following principles:

- **Partition storage distribution in topic**
  In the file storage of CKafka, a topic has multiple different partitions, each of which physically corresponds to a folder. Messages and index files are stored in these partitions. For example, if two topics are created where topic 1 has 5 partitions and topic 2 has 10 partitions, then a total of 5 + 10 = 15 folders will be generated in the cluster.

- **File storage method in partition**
  A partition is physically composed of multiple segments of equal size. These segments are read from/written to sequentially and are deleted quickly upon expiration, which improves the disk utilization.

Scale-out

- One topic can include multiple partitions distributed in one or more brokers.
- One consumer can subscribe to one or more of these partitions.
- A producer is responsible for equally assigning messages to partitions.
- Messages in partitions are sequential.

Consumer Group

©2013-2022 Tencent Cloud. All rights reserved.

- CKafka does not delete consumed messages.
- A consumer must belong to a group.
- Consumers in the same consumer group do not consume the same partition at the same time.
- Different groups consume the same message at the same time, which is more diversified (queuing model and publish-subscribe model).

Multiple Replicas

The multi-replica design can enhance the system availability and reliability.
Replicas are evenly distributed across the entire cluster. The replica algorithm is as follows:

1. All brokers (assuming n brokers in total) and the partitions to be assigned are sorted.
2. The ith partition is assigned to the (i mod n)th broker.
3. The jth replica of the ith partition is assigned to the ((i + j) mod n)th broker.

Leader Election Mechanism

CKafka dynamically maintains a set of in-sync replicas (ISR) in ZooKeeper, and all replicas in ISR catch up to the leader. Only members of the ISR can be elected as leaders.

- If there are f + 1 replicas in ISR, a partition can tolerate f replica failures while guaranteeing that committed message will not be lost.
- If there is a total of 2f + 1 replicas (including the leader and followers), it must be guaranteed that f+1 replicas have replicated the messages before the commit operation. To ensure that a new leader can be correctly elected, the number of failed replicas cannot exceed f.

©2013-2022 Tencent Cloud. All rights reserved.

Comparison with Apache Kafka

Last updated：2022-08-11 16:42:22

The performance comparison between CKafka and open-source Apache Kafka is as follows:

Feature	CKafka	Apache Kafka
Basic features	Supports adjusting topic parameter configurations and the number of topic partitions, sending messages, and resetting the consumption offset in the console; supports visual management for clusters, topics, and consumer groups.	Topic parameters and the number of partitions can only be configured with command lines in a non-customized manner; message sending is not supported; the consumption offset must be reset manually, which is complicated and error-prone; and an open-source management system is required, making Apache Kafka less user-friendly.
Smart Ops	Supports automatic disk capacity scaling, quick diagnosis, smart inspection (with solutions provided), and smart disk configuration.	Not supported.
Isolation	The following features are supported: creating multiple instances based on one physical cluster; usage limit based on bandwidth and disk capacity; topic-level traffic throttling; the configuration of multiple access points; the isolation between the control plane and data plane; logically isolating cross-account data.	Not supported.
Monitoring and alarming	Offers out-of-the-box deployment and monitoring solutions that are mature and standardized; supports multidimensional monitoring and alarming, metric sorting, and viewing the details of logs (such as exception event logs) in the console for troubleshooting.	Not supported.
High availability	Supports high-availability and multi-AZ deployment and offers mature failure recovery solutions; supports upgrading from single-AZ deployment to multi-AZ deployment as well as cross-region disaster recovery.	Supported, but the process is time-consuming.

©2013-2022 Tencent Cloud. All rights reserved.

Feature	CKafka	Apache Kafka
Security compliance	Supports ACL in topic dimension; supports configuring SASL and SSL authentication in the console; supports the permission control for management and control operations (which are rewindable) via CloudAudit	Parameters need to be configured with command lines, which is complicated and error-prone; operations are non-rewindable.
Others	You can seamlessly upgrade CKafka to a new version when the community edition has bugs or security vulnerabilities.	N/A

©2013-2022 Tencent Cloud. All rights reserved.

Use Limits

Last updated：2022-11-01 16:23:38

This document lists the limits of certain metrics and performance in CKafka. Be careful not to exceed the limits during use to avoid exceptions.

Item	Description
Number of topics	The topic limit depends on product specifications. For details, see Billing Overview.
Number of partitions	• One topic can support up to 3,000 partitions. • The number of instance-level partitions is the partition limit per topic * the number of topics (generally 2 or 3). • The number of partitions cannot be reduced.
Partition throughput	• In the case of `ack = 1`, the throughput of a single CKafka partition is between 30 and 60 MB/sec due to factors such as CKafka's partition architecture, business data size, and request frequency. • In the case of `ack = -1` (for strong consistency), **we recommend that the throughput of a single CKafka partition be between 10 and 20 MB/sec**, so that the stability of the request duration is not affected by factors such as CKafka's partition architecture, business data size, and request frequency.
Duration	Featuring high-traffic and high-throughput message queues, Kafka cannot guarantee that each request has low latency. The recommended timeout period settings are as follows: • For the producer, when ACK = 1, the timeout period defaults to **30s**; • For the producer, when ACK = -1, the timeout period defaults to **60s**; • The timeout period for the consumer is set to **60s**.
Number of consumer groups	• For Standard Edition, the recommended number of instance-level consumer groups is up to 50. • For Pro Edition, the recommended number of instance-level consumer groups is up to 200. You can submit a ticket to apply for more.
Instance	• The region attribute of instances cannot be changed. • The maximum number of client connections to a Standard Edition instance is 5,000, and to a Pro Edition instance, 10,000. When the limit is reached, the client cannot create more connections. If this maximum value is unreasonable for your actual business, you can submit a ticket to increase it.
Version	• Standard Edition: It is compatible with open-source versions 0.9, 0.10, and 1.1. v1.1 is installed by default. Customized versions are not supported. • Pro Edition: It is compatible with open-source versions 0.9, 0.10, 1.1, 2.4, and 2.8.

©2013-2022 Tencent Cloud. All rights reserved.

Item	Description
Routes	An instance can create up to five routes, with only one of them being a public network route.
Public network bandwidth	CKafka provides a public network bandwidth of 3 Mbps for free by default. Pro Edition instances can upgrade public network bandwidth to 198 Mbps additionally.
Exposing ZooKeeper	It is not supported.
Exposing underlying resources	It is not supported so as to avoid risks caused by user's operation.
Message size	It cannot exceed 12 MB; otherwise, messages will fail to be sent.
Tag	Each Tencent Cloud resource can have up to 50 tags.
Concurrent operations in the console	Some management and control operations may cause the instance metadata to be modified, leading to data inconsistency when the concurrency is high, and underlying data distribution will be affected. Therefore, some API operations will be locked to limit the request concurrency. To improve the stability and success rate of console operations, up to **20** concurrent console requests (including those directly initiated via SDK) can be initiated for **a single instance**.

Note

Due to CKafka's partition architecture, business data size, request frequency, and the stability of the physical layer, CKafka cannot guarantee that each request has low latency.

- We will try our best to guarantee that:

1. The percentage of an instance's monthly low production duration is the same as the instance's stability SLA.
2. The percentage of an instance's monthly low consumption duration is the same as the instance's stability SLA.

- The calculation formula is as follows:
  The percentage of low production duration in a month = (the number of minutes when the production duration 99.9th percentile per minute is less than or equal to 30s that month / the monthly service duration in minutes) * 100%
  The percentage of low consumption duration in a month = (the number of minutes when the consumption

©2013-2022 Tencent Cloud. All rights reserved.

duration 99.9th percentile per minute is less than or equal to 60s that month / the monthly service duration in minutes) * 100%

©2013-2022 Tencent Cloud. All rights reserved.

Regions and AZs

Last updated：2022-11-11 11:12:36

A region is the physical location of an IDC. Availability Zone (AZ) refers to the physical IDC of Tencent Cloud in the same region with independent power supplies and network resources. For more information, see CVM - Regions and AZs.

Supported Regions

China

Region		Value
South China	Guangzhou	ap-guangzhou
	Shenzhen	ap-shenzhen
East China	Shanghai	ap-shanghai
	Nanjing	ap-nanjing
	Hangzhou	ap-hangzhou
North China	Beijing	ap-beijing
	Tianjin	ap-tianjin
Central China	Changshan	ap-changsha
Southwest China	Chengdu	ap-chengdu
	Chongqing	ap-chongqing
Hong Kong/Macao/Taiwan (China)	Taipei (China)	ap-taipei
	Hong Kong (China)	ap-hongkong

Other countries and regions

Region		Value
Southeast Asia Pacific	Singapore	ap-singapore

©2013-2022 Tencent Cloud. All rights reserved.

Region		Value
	Bangkok	ap-bangkok
	Jakarta	ap-jakarta
South Asia Pacific	Mumbai	ap-mumbai
Northeast Asia Pacific	Seoul	ap-seoul
	Tokyo	ap-tokyo
West US	Silicon Valley	na-siliconvalley
East US	Virginia	na-ashburn
North America	Toronto	na-toronto
South America	São Paulo	sa-saopaulo
Europe	Frankfurt	eu-frankfurt
	Moscow	eu-moscow

Region and AZ Selection

When selecting a region and AZ, take the following into consideration:

- The geographic locations of CKafka instances, your business, and your target users:
  We recommend you choose the region closest to your end users when purchasing CKafka instances to minimize access latency and improve access speed.
- Relationship between CKafka and other Tencent Cloud services:
  When you select other Tencent Cloud services, we recommend you try to locate them all in the same region and AZ to allow them to communicate with each other through the private network, reducing access latency and increasing access speed.
- High availability and disaster recovery.
  Even if you have just one VPC, we still recommend you deploy your businesses in different AZs to prevent a single point of failure and enable cross-AZ disaster recovery.
- There may be network latency among different AZs. We recommend you assess your business requirements and find the optimal balance between high availability and low latency.
- If you need access to CKafka instances in other countries or regions, we recommend you select an instance in those other countries or regions. If you use a CKafka instance in China to access servers in other countries and

©2013-2022 Tencent Cloud. All rights reserved.

regions, you may encounter a high higher network latency.

Resource Availability

The following table describes which CKafka resources are global, which are regional, and which are specific to AZs.

Resource	Resource ID Format -8-Digit String of Numbers and Letters	Type	Description
User Account	No limit	Globally unique	Users can use the same account to access Tencent Cloud resources from around the world.
CKafka instance	ckafka-xxxxxxxx	Instances are specific to an AZ.	A CKafka instance created in an AZ is not available to other AZs.

Related Operations

Migrating instance to another AZ

CKafka supports instance migration across AZs in the same region. If you need this, submit a ticket for assistance.

©2013-2022 Tencent Cloud. All rights reserved.