KAFKA

Kafka-SSL , AzureKafka-SASSL .

Kafka

.

Auzere EventHub(KAFKA) With Actor

:

- Kafka : KAFKA
- AKKA AkkaDotModule : AkkaDotModule
- Kafka AKKA-

Git : https://github.com/psmon/AkkaDotModule

Consumer Producer

Consumer Producer .

```
## KAfka Stream

  Kafka    .

    // KAFKA
    var consumerSystem = app.ApplicationServices.GetService<ConsumerSystem>();
    var producerSystem = app.ApplicationServices.GetService<ProducerSystem>();

    // :
    consumerSystem.Start(new ConsumerAkkaOption()
    {
        KafkaGroupId = "testGroup",
        KafkaUrl = "kafka:9092",
        RelayActor = null,          //    ,
        Topics = "akka100"
    });

    // :
    producerSystem.Start(new ProducerAkkaOption()
    {
        KafkaUrl = "kafka:9092",
        ProducerName = "producer1"
    });

    List<string> messages = new List<string>();
    // :    .(      )
    int tps = 10;
    producerSystem.SinkMessage("producer1", "akka100", messages, tps);
```

Kafka Client

 ,   .

akka.kafka.conf - Kafka

```
akka.loglevel = ERROR

# // #producer-settings
# Properties for akka.kafka.ProducerSettings can be
# defined in this section or a configuration section with
# the same layout.
akka.kafka.producer {
```

```
# Config path of Akka Discovery method
# "akka.discovery" to use the Akka Discovery method configured for the ActorSystem
discovery-method = akka.discovery

# Set a service name for use with Akka Discovery
# https://doc.akka.io/docs/alpakka-kafka/current/discovery.html
service-name = ""

# Timeout for getting a reply from the discovery-method lookup
resolve-timeout = 3 seconds

# Tuning parameter of how many sends that can run in parallel.
# In 2.0.0: changed the default from 100 to 10000
parallelism = 10000

# Duration to wait for `KafkaProducer.close` to finish.
close-timeout = 60s

# Call `KafkaProducer.close` when the stream is shutdown. This is important to override to false
# when the producer instance is shared across multiple producer stages.
close-on-producer-stop = true

# Fully qualified config path which holds the dispatcher configuration
# to be used by the producer stages. Some blocking may occur.
# When this value is empty, the dispatcher configured for the stream
# will be used.
use-dispatcher = "akka.kafka.default-dispatcher"

# The time interval to commit a transaction when using the `Transactional.sink` or `Transactional.flow`
# for exactly-once-semantics processing.
eos-commit-interval = 100ms

# Properties defined by org.apache.kafka.clients.producer.ProducerConfig
# can be defined in this configuration section.
kafka-clients {
}
}
# // #producer-settings

# // #consumer-settings
# Properties for akka.kafka.ConsumerSettings can be
# defined in this section or a configuration section with
# the same layout.
akka.kafka.consumer {
# Config path of Akka Discovery method
# "akka.discovery" to use the Akka Discovery method configured for the ActorSystem
discovery-method = akka.discovery

# Set a service name for use with Akka Discovery
# https://doc.akka.io/docs/alpakka-kafka/current/discovery.html
service-name = ""

# Timeout for getting a reply from the discovery-method lookup
resolve-timeout = 3 seconds

# Tuning property of scheduled polls.
# Controls the interval from one scheduled poll to the next.
poll-interval = 50ms

# Tuning property of the `KafkaConsumer.poll` parameter.
# Note that non-zero value means that the thread that
# is executing the stage will be blocked. See also the `wakup-timeout` setting below.
poll-timeout = 50ms

# The stage will delay stopping the internal actor to allow processing of
# messages already in the stream (required for successful committing).
# This can be set to 0 for streams using `DrainingControl`.
stop-timeout = 30s

# Duration to wait for `KafkaConsumer.close` to finish.
close-timeout = 20s
```

```
# If offset commit requests are not completed within this timeout
# the returned Future is completed `CommitTimeoutException`.
# The `Transactional.source` waits this ammount of time for the producer to mark messages as not
# being in flight anymore as well as waiting for messages to drain, when rebalance is triggered.
commit-timeout = 15s

# If commits take longer than this time a warning is logged
commit-time-warning = 1s

# Not relevant for Kafka after version 2.1.0.
# If set to a finite duration, the consumer will re-send the last committed offsets periodically
# for all assigned partitions. See https://issues.apache.org/jira/browse/KAFKA-4682.
commit-refresh-interval = infinite

# Fully qualified config path which holds the dispatcher configuration
# to be used by the KafkaConsumerActor. Some blocking may occur.
use-dispatcher = "akka.kafka.default-dispatcher"

# Properties defined by org.apache.kafka.clients.consumer.ConsumerConfig
# can be defined in this configuration section.
kafka-clients {
# Disable auto-commit by default
enable.auto.commit = false
}

# Time to wait for pending requests when a partition is closed
wait-close-partition = 500ms

# Limits the query to Kafka for a topic's position
position-timeout = 5s

# When using `AssignmentOffsetsForTimes` subscriptions: timeout for the
# call to Kafka's API
offset-for-times-timeout = 5s

# Timeout for akka.kafka.Metadata requests
# This value is used instead of Kafka's default from `default.api.timeout.ms`
# which is 1 minute.
metadata-request-timeout = 5s

# Interval for checking that transaction was completed before closing the consumer.
# Used in the transactional flow for exactly-once-semantics processing.
eos-draining-check-interval = 30ms

# Issue warnings when a call to a partition assignment handler method takes
# longer than this.
partition-handler-warning = 5s

# Settings for checking the connection to the Kafka broker. Connection checking uses `listTopics` requests with
the timeout
# configured by `consumer.metadata-request-timeout`
connection-checker {

#Flag to turn on connection checker
enable = false

# Amount of attempts to be performed after a first connection failure occurs
# Required, non-negative integer
max-retries = 3

# Interval for the connection check. Used as the base for exponential retry.
check-interval = 15s

# Check interval multiplier for backoff interval
# Required, positive number
backoff-factor = 2.0
}

}
# // #consumer-settings
# // #committer-settings
# Properties for akka.kafka.CommitterSettings can be
# defined in this section or a configuration section with
# the same layout.
akka.kafka.committer {

# Maximum number of messages in a single commit batch
max-batch = 1000

# Maximum interval between commits
max-interval = 10s

# Parallelsim for async committing
parallelism = 100

# API may change.
# Delivery of commits to the internal actor
# WaitForAck: Expect replies for commits, and backpressure the stream if replies do not arrive.
# SendAndForget: Send off commits to the internal actor without expecting replies (experimental feature since
1.1)
delivery = WaitForAck
}
# // #committer-settings

# The dispatcher that will be used by default by consumer and
# producer stages.
akka.kafka.default-dispatcher {
type = "Dispatcher"
executor = "thread-pool-executor"

thread-pool-executor {
fixed-pool-size = 16
}
}
```

KAFKA Docker Compose

DockerCompose .

```
version: '3.5'
services:
  zookeeper:
    image: 'bitnami/zookeeper:latest'
    ports:
    - '2181:2181'
    environment:
    - ALLOW_ANONYMOUS_LOGIN=yes
  kafka:
    hostname: kafka
    image: 'bitnami/kafka:latest'
    ports:
    - '9092:9092'
    environment:
    - KAFKA_ADVERTISED_HOST_NAME=kafka
    - KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181
    - ALLOW_PLAINTEXT_LISTENER=yes
```

> ### *kafka*
>
> *host*
>
> *kafka 127.0.0.1*

: Apakka Kafka Stream Akka .

- https://alpakka.getakka.net/ -
- https://doc.akka.io/docs/alpakka/current/index.html -