



### Publishing your Mainframe Data ...

IT requirements have picked up on the speed in which the world seems to change, with

- Constant new feature requests to keep ahead of the competition
- Extremely valuable business information, where data is key
- Data to be available anywhere at any time – and still be secure
- Agile development and self-optimizing DevOps requirements



### ... to your Kafka Cluster.

Kafka has become one of the best known platforms for event-based processing and streaming of data.

A Kafka cluster shares the mainframes core capabilities: high availability (resilience), scalability and permanent storage.

Kafka client libraries allow to connect from almost anywhere and anything to your Kafka cluster. On z/OS though, connecting to a Kafka cluster in the open world is quite complex. There is no easy way to integrate Kafka client libraries.

On the other side, many of the most important business processes and a huge amount of business data resides on the mainframe, with the need to be integrated into your cloud infrastructure.

### zKafka Connector – A Smart Link between the Mainframe and your Kafka Cluster

The zKafka Connector offers an easy way to produce (publish) and consume (read) data from any z/OS application to and from your Kafka cluster. It does not require any further program product, nor middleware. The zKafka Connector significantly simplifies your mainframe data integration with the cloud.

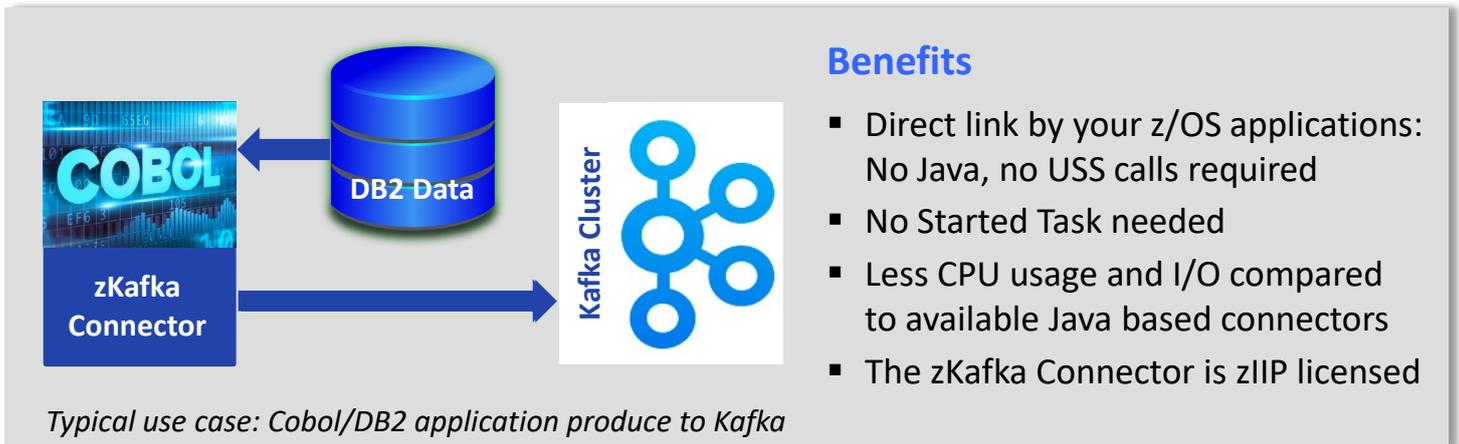
*Your data is exchanged directly between the z/OS environment and your Kafka cluster, wherever this resides.*



# Direct Kafka Connections for all of your z/OS Applications.

## Features

- The integration into any of your applications is easy with standard z/OS CALL APIs
- With just a few API calls and a simple list of parameters your z/OS applications connect to a Kafka cluster
- The zKafka Connector is available for z/OS and Open Systems.
- It provides state-of-the-art security using SSL/TLS encryption and SASL authentication
- No additional software or middleware is required



## Benefits

- Direct link by your z/OS applications: No Java, no USS calls required
- No Started Task needed
- Less CPU usage and I/O compared to available Java based connectors
- The zKafka Connector is zIIP licensed

## Comfort

- The zKafka Connector manages the complex connections to your Kafka cluster
- In case of connection loss, the zKafka Connector performs the error handling and recovery
- The accurate states of transfers and all errors are reported back by the zKafka Connector API
- Use the standard z/OS CALL API in any z/OS application (Cobol, PL/1, C/C++, REXX, ASM)
- The zKafka Connector supports data conversion between EBCDIC and UTF-8 formats, or just sends binary data
- Different levels of debug messages are available for development and testing

### \* COBOL API

```
CALL "RZKPROD" USING
  BY VALUE      RZK_HANDLE
  BY CONTENT    "MyKey" BY VALUE 5
  BY CONTENT    "MyMsg" BY VALUE 5
  BY VALUE      UTF_8
  RETURNING     RZK_RC.
```

### /\* REXX API \*/

```
Hnd = RzkOpen ();
Rzk = RzkConn (Hnd, "domain", "port", "topic");
Rzk = RzkProd (Hnd, "MyKey", "MyMsg", "UTF-8");
Rzk = RzkClse (Hnd);
```

RETURN 0;

### /\* Batch Utility

```
//PRODUCE EXEC PGM=RZKCOPY
//RZKFILE DD DISP=SHR,DSN=TO.PRODUCE
//RZKPARM DD *
PRODUCE domain port topic
MyKey
UTF-8
/*
```