

KIP-431: Support of printing additional ConsumerRecord fields in DefaultMessageFormatter

- [Proposed Changes](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)

Status

Current state: *Accepted*

Discussion thread: [here](#)

Voting thread: [here](#)

JIRA: [KAFKA-6733](#)

PR: [#9099](#) (based on [#4807](#))

Motivation

ConsoleConsumer (kafka-console-consumer) is a very important debugging tool in Kafka. However, currently it cannot print offset, partition and headers of a Kafka record. So, during debugging session, if we need to know those information, we have to use kafka-dump-log, which require file system access to the Kafka broker hosts, or use custom application, like kafkacat. This KIP is proposing to rectify this issue.

Currently, without any extra property, ConsoleConsumer prints the value of the record.

And the following properties control the ConsoleConsumer output format:

- print.timestamp
- print.key
- print.value
- print.partition
- key.separator
- line.separator
- key.deserializer
- value.deserializer

As you see, there is no way to print message offset, partition and headers. Also, there is no easy way to differentiate null value or blank string.

Public Interfaces

New properties to DefaultMessageFormatter which can be specified in kafka-console-consumer command line arguments with `--property` argument.

Property	Valid Values	Default Value	Description
print.offset	"true" or "false"	"false"	print message offset
print.headers	"true" or "false"	"false"	print headers, will print NO_HEADERS if the record has no headers
header.separator	string	","	separator printed between each header's key:value pair
headers.deserializer	class name	"StringDeserializer"	header value deserializer
null.literal	string	"null"	literal to print if the value is NULL (size -1)

Example output:

```
$ kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic test --from-beginning --property print.partition=true --property print.key=true --property print.timestamp=true --property print.offset=true --property print.headers=true --property key.separator='|'
```

```
CreateTime:1592475472398|Partition:0|Offset:3|h1:v1,h2:v2|key1|value1
CreateTime:1592475472456|Partition:0|Offset:4|NO_HEADERS|key2|value2
```

Proposed Changes

Add new properties to `DefaultMessageFormatter` as per the above:

- `print.offset`
- `print.headers`
- `header.separator`
- `headers.deserializer`
- `null.literal`

Compatibility, Deprecation, and Migration Plan

KIP-431 introduces incompatibility when `"print.partition=true"` (this property exists in the code before KIP-431 but not documented). Before KIP-431, `"kafka-console-consumer"` prints the partition as a number after the value for example: `"key1|value1|0"`. After this KIP, `"kafka-console-consumer"` prints the partition number prefixed with `"Partition:"` before the key (if printed) and value, for example: `"Partition:0|key1|value1"`. Because this property was not documented, no migration plan is implemented in KIP-431.

The other changes are backward compatible because they do not exist before. Apart from `"print.partition=true"`, if a user does not use any new parameters, then the output of console consumer will look the same as before.

Rejected Alternatives

- Use `kafka-dump-log` command to get these information. Rejected because this requires file system access to the Kafka broker's `log.dirs`, which is not secure.
- Use other console consumer application to access these information. Rejected because Kafka brokers may run in airtight environment or as containers, preventing us from installing other applications than the one coming with Kafka brokers.