

KIP-407: Kafka Connect support override worker kafka api configuration with connector configuration that post by rest api

- [Status](#)
- [Motivation](#)
- [Public Interfaces](#)
- [Proposed Changes](#)
 - [use case](#)
- [Compatibility, Deprecation, and Migration Plan](#)

Status

Current state: *[One of "Under Discussion", "Accepted", "Rejected"]*

Discussion thread: [here](#) *[Change the link from the KIP proposal email archive to your own email thread]*

JIRA: [here](#)

Motivation

I'm using kafka sink connect; The config "auto.offset.reset" is set in connect-distributed*.properties; It works for all connector which in one worker; So the consumer will poll records from latest or earliest; I can not control the 'auto.offset.reset' connector configs post with rest api;

I think is necessary to override worker kafka api configs with connector configs;

With the improvement, connectors with different auto.offset.reset policy can be exist in the same worker;

Public Interfaces

The new configurations start with 'consumer.' for sink connector and new configurations start with 'producer.' will be support for REST interface **'POST /connectors'**;

```
{
  "name": "test",
  "config": {
    "consumer.auto.offset.reset": "latest", # override consumer.auto.offset.reset in connect-distributed*.properties
    "consumer.max.poll.records": "200",
    "consumer.xxx": "xxx"
    "connector.class": "com.laomei.sis.solr.SolrConnector",
    "tasks.max": "1",
    "poll.interval.ms": "100",
    "connect.timeout.ms": "60000",
    "topics": "test"
  }
}
```

Proposed Changes

use case

For the sink connector, we can support override kafka consumer with 'consumer.xxx' configurations;

For the source connector, we can support override kafka producer with 'producer.xxx' configurations;

```
{
  "name": "test",
  "config": {
    "consumer.auto.offset.reset": "latest", # override consumer.auto.offset.reset in connect-distributed*.
properties
    "consumer.xxx": "xxx"
    "connector.class": "com.laomei.sis.solr.SolrConnector",
    "tasks.max": "1",
    "poll.interval.ms": "100",
    "connect.timeout.ms": "60000",
    "topics": "test"
  }
}
```

Compatibility, Deprecation, and Migration Plan

- If there are kafka connect implementation which the prefix of configurations start with 'consumer.' or 'producer.', the connect will be impact;
- The older behavior will not be changed if the older configurations not contains config start with 'consumer' in sink connector or 'producer' in source connector;
- No migration tool required