KIP-913: add new method to provide possibility for accelerate first record's sending

Status

Current state: *Under Discussion*

Discussion thread: *here*

JIRA: Kafka-14768

Motivation

Sometimes, we found the users' functional interaction take a lot of time. At last, we figure out the root cause is that after we complete deploy or restart the servers.

The first message's delivery on each application server by kafka client will take much time.

After analyzing the source code about the first time's sending logic. The time cost is caused by the getting metadata before the sending. The latter's sending won't take the much time due to the cached metadata. The metadata's fetching logic is right and necessary. Thus, we still want to improve the experience for the first message's send/user first interaction.

So, This KIP try to raise one solution to improve it.

The solution is that we can provide one method for metadata fetch into Producer. When the application restarted/started, it can call it before mark the application is ready for handle requests. So, when the first request/record be handled. the metadata had been fetched so that it's handle's speed will be much faster.

```
public Cluster getCluster(String topic, long maxBlockTimeMs) {
        Objects.requireNonNull(topic, "topic cannot be null");
        try {
            return waitOnMetadata(topic, null, time.milliseconds(), maxBlockTimeMs).cluster;
        } catch (InterruptedException e) {
            throw new InterruptException(e);
}
}
```

note: waitOnMetadata(topic, null, time.milliseconds(), maxBlockTimeMs) is the existed method with provide modifier.

Public Interfaces

add new interface with tiny refactor which reduce the duplicated code.

Cluster getCluster(String topic, long maxBlockTimeMs);

Proposed Changes

The changes can refer to the example PR: https://github.com/apache/kafka/pull/13320/files

The core code had been listed in Motivation part.

Compatibility, Deprecation, and Migration Plan

If user want to use the feature, user can upgrade the client can call the new added method.

If user don't have requirement for it, there isn't any need to do any change.

What's more, new client version's upgrade also won't influence existed behavior.

- *What impact (if any) will there be on existing users?*
  no impact on existed users.
- *If we are changing behavior how will we phase out the older behavior?*
  no changing older behavior.
- *If we need special migration tools, describe them here.*
  no.
- *When will we remove the existing behavior?*
  no need to remove.

Test Plan

Compare the first record's send time cost to see if any improvement happens.

Rejected Alternatives

maybe we can provide one dedicated method with more naming instead of "getCluster".