

KIP-520: Add overloaded Consumer#committed for batching partitions

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

Status

Current state: *Accepted (2.4.0)*

Discussion thread: [link](#)

JIRA: [KAFKA-8880](#)

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

We've observed that many usage of the `consumer.committed` calls are not only for a single partition, but for a batch of partitions. On the other hand, the OffsetFetchRequest protocol actually allows for multiple partitions within one request.

So it's a natural proposal to extend our existing `consumer#committed` function to take a set of topic-partitions.

Public Interfaces

We propose to add two overloaded function of `consumer#committed` that takes a set of topic-partitions, and deprecate the existing two functions that only take one topic-partition. At the moment the latter would be implemented as passing in a singleton set to the newly added functions.

```
@Deprecated
OffsetAndMetadata committed(TopicPartition partition);

@Deprecated
OffsetAndMetadata committed(TopicPartition partition, final Duration timeout);

Map<TopicPartition, OffsetAndMetadata> committed(Set<TopicPartition> partitions); // new API

Map<TopicPartition, OffsetAndMetadata> committed(Set<TopicPartition> partitions, final Duration timeout); //
new API
```

Proposed Changes

As above. Internal implementations are straight-forward and would not be discussed here.

Compatibility, Deprecation, and Migration Plan

- *There is no direct impact on users since we only add new overloaded functions while deprecating the old ones.*
- *When we eventually remove the deprecated functions users need to change their code to call the new functions with singleton set.*

Rejected Alternatives

None.