

KIP-469: Pre-registration support for static members

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

Status

Current state: *Under Discussion*

Discussion thread: TBD

JIRA: [KAFKA-8397](#) - Getting issue details... STATUS

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

Motivation

Rebalance during scaling up is always painful. Every newly joined member will keep the group at rebalancing stage until all of new instances finished bootstrapping. There could be multiple shuffling of active tasks around existing and new instances, thus decreasing the entire system availability. This negative impact has been mitigated after we introduced [KIP-345](#). Under static membership, user could provide a list of hard-coded `group.instance.id`'s to pre-register their identities on broker if the new host info is known, so that broker coordinator could respond to scaling operations more intelligently. For example when we scale up the fleet by defining 4 new client instance ids, the server shall wait until all 4 new members to join the group before kicking out only one rebalance, instead of four in the worst case.

Proposed Changes

This change requires us to change JoinGroup protocol to batch mode in order to easily scale multiple members at once.

Public Interfaces

We will bump JoinGroup request/response version to support batch adding members.

```

JoinGroupRequest => GroupId SessionTimeout RebalanceTimeout MemberId GroupInstanceId ProtocolType GroupProtocols
  GroupId          => String
  SessionTimeout   => int32
  RebalanceTimeout => int32
  MemberId         => String // removed
  GroupInstanceId  => String // removed
  ProtocolType     => String
  GroupProtocols   => [Protocol MemberMetadata]
  Protocol         => String // removed
  MemberMetadata   => bytes // removed
  JoinGroupMembers => []JoinGroupRequestMember // new
                                     MemberId          => String // new
                                     GroupInstanceId    => String // new
                                     Protocol           => String // new

JoinGroupResponse => ThrottleTime ErrorCode GenerationId ProtocolName LeaderId MemberId Members
  ThrottleTime      => int16
  ErrorCode          => int16 // removed
  GenerationId      => int32
  ProtocolName      => String
  LeaderId          => String
  MemberId          => String // removed
  Members           => []JoinGroupResponseMember
                                     GroupInstanceId => String
                                     Metadata        => bytes
  MemberJoinResponseList => []JoinGroupResult // new
                                     MemberInfo     => JoinGroupResponseMember // new
                                     ErrorCode       => int16 // new

```

A new admin request shall be created for user to supply a list of `group.instance.id` to batch join the group:

AdminClient.java

```

public static AddMemberResult addMembersToGroup(String groupId, list<String> groupInstanceIdsToAdd,
AddMemberToGroupOptions options);

```

In the meantime, for better visibility for static members, we are also going to bump DescribeGroup request/response protocol to include `group.instance.id`:

```

DescribeGroupRequest => ThrottleTime Groups
  ThrottleTime      => int16
  Groups            => []DescribeGroups
                                     ErrorCode       => int16
                                     GroupId         => String
                                     GroupState      => String
                                     ProtocolType    => String
                                     ProtocolData    => int16
                                     Members         => []DescribedGroupMember
                                                         MemberId
=> String

GroupInstanceId => String // new

ClientId        => String

ClientHost      => String

MemberMetadata  => bytes

MemberAssignment => bytes

```

Compatibility, Deprecation, and Migration Plan

- User needs to upgrade broker to latest version to be able to use this new feature.

- *Since we are only introducing new admin API, the change should be backward compatible.*

Rejected Alternatives

We could trigger multiple join group requests at the same time without changing JoinGroup protocol. However, considering our change in [LeaveGroupRequest](#), it's hard to handle multiple responses within single admin client request. Changing the protocol to adapt to this change shall be more consistent.