CIP-16 Merge resource proto and transport proto
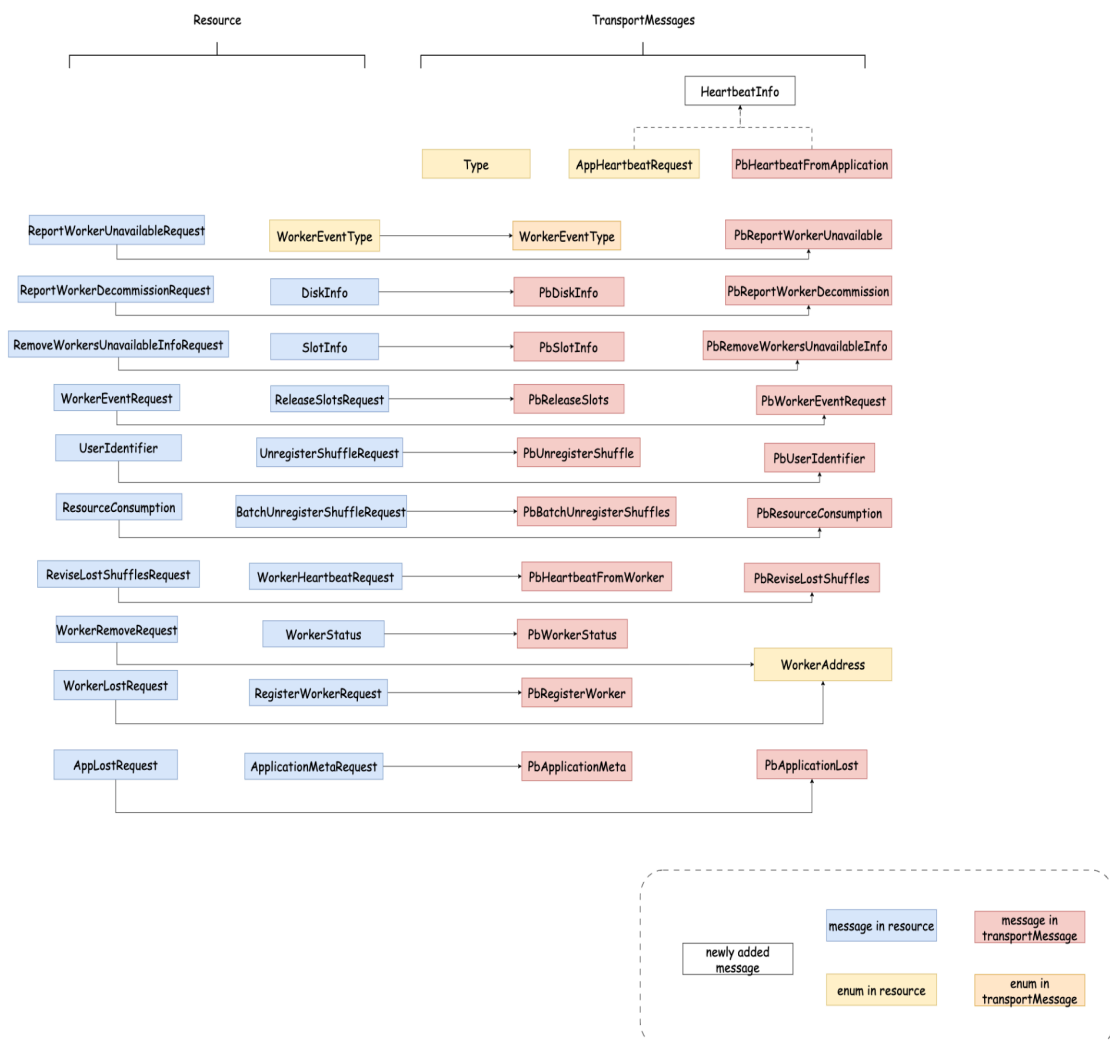
CIP-16 Merge resource proto and transport proto

Motivation

1. Merge resource.proto into TransportMessages.proto.
2. Using protobuf version 3.

Public Interfaces

The messages and enums need to be updated as shown in the picture below. Type, AppHeartRequest, and WorkerAddress should be moved from Resource to TransportMessage. The other messages and enums in Resource can be replaced as indicated in the picture.

1. Changes in Resource.proto

```
enum Type {
  Unknown = 0;
    …..
}
```

enum WorkerEventType(Replaced by WorkerEventType in TransportMessages)

message DiskInfo(Replaced by PbDiskInfo in TransportMessages)

message SlotInfo(Replaced by PbSlotInfo in TransportMessages)

message ReleaseSlotsRequest(Replaced by PbReleaseSlots in TransportMessages)

message UnregisterShuffleRequest(Replaced by PbUnregisterShuffle in TransportMessages)

message BatchUnregisterShuffleRequest(Replaced by PbBatchUnregisterShuffles in TransportMessages)

message WorkerLostRequest(Replaced by PbWorkerLost in TransportMessages)

message WorkerHeartbeatRequest(Replaced by PbHeartbeatFromWorker in TransportMessages)

message WorkerStatus(Replaced by PbWorkerStatus in TransportMessages)

message RegisterWorkerRequest(Replaced by PbRegisterWorker in TransportMessages)

message ReportWorkerUnavailableRequest(Replaced by PbReportWorkerUnavailable in TransportMessages)

message ReportWorkerDecommissionRequest(Replaced by PbReportWorkerDecommission in TransportMessages)

message RemoveWorkersUnavailableInfoRequest(Replaced by PbRemoveWorkersUnavailableInfo in TransportMessages)

message WorkerEventRequest(Replaced by PbWorkerEventRequest in TransportMessages)

```
message UserIdentifier(Replaced by PbUserIdentifier  in TransportMessages)


message ResourceConsumption(Replaced by PbResourceConsumption  in
TransportMessages)

message ReviseLostShufflesRequest(Replaced by PbReviseLostShuffles  in
TransportMessages)

message HeartbeatInfo {
  string appId = 1;
  int64 totalWritten = 2;
  int64 fileCount = 3;
  int64 shuffleCount = 4;
  map<string, int64> shuffleFallbackCounts = 5;
}

message AppHeartbeatRequest {
  string appId = 1;
  int64 time = 2;
  int64 totalWritten = 3;
  int64 fileCount = 4;
  int64 shuffleCount = 5;
  map<string, int64> shuffleFallbackCounts = 6;
  HeartbeatInfo heartbeatInfo = 7;
}


message WorkerRemoveRequest(Replaced by WorkerAddress)

message WorkerLostRequest(Replaced by WorkerAddress)

message AppLostRequest(Replaced by PbApplicationLost)

message ApplicationMetaRequest(Replaced by PbApplicationMeta)

enum Status {
  UNKNOWN = 0;
  OK = 1;
  INTERNAL_ERROR= 2;
}
```

2. Changes in TransportMessages.proto.The changes in TransportMessages.proto do not
include the merged Resource protobuf.

```
message PbHeartbeatFromApplication {
 string appId = 1;
  int64 totalWritten = 2;
  int64 fileCount = 3 ;
  string requestId = 4;
  PbWorkerInfo needCheckedWorkerList = 5;
  bool shouldResponse = 6;
  int64 shuffleCount = 7;
  map<string, int64> shuffleFallbackCounts = 8;
  HeartbeatInfo heartbeatInfo = 9;
}
```

Proposed Changes

1. **Upgrade protobuf version**: Convert Resource.proto from Protocol Buffers version 2 to
version 3.
2. **Remove required and default keyword**s: Since protobuf v3 does not use the required
keyword (all fields are optional by default), and default values are not supported, remove
them from Resource.proto.
3. **Apply the changes in Resource.proto**: Make the necessary changes in the
Resource.proto according to the above definition.
4. **Merge Resource.proto into TransportMessages.proto**: Incorporate the Resource
definitions from Resource.proto directly into TransportMessages.proto.
5. **Apply the changes in TransportMessages.proto**: Make the change mentioned above
in TransportMessages.proto

Compatibility, Deprecation, and Migration Plan

1. Old Resource.proto will be kept until one or two major releases.

2. We can safely remove the messages and enums from the old Resource.proto file after merging Resource.proto into TransportMessages.proto. This is because, as long as the fields have the same type and field number, they can still be serialized and deserialized correctly in a protobuf structure.

3. TransportMessages.proto can do the changes directly without compatibility issues.