"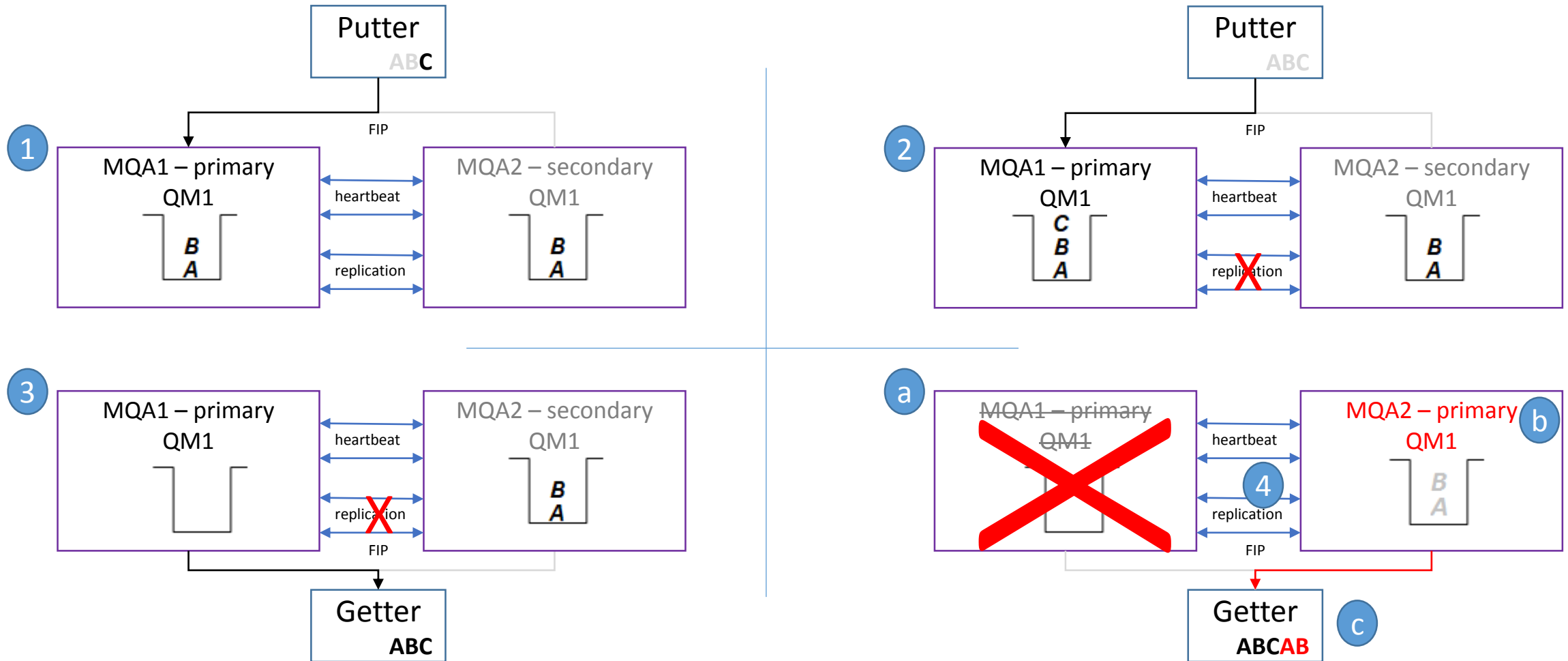In a given situation where a HA queue manager cannot determine which data is valid, the current design of the MQA HA solution favours availability rather than integrity. AIB wants the option to change this behaviour. To demonstrate the issue, the below diagram shows a scenario with 2 MQ Appliances configured to run in a HA Group:

1. Putter application sends messages to queue manager primary/active QM1 which is running on appliance MQA1. Messages are replicated to secondary MQA2 appliance.
2. Data replication link is lost. While not replicating, the Putter sends another message to QM1
3. The Getter application starts consuming the messages from QM1 on MQA1
4. MQA1 re-stablish data replication link with MQA2, but they don't agree on the data – they become "HA Status: Partitioned"
    a. MQA1 primary appliance goes down; the secondary/standby queue manager is requested to start on MQA2;
    b. The queue manager fails over to MQA2, regardless of its partitioned status
    c. The Getter application consumes aged/duplicated messages, compromising data integrity!

The example above was tested in our environment, demonstrating that automatically starting a queue manager when it is in "partitioned status" could bring integrity issues.
AIB wants to change this behaviour in a way that, in the case a HA queue manager is in "partitioned status" and cannot determine which data is valid, it will not start.