|  |
| --- |
| Software Engineering of Distributed Systems, KTH |
| Distributed Systems Advanced Homework 3 |
| Implementation of Reliable Broadcast Component, Unreliable Broadcast Component and Lazy Probabilistic Broadcast Component |

|  |
| --- |
| Shanbo Li and Sike Huang  2/19/2008 |

### Exercise 1. Modify Algorithm 3 such that it garbage collects the delivered set. Messages that no longer need to be maintained in the delivered set should be removed. Update your implementation of RB and describe the new algorithm in the report.

Answer:

We record the number of the same message which we received. If this number equals current alive nodes’ number then we affirmative that the initial source of this message has already crashed and all other nodes has forward this message to me. So I will never get this message again. So there is no need to store this message in deliver list to filter. The code for this algorithm is illustrated below:

|  |
| --- |
| Data structure: |
| private Map<SourceMessagePair, Integer> delivered; |

|  |
| --- |
| Algorithm: |
| public void handleBebDeliverEvent(BebDeliverEvent bebDeliverEvent) { |
| … |
| delivered.put(sourceMessagePair, new Integer(delivered.get(sourceMessagePair) + 1)); |
| … |
| if (delivered.get(sourceMessagePair).equals(new Integer(correct.size()))) { |
| delivered.remove(sourceMessagePair); |
| System.out.println(""); |
| System.out.println("Garbage collect delivered list, remove message \"" |
| + sourceMessagePair.getMessage() + "\" from Node " |
| + sourceMessagePair.getSource().getId()); |
| } |

//todo plus 1