Enhancements Report for Project 1
Distributed Systems (SDIS)

BACKUP

Our backup enhancement consists of telling a Peer that he should only store a Chunk if the Chunk's desired replication degree has not been met yet.

This is achieved by storing the real degree of replication of a Chunk in every Peer and by checking if it is lower than the degree that is specified on the PUTCHUNK message. If the degree is lower, the Chunk is stored. If it is not, the Chunk is not stored.

The enhancement allows the Peers to save disk space and not flood the multicast channels with STORED messages.

Enhancements Report for Project 1
Distributed Systems (SDIS)

RESTORE

For the restore function, we decided to enhance it by making a Peer which receives a GETCHUNK message send a Chunk directly to the other Peer asking for it, instead of making it go to the multicast channel.

We did this by creating a DatagramSocket for each Peer, that makes use of the vanilla restore functions, but instead of sending the packet contatning the chunk thorugh the Restore multicast, it sends it to the address and port of the GETCHUNK packet received through control.

This way, information that is useless to other peers does not circulate in the Multicast channel.

Enhancements Report for Project 1
Distributed Systems (SDIS)

RECLAIM

In the reclaim function, to make it safer, we made use of our backup protocol and implemented it inside reclaim. The backup functions we have know whether a Chunk has the desired replication degree or not. We made use of this information in our reclaim enhancement.

We start by telling a Peer to remove a Chunk from the database, but without actually doing it, and we call the Backup protocol on that Chunk, as if we were the initiator Peer. The Chunk is then stored on the other Peers and we get the information of whether the replication degree has been met or not. If it was, the chunk is safely removed from the Peer. If not, we proceed to the next Chunk.