Receiving Algorithm:

Step 1. Initialization.

Step 2. Call receiver state machine.

Step 3. Check for any EOS, DONE or data packets. If data packet is available, read it into application buffer.

Step 4. Strip data packet of its header and place in the appropriate place in the receiving buffer depending on the packet number in the header.

Step5. If EOS/DONE packet is available, send an acknowledgment for the segment on the feedback socket.

Step 6. If all the packets in a chunk have been completely received, send a high priority chunk completed packet on the completed socket.

Step 7. Write out the received chunk and go to step 1.

Step 8. Repeat steps 1 to 8 until all the chunks have been received.

3.4 Scalable Acknowledgment and Retransmission Scheme

A critical component of FOBS is the acknowledgment and retransmission mechanism. Although various schemes for acknowledgment of the received packets exist, most of them are not efficient for acknowledging a large number of packets transferred over high-latency, high-bandwidth networks. For example, the sliding window based acknowledgment scheme of TCP, which works well for low-bandwidth, low-latency networks, is not efficient for the backbone networks of the computational Grid.

For FOBS, we have designed an acknowledgment scheme that is highly efficient. Equally important, this acknowledgment scheme is scalable, in terms of the size of the data set,

the latency in the network and the size of the pipe. This is achieved by segmenting the acknowledgment for the chunk of data that is currently being transferred. Using the application buffer and treating it as circular, the FOBS design can claim to have an infinite acknowledgment window as explained in [8].

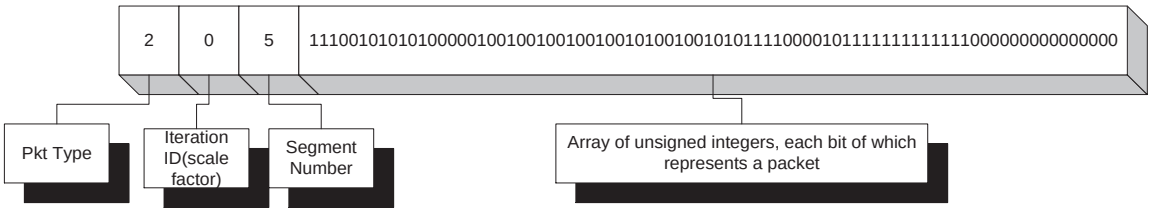
The first and the simplest acknowledgment scheme is the segmented bit acknowledgment scheme described below.

Segmented Bit Acknowledgment on TCP:

Acknowledgments transmitted over UDP are lost. TCP is a better choice for transmitting acknowledgments, as it is reliable and efficient for small transfers. This means that the acknowledgment packets will have to be as small as possible to fit into a single frame (usually 1500 bytes) so that the time taken to transfer the acknowledgment packet is kept to a minimum. Also, since transmitting the acknowledgment on TCP makes sure that each and every packet was received, the need for overlapping the acknowledgment packets is moot.

Specifically, in the segmented approach the chunk is split into segments of a fixed number of packets and an acknowledgment is sent for each segment separately. At the same time, there is an overhead in preparing and transmitting the acknowledgment packet on TCP, interrupting the receiver and sender from their primary tasks of receiving data packets and sending data packets respectively. In order to adjust for this problem, an optimal frequency (segment size) must be picked, depending on the latency and

bandwidth of the network. Another factor to be considered is the contention in the network.



```
ERROR: invalidrestore
OFFENDING COMMAND: restore

STACK:

-savelevel-
-savelevel-
```