CRC or Cyclic Redundancy Check is an error control method used in Data Link Layer, where a checksum is computed and appended to the Data Packet being sent across the channel.

Please explain briefly, in your own words.

**Data Link Layer almost always put the CRC in a trailer rather than in a header of the data packet. Why?**

A Cyclic Redundancy Check (CRC) detects errors in data transmitted over a network by adding a series of check values to the end of a block of data. The check values are the remainder of a polynomial division of their contents. When data is received, the calculation is performed again with the expectation that the remainder will be the same. If this is not the case, then the data block has been corrupted.

Usually, the CRC is appended to the end of a block of data. This is because the remainder of the original block of data plus check value attached when bitwise XOR Long-Division is computed is by definition zero. This simplifies things for the receiver because all they now have to do is perform the polynomial division with the correct divisor and check that the remainder is indeed zero. Were the CRC appended to the beginning of the block of data, then the receiver would have to ensure a match in the remainders of the transmitted remainder and the remainder which is recomputed on the receiver side. (“Computation of cyclic redundancy checks”, 2014). Another consideration, is that since the computations are occurring from the left of the block of data to the right, having the CRC at the tail end of the block allows all of the bits in the data block to be processed (A painless guide to CRC error detection algorithms index V3.00.”, 1996).

References

“A painless guide to CRC error detection algorithms index V3.00.” (1996). Repairfaq. Retrieved from <http://www.repairfaq.org/filipg/LINK/F_crc_v3.html>

“Computation of cyclic redundancy checks.” (2014). Wikipedia. Retrieved from <http://en.wikipedia.org/wiki/Computation_of_cyclic_redundancy_checks>