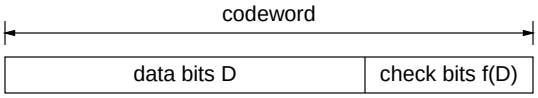
Reliability

Adam M. Costello <amc@cs.berkeley.edu> 1999-Oct-13-Wed

A reliable channel is error-free (data out equals data in).
An unreliable channel can introduce transmission errors.

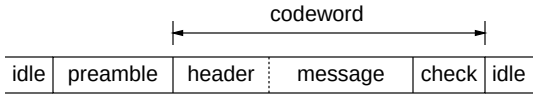
To bulid a reliable channel out of an unreliable channel, the receiver must be able to detect errors. General solution: The sender sends only codewords, and the receiver accepts only codewords. Only a very small fraction of possible strings are valid codewords, so the probability of accepting an erroneous string is very small.



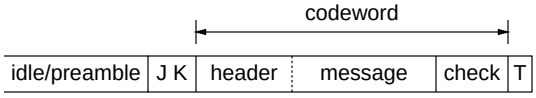
The receiver needs to see codeword boundaries. For a reliable channel on top of a packet service, this is easy: each codeword is a packet (for example, a TCP segment is a codeword carried in an IP datagram). But over a bit pipe (like a physical layer) we must first provide framing (a way of delimiting the codewords).

Ethernet and FDDI are not reliable channels, but are examples of framing and error detection. They do not recover from errors, but do avoid passing erroneous data up to higher layers.

The Ethernet physical layer has three symbols (0, 1, idle):



The FDDI physical layer has 25 symbols, 16 used for groups of four bits, the rest used for framing:



If the physical layer provides only two symbols (0 and 1), then delimiting frames is trickier. Usually a special sequence marks the start and end, and the codeword is modified (reversibly) so that it does not contain the special sequence. A common data link protocol called HDLC marks frame boundaries with 01111110. Since the codeword might contain 01111110, the sender inserts a 0 after every 11111, and the receiver removes every 0 after 11111. This is called bit-stuffing.

There are two approaches to recovering from errors:

Error correction

Some codeword schemes allow the receiver to map the erroneous string to the ‘‘nearest’’ valid codeword, which is with very high probability the one that was sent.

Requires more check bits, so less channel capacity is available for user data.

Does not work for errors affecting too many bits.

Recovers from an error very quickly, so is good for real-time applications (like voice).

Requires no back channel, so good for storage media (like compact discs).

Retransmission

The receiver detects errors and informs the sender of what has (or has not) arrived, so the sender can retransmit frames as necessary.

Requires fewer check bits.

Works for all kinds of errors (except malicious attacks).

Takes at least one round-trip time to recover.

A retransmission protocol is correct if it delivers every message exactly once.

The efficiency of a retransmission protocol is:

$$\frac{\text{message delivery rate}}{\text{frame rate of underlying unreliable channel}}$$

The efficiency may depend on the data rate, delay, and error rate of the channel, and on packet lengths.