**Give any two reasons why networks might use Error Correcting Codes instead of Error Detection and Retransmission**

When networks communicate data from sender to receiver, errors can happen. This is especially true over unreliable connections. When data does becomes corrupted, there are ways to handle them: Error Detection and Retransmission or Error Correcting Codes.

**Difference Between the Two Methods**

Error Detection and Retransmission is a simple idea. Check codes are passed from the sender to receiver along with the data. When the receiver gets the data, the check data is compared to what has been sent. If there is a match, the data is good. If there is a mismatch, the receiver can signal and error and ask the sender to retransmit the corrupted packet (Costa, 2008).

Error Correcting Codes go one step further. Not only can they allow the receiver to detect the error, but they enable the receiver to fix the corrupted packet without need for retransmission (Costa, 2008).

**Which to Choose and Why**

There are two reasons why networks would used Error Correcting Codes instead of Error Detection and Retransmission:

* The first reason is obvious. The bandwidth needed to send back requests for retransmissions and the retransmissions themselves will be higher than just sending the data forward a single time.

* The second reason is Error correcting Codes are suitable for simplex communication transmissions (“Error detection and correction”, 2014). That is, the receiver does not need to send error information to the sender nor does the receiver need to ask for retransmission. As such, the sender can simply transmit the data, knowing that if there are errors, they can be handled by the receiver.

References

Costa, P. (2008). Computer Networks: Medium Access Control Sublayer. Retrieved from<http://research.microsoft.com/en-us/um/people/pcosta/cn_slides/cn_04.pdf>

“Error detection and correction” (2014). Wikipedia. Retrieved from<http://en.wikipedia.org/wiki/Error_detection_and_correction#Error_detection_schemes>