Data link layer

From Wikipedia, the free encyclopedia

In the seven-layer OSI model of computer networking, the **data link layer** is **layer 2**. In TCP/IP reference model, it corresponds to, or is part of the link layer.

The data link layer is the protocol layer that transfers data between adjacent network nodes in a wide area network or between nodes on the same local area network segment.[1] The data link layer provides the functional and procedural means to transfer data between network entities and might provide the means to detect and possibly correct errors that may occur in the physical layer. Examples of data link protocols are Ethernet for local area networks (multi-node), the Point-to-Point Protocol (PPP), HDLC and ADCCP for point-to-point (dual-node) connections.

The data link layer is concerned with local delivery of frames between devices on the same LAN. Data-link frames, as these protocol data units are called, do not cross the boundaries of a local network. Inter-network routing and global addressing are higher layer functions, allowing data-link protocols to focus on local delivery, addressing, and media arbitration. In this way, the data link layer is analogous to a neighborhood traffic cop; it endeavors to arbitrate between parties contending for access to a medium, without concern for their ultimate destination.

When devices attempt to use a medium simultaneously, frame collisions occur. Data-link protocols specify how devices detect and recover from such collisions, and may provide mechanisms to reduce or prevent them.

Contents

Overview

Delivery of frames by layer-2 devices is effected through the use of unambiguous hardware addresses. A frame's header contains source and destination addresses that indicate which device originated the frame and which device is expected to receive and process it. In contrast to the hierarchical and routable addresses of the

network layer, layer-2 addresses are flat, meaning that no part of the address can be used to identify the logical or physical group to which the address belongs.

The data link thus provides data transfer across the physical link. That transfer can be reliable or unreliable; many data-link protocols do not have acknowledgments of successful frame reception and acceptance, and some data-link protocols might not even have any form of checksum to check for transmission errors. In those cases, higher-level protocols must provide flow control, error checking, and acknowledgments and retransmission.

In some networks, such as IEEE 802 local area networks, the data link layer is described in more detail with media access control (MAC) and logical link control (LLC) sublayers; this means that the IEEE 802.2 LLC protocol can be used with all of the IEEE 802 MAC layers, such as Ethernet, token ring, IEEE 802.11, etc., as well as with some non-802 MAC layers such as FDDI. Other data-link-layer protocols, such as HDLC, are specified to include both sublayers, although some other protocols, such as Cisco HDLC, use HDLC's low-level framing as a MAC layer in combination with a different LLC layer. In the ITU-T G.hn standard, which provides a way to create a high-speed (up to 1 Gigabit/s) local area network using existing home wiring (power lines, phone lines and coaxial cables), the data link layer is divided into three sub-layers (application protocol convergence, logical link control and medium access control).

Within the semantics of the OSI network architecture, the data-link-layer protocols respond to service requests from the network layer and they perform their function by issuing service requests to the physical layer.

Sublayers of the data link layer

The data link layer has two sublayers: *logical link control* (LLC) and *media access control* (MAC).[2]

Logical link control sublayer

The uppermost sublayer, LLC, multiplexes protocols running atop the data link layer, and optionally provides flow control, acknowledgment, and error notification. The LLC provides addressing and control of the data link. It specifies which mechanisms are to be used for addressing stations over the transmission medium and for controlling the data exchanged between the originator and recipient machines.

Media access control sublayer

MAC may refer to the sublayer that determines who is allowed to access the media at any one time (e.g. CSMA/CD). Other times it refers to a frame structure delivered based on MAC addresses inside.

There are generally two forms of media access control: distributed and centralized. Both of these may be compared to communication between people. In a network made up of people speaking, i.e. a conversation, we look for clues from our fellow talkers to see if any of them appear to be about to speak. If two people speak at the same time, they will each pause a random amount of time and then attempt to speak again, effectively establishing a long and elaborate game of saying "no, you first".

The Media Access Control sublayer also determines where one frame of data ends and the next one starts – frame synchronization. There are four means of frame synchronization: time based, character counting, byte stuffing and bit stuffing.

- The *time based* approach simply puts a specified amount of time between frames. The major drawback of this is that new gaps can be introduced or old gaps can be lost due to external influences.

- *Character counting* simply notes the count of remaining characters in the frame's header. This method, however, is easily disturbed if this field gets faulty in some way, thus making it hard to keep up synchronization.
- *Byte stuffing* precedes the frame with a special byte sequence such as DLE STX and succeeds it with DLE ETX. Appearances of DLE (byte value 0x10) have to be escaped with another DLE. The start and stop marks are detected at the receiver and removed as well as the inserted DLE characters.
- Similarly, *bit stuffing* replaces these start and end marks with flag consisting of a special bit pattern (e.g. a 0, six 1 bits and a 0). Occurrences of this bit pattern in the data to be transmitted are avoided by inserting a bit. To use the example where the flag is 01111110, a 0 is inserted after 5 consecutive 1's in the data stream. The flags and the inserted 0's are removed at the receiving end. This makes for arbitrary long frames and easy synchronization for the recipient. Note that this stuffed bit is added even if the following data bit is 0, which could not be mistaken for a sync sequence, so that the receiver can unambiguously distinguish stuffed bits from normal bits.

Data link layer services

- Encapsulation of network layer data packets into frames
- Frame synchronization
- Logical link control (LLC) sublayer:
    - Error control (automatic repeat request,ARQ), in addition to ARQ provided by some transport-layer protocols, to forward error correction (FEC) techniques provided on the physical layer, and to error-detection and packet canceling provided at all layers, including the network layer. Data-link-layer error control (i.e. retransmission of erroneous packets) is provided in wireless networks and V.42 telephone network modems, but not in LAN protocols such as Ethernet, since bit errors are so uncommon in short wires. In that case, only error detection and canceling of erroneous packets are provided.
    - Flow control, in addition to the one provided on the transport layer. Data-link-layer error control is not used in LAN protocols such as Ethernet, but in modems and wireless networks.
- Media access control (MAC) sublayer:
    - Multiple access protocols for channel-access control, for example CSMA/CD protocols for collision detection and re-transmission in Ethernet bus networks and hub networks, or the CSMA/CA protocol for collision avoidance in wireless networks.
    - Physical addressing (MAC addressing)
    - LAN switching (packet switching) including MAC filtering and spanning tree protocol
    - Data packet queuing or scheduling
    - Store-and-forward switching or cut-through switching
    - Quality of Service (QoS) control
    - Virtual LANs (VLAN)

Error detection and correction

- Besides framing, data link layers also include mechanisms to detect and even recover from transmission errors.
- For a receiver to detect transmission error, the sender must add redundant information (in the form of bits) as an error detection code to the frame sent.
  - When the receiver obtains a frame with an error detection code it recomputes it and verifies whether the received error detection code matches the computed error detection code. If they match the frame is considered to be valid.
- An error detection code can be defined as a function that computes the r (amount of redundant bits) corresponding to each string of N total number of bits.
- The simplest error detection code is the Parity bit.
  - The parity bit allows a receiver to detect transmission errors that have affected a single bit among the transmitted N+r bits. If there are two or more bits in error, the receiver may not be able to detect the transmission error.

A simple example of how this works by using meta-data. Say we want to transmit the word 'HELLO'. To keep things simple we will change each letter in the alphabet as its position in the alphabet. Thus, the letter A is coded as 1, B as 2, and so on:

H	E	L	L	O
8	5	12	12	15

Adding the digits $8 + 5 + 12 + 12 + 15 = 52$. Then we add $5 + 2 = 7$ to get the meta-data. We then transmit:

8	5	12	12	15	7

If there are no errors, the receiver will get 8 5 12 12 15 7. The receiver knows that the last number received is the error-detecting meta-data and that all data before is the message. It can recalculate the above math and if it comes to the same meta-data answer, it can be concluded that the data was received without errors. If it receives something like 7 5 12 12 15 it can run the check by: $7 + 5 + 12 + 12 + 15 = 51$ and $5 + 1 = 6$, Since 6 does not equal 7 the receiver can discard the received data as defective.

Protocol examples

- Address Resolution Protocol (ARP)
- ARCnet
- ATM
- Cisco Discovery Protocol (CDP)
- Controller Area Network (CAN)
- Econet
- Ethernet
- Ethernet Automatic Protection Switching (EAPS)

- Fiber Distributed Data Interface (FDDI)
- Frame Relay
- High-Level Data Link Control (HDLC)
- IEEE 802.2 (provides LLC functions to IEEE 802 MAC layers)
- IEEE 802.11 wireless LAN
- LattisNet
- Link Access Procedures, D channel (LAPD)
- LocalTalk
- Multiprotocol Label Switching (MPLS)
- Nortel Discovery Protocol (NDP)
- OpenFlow (SDN)
- Split multi-link trunking (SMLT)
- Point-to-Point Protocol (PPP)
- Serial Line Internet Protocol (SLIP) (obsolete)
- IEEE 802.1aq - Shortest Path Bridging
- Spanning Tree Protocol
- StarLan
- Token ring
- Unidirectional Link Detection (UDLD)
- and most forms of serial communication.

Software interfaces

Data link layer implementations may be in software only, simulating a network interface.

Relation to TCP/IP model

In the TCP/IP (Internet Protocol Suite) model, OSI's data link layer (in addition to other components) is contained within the TCP/IP's lowest layer, the link layer. The Internet Protocol's link layer only concerns itself with hardware issues to the point of obtaining hardware addresses for locating hosts on a physical network link and transmitting data frames onto the link. Thus, the link layer is broader in scope and encompasses all methods that affect the local link, which is the group of connections that are limited in scope to other nodes on the local access network.

The TCP/IP model is not a top/down comprehensive design reference for networks. It was formulated for the purpose of illustrating the logical groups and scopes of functions needed in the design of the suite of internetworking protocols of TCP/IP, as needed for the operation of the Internet. In general, direct or strict comparisons of the OSI and TCP/IP models should be avoided, because the layering in TCP/IP is not a principal design criterion and in general considered to be "harmful" (RFC 3439). In particular, TCP/IP does not dictate a strict hierarchical sequence of encapsulation requirements, as is attributed to OSI protocols.

See also

- ODI
- NDIS
- SANA-II – Standard Amiga Networking Architecture, version 2

References

1. ^ "What is layer 2, and Why Should You Care?"
   (http://wayback.archive.org/web/20100218075030/http://www.accel-networks.com/blog/2009/09/what-is-layer-2-and-why-should-you-care.html). accel-networks.com. Archived from the original (http://www.accel-networks.com/blog/2009/09/what-is-layer-2-and-why-should-you-care.html) on 2010-02-18. Retrieved 2009-09-29.
2. ^ Regis J. Bates and Donald W. Gregory (2007). *Voice & data communications handbook* (http://books.google.com/books?id=eq1kRHdyXSUC&pg=PA45) (5th ed.). McGraw-Hill Professional. p. 45. ISBN 978-0-07-226335-0.

- S. Tanenbaum, Andrew (2005). *Computer Networks* (4th ed.). 482,F.I.E., Patparganj, Delhi 110 092: Dorling Kindersley(India)Pvt. Ltd.,licenses of Pearson Education in South Asia. ISBN 81-7758-165-1.
- Odom, Wendel (2013). *CCENT/CCNA ICND1 100-101, CCENT Official cert guide*. Paul Boger, cisco press. ISBN 1-58714-385-2.

External links

- DataLink layer simulation, written in C# (http://www.codeproject.com/Articles/57072/DataLink-Simulator)
- DataLink Layer, Part 2: Error Detection and Correction (http://www.cs.gmu.edu/~huangyih/656/error.pdf)

Retrieved from "http://en.wikipedia.org/w/index.php?title=Data_link_layer&oldid=623899204"

Categories: OSI protocols │ Link protocols