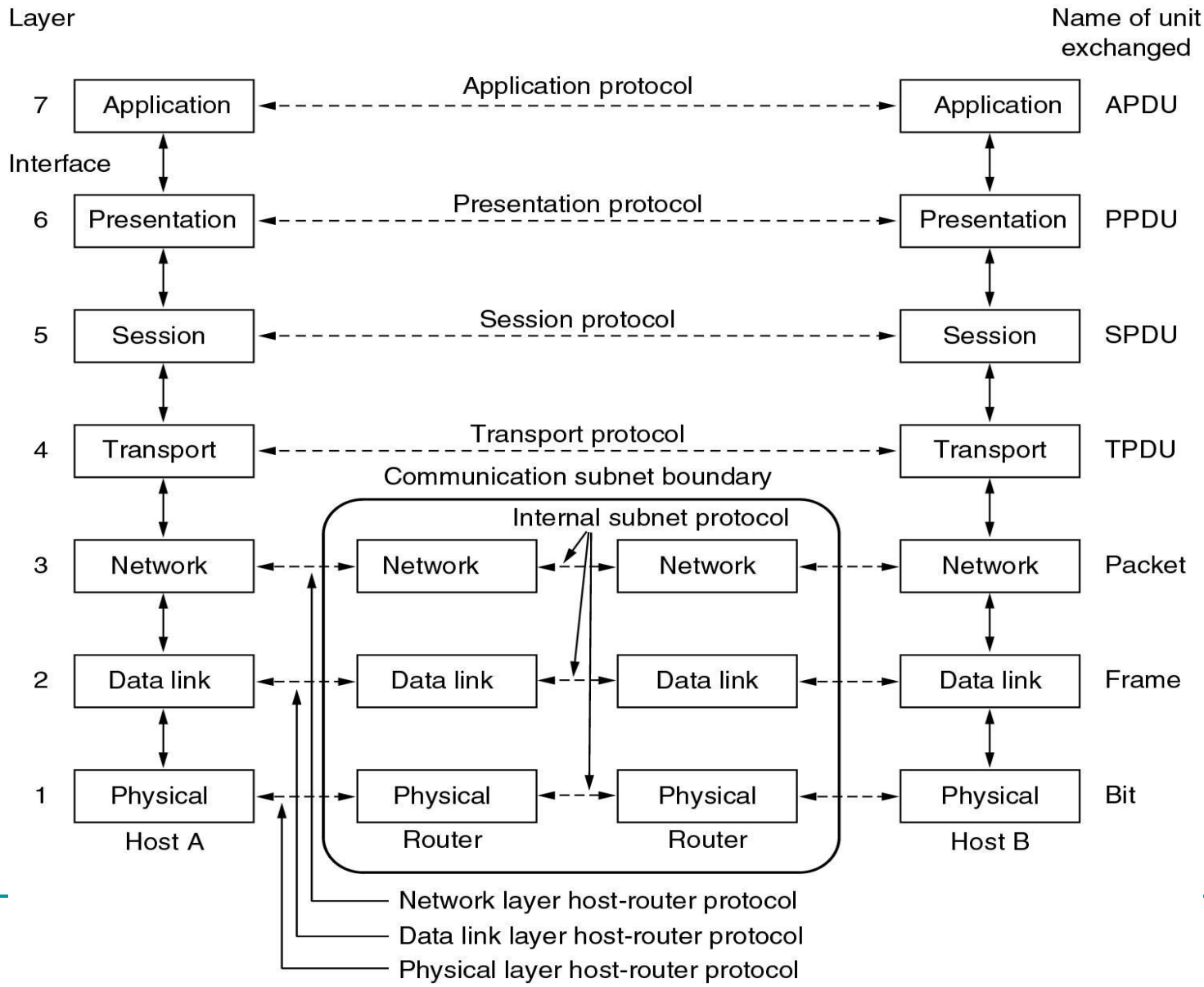
Other Layers

Internet Technologies
COMP90007

Reading

- Not direct reading of any section is required for this lecture

Recall: the OSI Model



Two More Layers from the OSI Model

- **Presentation Layer**
- **Session Layer**

- They did not see a ***distinct*** use or labelling through out the years for the Internet

- Lets see if they do not exist on todays Internet at all

Presentation Layer

- Formatting related issues
- For example:
  - Given complex data, such as a data structure
  - You may want to serialize the data to send across
  - Create an XML representation for example
  - This should not be the duty of the application
  - But it is commonly done in applications today…

Presentation Layer Contd.

- Encryption/Decryption:
  - Should be done in Presentation Layer as well
  - Although it is commonly done at Application layer
- Compression:
  - We have discussed for example in Multimedia data that decompression can be done
  - It should be done at this layer rather than in applications

Why not have an Explicit Presentation Layer?

- Todays Internet does not have this clear distinction
- A key reason is many Presentation layer list of things to do that we discussed is considered to be application specific
- Thus, Application Layer and Presentation is not explicitly separated for today's Internet
- Nevertheless, this is not good form in SE

How about the Session Layer?

- Common services of this layer
  - Authentication
  - Session management
    - Monitoring connections
    - Disconnect if not used
    - Reconnect if needed
  - These are also seen as a part of the Application layer duties today depending on different requirements of applications of todays' Internet
  - A few are done at Transport Layer (e.g., SCTP)
  - Especially session management in a simple client server architecture was seen as trivial

Are These two Layers Really not There Anywhere Else?

- If you have a need to do compression, session management, etc then think before implementing
- You should create a better software design by creating your own little Session Layer as a separate layer in your software architecture
- Similarly most software architects do create a ***middleware layer*** for their software and other similar softwares → ***Missing layers***

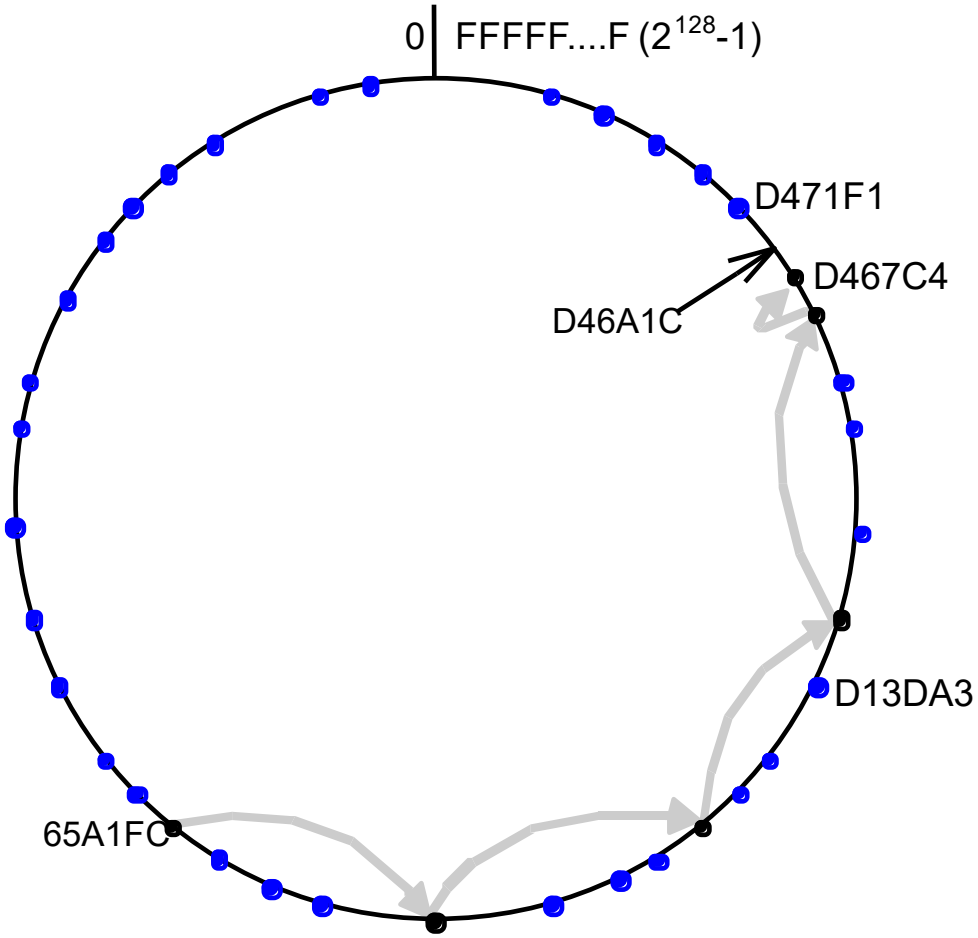
Case Study for Today: P2P

- Separation of real application related tasks from the tasks of the missing layers for networked systems
- We got a glimpse of this with Cloud computing presentation as well

Case Study Contd

- Client server systems dominated the Internet for a while
- They are simple to implement
- However other are more complex, **Peer-to-Peer (P2P) way of implementing things exist**
- **P2P systems are good on certain fronts**
- A P2P system:
  - Does not have clients and server but just peers
  - Does not have a central point of control
  - Advantages:
    - No central point of control or failure
    - Potential to scale without a bottleneck
  - Disadvantages:
    - **Harder to develop applications on such a dynamic platform where PCs come and go**
    - …

A Simple Way to Connect Peers



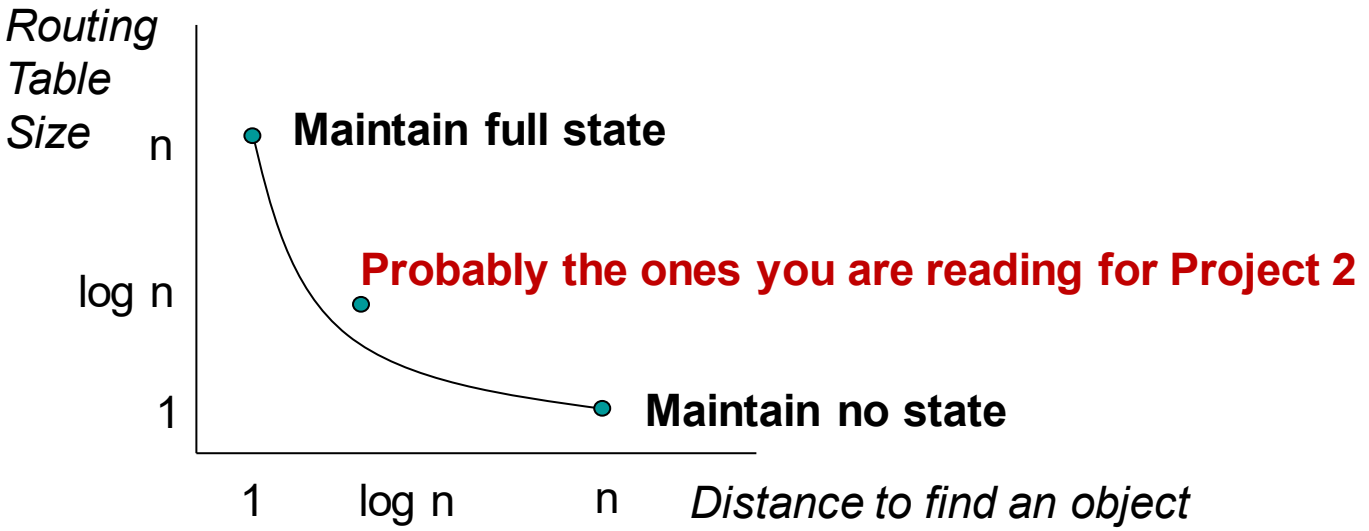
Not all peers can know about all others in a large system

Thus: New methods are invented (Related to your Project 2)

The black dots depict live nodes/PCs. The blue ones are files. The address space is considered as circular: node 0 is adjacent to node (2^{128}-1). The diagram illustrates the routing from node 65A1FC to D46A1C.

For a file browser app where should this algorithm exist?

Methods that Do P2P Routing



Building A P2P File System, A Game, Etc over the Net