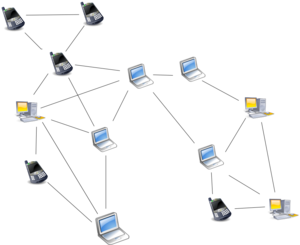
**Peer-To-Peer (P2P) Architecture**

Peer to peer is a commonly used software architecture where each computer or node has equal privileges and capabilities. Unlike the Client/server style where client nodes can only access data from the server nodes and the server nodes can control which clients to be served or not, in P2P, each node is both a client and a server at the same time and can request for information as well as send information to other nodes. <https://www.techopedia.com/definition/454/peer-to-peer-architecture-p2p-architecture>

In P2P, the workload is distributed between the peers, and all peers contribute and consume resources within the network without the need for a centralized server.

Some common uses of the P2P architecture are:

* File Sharing
* Instant Messaging
* Voice Communication
* Distributed computing

Some examples of P2P architecture:

* Napster - it was shut down in 2001
* Bit Torrent - popular P2P file-sharing protocol
* Skype - it used to use proprietary hybrid P2P protocol, now uses client-server model after Microsoft’s acquisition
* Bitcoin - P2P cryptocurrency without a central monetary authority
* Gnutella
* Freenet

P2P networks are divided into structured and unstructured systems.

**Unstructured peer-to-peer networks** do not have a particular structure on the network, but rather are formed by nodes that randomly form connections to each other. Since no structure is required, these are easy to build and can grow very large. This causes delays in searches since search queries have to be applied over the entire network.

**Structured peer-to-peer networks** are organized into a specific topology, and the protocol ensures that any node can efficiently search the network for a file/resource. The most common type of structured P2P networks implement a [distributed hash table](https://en.wikipedia.org/wiki/Distributed_hash_table) (DHT), in which a hashing variable is used to assign ownership of each file to a particular peer.

Hybrid architecture models are a combination of P2P and client server models where the central server is used to help peers find other peers. A common example of hybrid is Spotify.

**Advantages of Peer to Peer:**

* The main advantage of peer to peer network is that it is easier to set up
* In peer-to-peer networks all nodes are act as server as well as client therefore no need of dedicated server.
* The peer to peer network is less expensive.

**Disadvantages of Peer to Peer:**

* A computer can be accessed anytime.
* Network security has to be applied to each computer separately.
* Backup has to be performed on each computer separately.
* No centralized server is available to manage and control the access of data.
* Users have to use separate passwords on each computer in the network.
* As with most network systems, unsecure and unsigned codes may allow remote access to files on a victim's computer or even compromise the entire network
* Virus, Trojan, Worm, Key logger program attachments.

References:

<http://infosecawareness.in/peer-to-peer-network>

<https://en.wikipedia.org/wiki/Peer-to-peer>

<https://www.student.cs.uwaterloo.ca/~cs446/1171/Arch_Design_Activity/Peer2Peer.pdf>

**Pipe and Filter Architecture**

The pipe and filter is an Architectural style which is made up of a stream of a data and flows in a stream through pipes and filters. In this pattern, there are many components, which are referred to as filters, and connectors between the filters that are called pipes. Each filter is responsible for applying a function to the given data; this is known as filtering. Filters can work asynchronously. The final output is given to the consumer, known as a sink.

The filters act as individual processes that can be run simultaneously and one filter does not control the operation of the others. This style is used for large projects that can be broken down into smaller individual processes.

This kind of style is used in some UNIX programs where the output of one program can be linked to the input of another program.

They are also used in compilers where the consecutive filters perform lexical analysis, parsing, semantic analysis and code generation to pass on to the pipe.

**Advantages of Pipe and Filter:**

* Any changes can be made very easily to the software as we can always introduce new filters and steps
* Simple: Allows easy understanding of the structure and functioning of the system
* Easy to maintain and reuse: New filters can be added with pipe connectors or removed easily.
* Concurrent Execution: Each filter can operate and execute simultaneously with others.

**Disadvantages of Pipe and Filter:**

* If the pipe allows only for one kind of data type (a byte or character) the filters will have to do some parsing and un-parsing. This will complicate things and slow down the whole process
* Interaction between different pipes may not be very efficient as in this style filters only communicate by input/output functions
* No Filter cooperation since they are independent
* Filters may have to wait to receive all data from the pipes and thus there buffers may over flow if not considered for.

References:

<http://www.dossier-andreas.net/software_architecture/pipe_and_filter.html> <https://www.student.cs.uwaterloo.ca/~cs446/1171/Arch_Design_Activity/PipeFilter.pdf>