IPC Methods in Linux:

Signals are the most basic form of IPC in Linux. Their primary use is to notify processes of changes in states or events that occur within the kernel or other processes. They can be used for communication and synchronization, but there are better alternatives for both.

Pipes provide a mechanism for one process to stream data to another. One end of the pipe is the read-end which is associated with a file-descriptor that can only be read, and the other end is the write-end which is associated with a file descriptor that can only be written. Pipes can be setup and used only between processes that have a parent-child relationship, and the communication is only one-way. Pipes are generally used to implement Producer-Consumer design amongst processes.

Named pipes (or FIFO) are a variant of pipes that allow communication between processes that are not related to each other (no parent-child relationship required). Additionally, the FIFO is actually a file on the disk that multiple processes can open, read, and write to. So, you would want to use a FIFO when you want to perform something similar to pipe operations of a file.

Message Queues are set up so that one process writes a message packet to the message queue and exits, then another process can access the message packet from the same message queue at a later point. The advantage of message queues over pipes are that the sender processes do not have to wait for the receiver processes to connect.

Shared memory allows one process to share a region of memory with at least one other process. This allows two or more processes to share data more efficiently with minimal kernel intervention. Good when working with large sets of data that need to be shared between processes.

Semaphores are locking mechanisms most commonly used to provide synchronization when processes are sharing resources. Ideal when dealing with multiple processes running through a critical section.

Unix Domain Sockets are sockets that can be used between processes on the same Unix system. Sockets themselves are like a pipe with two-way communications rather than one-way. Usually when Unix sockets are used, they're used with server and client programs, similar to Internet sockets.

MapReduce:

MapReduce is a programming model that allows for processing large sets of data via parallel computing. MapReduce consists of two functions that achieve this goal. The first function is "map", which takes the set of tuples and converts it into a series of intermediate key/value pairs. The second function is "reduce", which takes the output from the map function and merges the pairs in order to produce a smaller set of key/value pairs. MapReduce allows for high scalability and makes it easier to perform operations on large sets of data. It essentially splits the data and performs some operation on the data to create intermediate pairs that can then be grouped by the reduce function.

Hadoop:

Hadoop is an open-source MapReduce framework. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs. It gives users the ability to store and process huge amounts of data quickly. It also provides high scalability, high flexibility, and has a low cost. Hadoop is widely used in today’s age because there’s a massive amount of data out there to work with, and the amount is only increasing. Hadoop provides an accessible way for developers to manage large sets of data at a relatively low cost.