10/04/2022:

Today's lesson included a discussion of distributed hash tables. It is possible to store and retrieve data linked to a key over a network of peer nodes using a type of distributed database known as distributed hash tables. Considering that these peer nodes can join or leave the network at any time, distributed hash tables are a type of distributed database. The nodes connect with one another to disseminate and store data throughout the network; no central entity controls the nodes. Distributed hash tables have the capacity to tolerate failure and bounce back swiftly from it when key/value pairs are duplicated. The peers' capacity to share data amongst themselves stands in stark contrast to the Blockchain model, in which every node maintains a copy of the whole ledger.

A routing layer is required because DHT nodes can't keep all the data, making it possible for any node to find the node that has a specific key. Because DHT nodes cannot keep all of the data, this is necessary. The way the routing table works and how it is updated as nodes enter and leave the network is a basic way that the different DHT algorithms differ from one another. DHT nodes often only store a portion of the whole routing data that is present in the network. This shows that for a node to function, routing information only has to be sent to a small number of peers when it joins the network.

Finding or storing a key/value pair necessitates visiting numerous nodes because each one only stores a portion of the routing table. Due to the fact that each node only holds a piece of the table, this is required. The number of nodes that must be reached is proportionate to the value because each node stores a different quantity of routing information. O(log n), where n is the total number of nodes in the network, is the notation used to describe how difficult a typical lookup is. One million nodes in a network, for instance, require only the location of twenty nodes. Some DHT algorithms provide for the potential of a tradeoff in which the quantity of router state can be increased in exchange for a greater reduction in the worst-case lookup cost. The size of the routing table would probably be increased in a production DHT using a WAN, which would further reduce the worst-case lookup by at least half. This decision would probably cut the worst-case lookup in half because route tables are rather small.

10/06/2022:

P2P systems were the focus of the first part of the lecture. The peer toolbox is a new effort that was born out of the real-time BrainBain BCI study. After creating the FieldTrip buffer to transmit and buffer EEG and MEG data over the network, we realized that we could utilize a buffer in other situations, including distributed computing, with a small adjustment. After we had already put the FieldTrip buffer into place, we came to this discovery.

Each peer in the network is made up of a MATLAB session as its component. A mex file is being executed within the context of that MATLAB session, which in turn triggers the background execution of several theads. The buffer is a little TCP server that can accept and store MATLAB variables that have been passed from one peer to another. The fact that it can receive and store these variables makes it the most important thread. The input variables of a calculation that has to be run on this peer or the output variables of a computation that was run on another peer are both sent to the buffer for processing. It is also possible to use either of these types of variables to get the outcomes of a calculation done on a different peer. Sending and receiving UDP multicast packets are handled by a separate thread in the background. These are really brief messages that are sent around the network to inform other peers of the peer's status and to announce its presence. The computers connected to the local network receive these multicast packets in their entirety.

The outcome is that every peer on that network can find every other peer. As a result, you are not needed to include a list of all peers in a configuration file. Assume you have three peers on the computers of your coworkers who are still at work but have already departed for the evening, in addition to one peer on your own computer. There is a distinct group of peers for each of these coworkers. Your coworker is in charge of the circumstance and has the authority to give orders to the other employees.