<u>**CS 7260 - Internet Architecture & Protocols**</u>

<u>**Scribe – 8/31**</u>

A packet, once it arrives at a router, follows a forwarding path to a destination port. The various methods through which it determines the destination address of this port and reaches it is what we are going to cover in this scribe.

1. Exact Match Look-up:

Exact Match Look-up is the simplest form of a database query. It searches the entire database to find the exact output port address match corresponding to a MAC address given as input.

MAC Addresses	**Output Port**
A1	P1
A2	P2
A3	P3
.	.
.	.
.	.
.	.

Fully Associated Cache

Figure 1

Message should be forwarded to the frame associated to the address e. g. P1 (frame) for address A1.

Static Set:

We take an assorted list of MAC addresses and do a binary search but the time taken to search the set in this case is **log n.** But if it seems that log n is taking too long, then we can also do pipelining. We put two nodes each in multiple memory modules and then do pipelining.

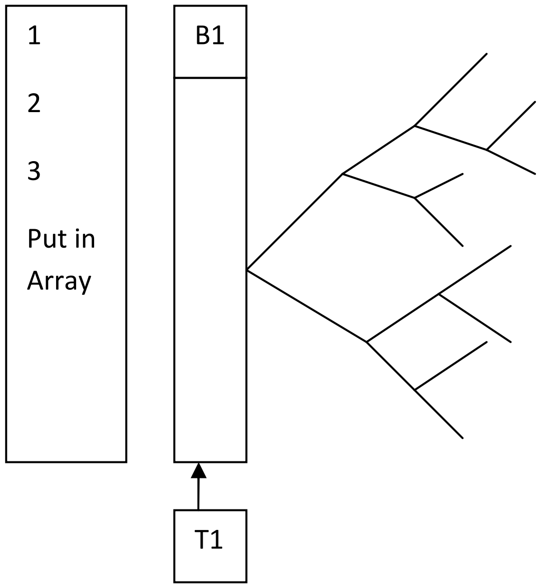

Figure 2

Search B1 at T1, B2 at T+1 and so on.

In this way, we are going to have one output every cycle. Also, on an average, every look-up takes only one memory access.

2. IP Look-up (Prefix Match Look-ups):

Since the cost of computation is dominated by memory accesses, the simplest measure of lookup speed is the worst-case number of memory accesses. Also, old school routers had limited memory allocated for IP Prefixes mostly till 100000. Since the IP Prefixes had a limited space, it resulted in variable-length prefixes since they make more efficient use of the address space.

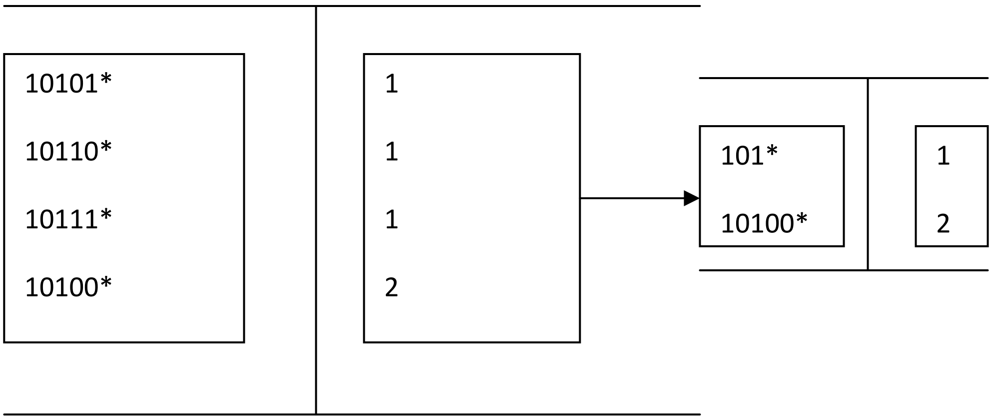

Figure 3

Longest Prefix Match:

We have a table, as shown below, containing a bunch of prefixes and a list of output port numbers.

0*	P1
00*	P2
010*	P3
10*	P4
11*	P5

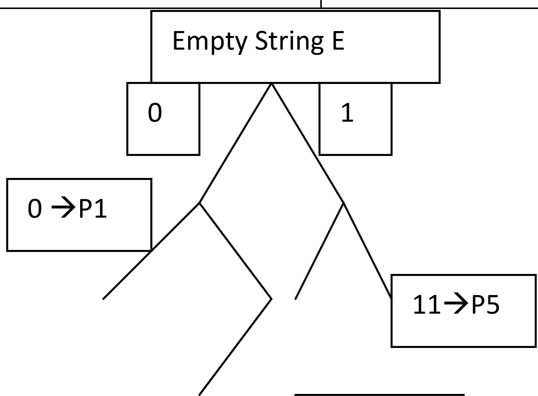

Figure 4

This special type of binary tree is called a **trie.** Mapping and best map is given as we down the tree.

Jump(>1) bits at a time:

The number of bits we jump at a time is known as the **stride length**.

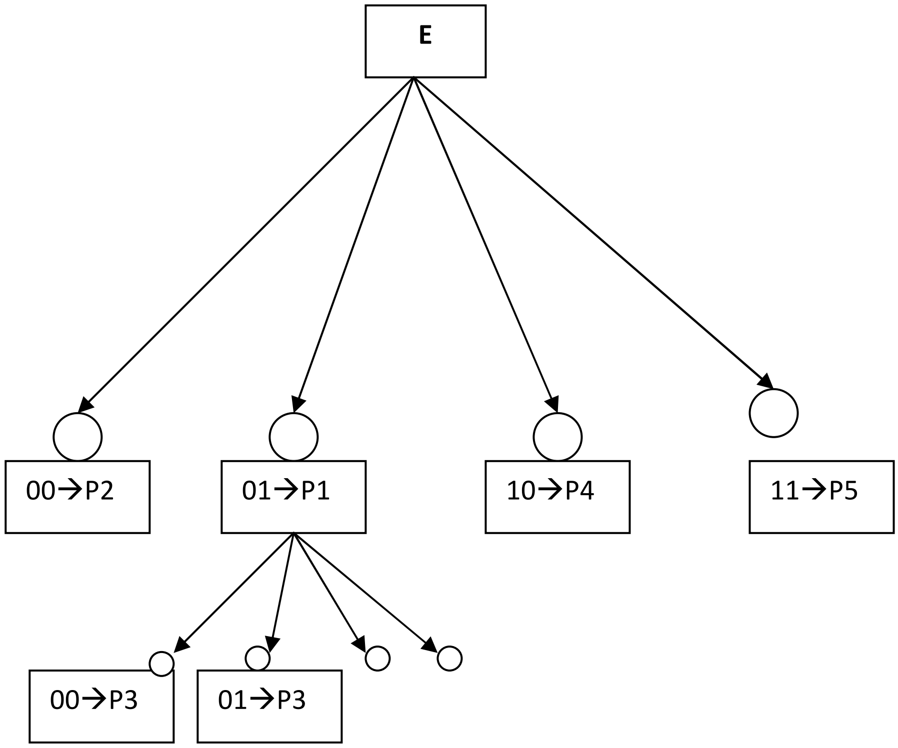

Figure 5

There are two things we care about:

a) Amount of memory being used

b) Minimize the number of memory accesses

Suppose we want to restrict memory accesses to only 2, then we need to increase the memory space.

Dynamic Programming is the solution to strike a balance between space and number of memory accesses problem.

--X--

References: Other than the lecture, I have referred the text book - Network Algorithmics by *George Varghese*

Manika Andotra