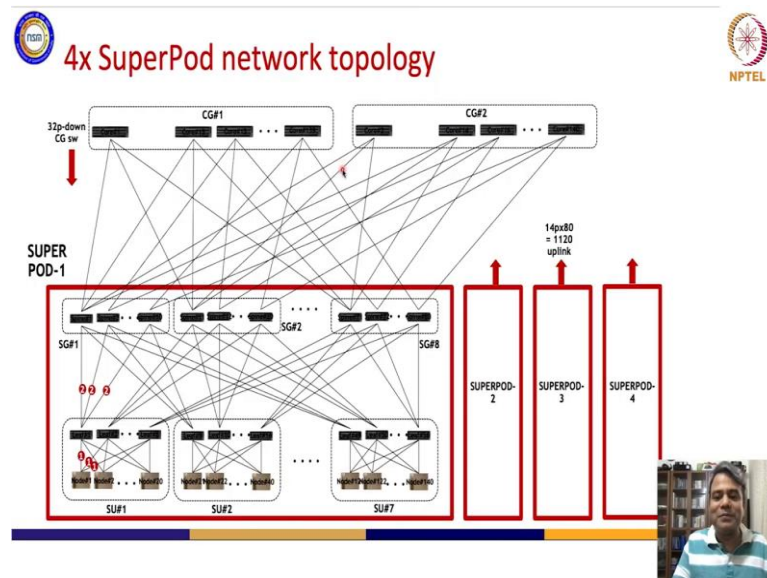


Applied Accelerated Artificial Intelligence  
Prof. Ashrut Ambastha  
School of Computer Science and Engineering  
Indian Institute of Technology, Madras

Lecture - 17  
Design Principles for Building High Performance Clusters  
Networking Fundamentals Part - 4

(Refer Slide Time: 00:15)



(Refer Slide Time: 00:16)

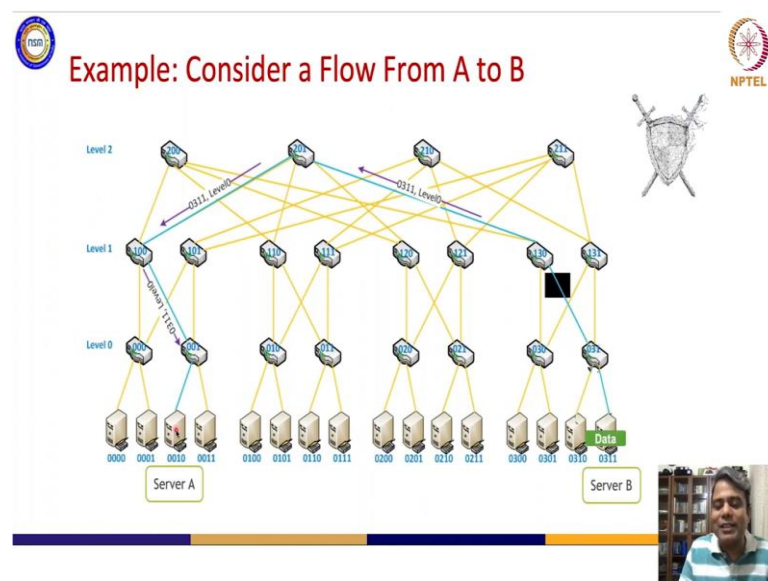


Ok. Now comes a question. So, let us talk about some more advancements you know very obvious question that comes into picture after seeing that that complicated large you know crisscross kind of diagram is; what if one link fails? I am running a parallel job and suddenly one link fails what will happen to all my mapping and all the things that I did right?

I tried to do so much of optimization I created a network in a particular manner, I made sure that my programming was done or programming was utilizing the library in such a manner that everything kind of came together into one big, you know data center as a compute unit right.

But what if one of the link fails? Everything breaks apart right. So, therefore, you need to have some more intelligence in the network so that it can be self-healing. If there is a link failure it should not be catastrophic, it should not break down your entire job, it should not; it should not be that you have to run your job from scratch.

(Refer Slide Time: 01:28)



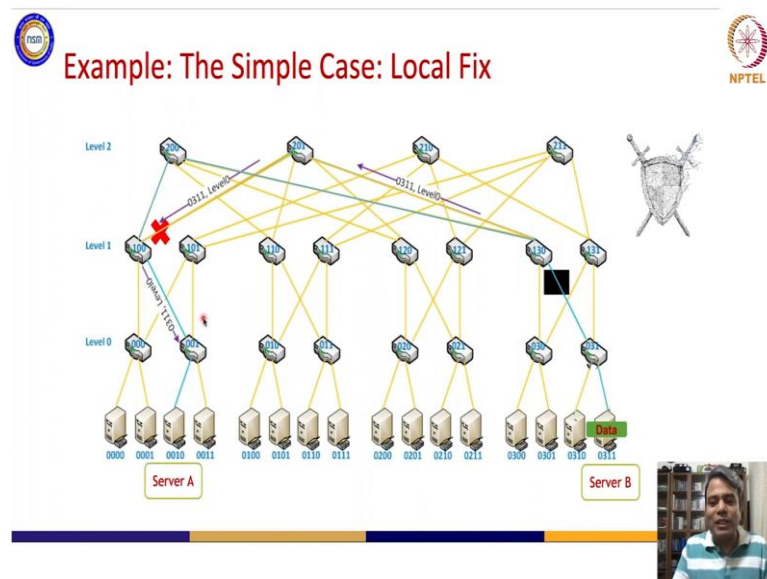
So, let us look at some properties of that fat tree network. Again I am showing you a same you know 3 tier network, lowest level is all the compute elements, then level 0 is nothing but 1st level of switching, then 2nd level of switching and 3rd level of switching.

When server A, which is 0010 wants to send data to server B it will send data packet to the first level of switch, the switch will say hey ok, it is meant for such certain server B

and I know that to reach that server I need to send it out from a certain port. It will send the data from that certain port. The next level of switch will do the same thing, it will switch. When the data reaches the top level it will switch down.

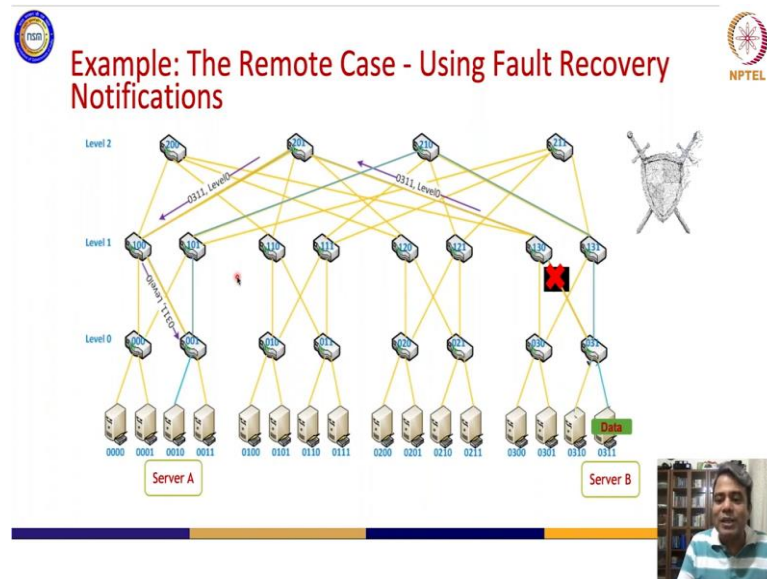
Now, when the data comes down, it will go to the first level switch again and it will reach the destination.

(Refer Slide Time: 02:24)



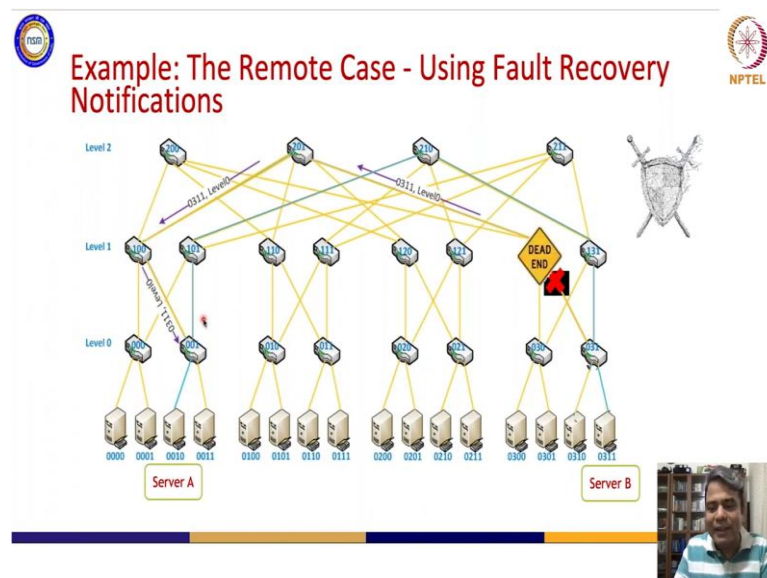
What if one of the uplink breaks? Over here let us say one of the uplink breaks, the switch contained a table to route the data through this particular link ok. So, when the data arrive to that particular switch it said “hey, I do not have that link available anymore,” networks are intelligent you know they have been intelligent for quite some time that they can actually make a decision. And if one link is broken and if there is a parallel link available, which is there it will route the packet through that parallel link ok. And data will again reach the destination.

(Refer Slide Time: 03:10)



There is a certain characteristic of this kind of tree topology is that if you have a down link failure ok.

(Refer Slide Time: 03:20)

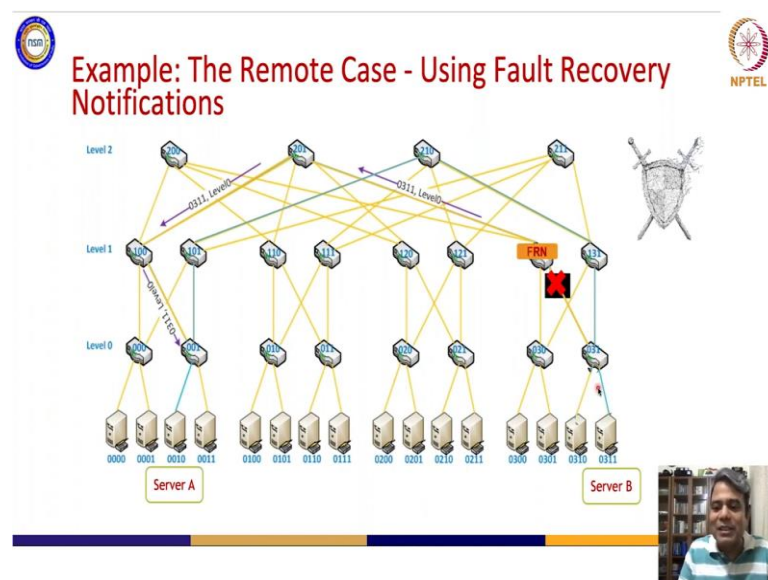


So, as long as it is in uplink failure you had multiple parallel paths, but if you had a downlink failure, there is a dead end because there is only now one way to reach down. You do not have for packets going down, you do not have a parallel path because this endpoint does not have two ports, in which one will connect to switch 1 and other will connect to switch 2.

Generally, in enterprise systems you have a dual port adapter that is fine right. But in HPC systems you do not do that, because when you put a dual port adapter you actually reduce for a certain amount of switching element, you automatically reduce the size of the cluster by half.

The main aim is to make a very large cluster, not reduce it for resiliency. So, better to have redundancy in such a manner, that if you have these kind of failures then something intelligent should happen and that is what happens in these high performance switches right.

(Refer Slide Time: 04:10)



Once a switch sees that there is a link failure it sends out between the switches a failure and routing notification, to all the switches which are in the critical path of the delivery port of a particular switch. Because of this when data packet goes from server A to the first switch, it automatically reroutes it to a different switch and a different path uplink path altogether ok. What it does is so you would say “hey, why do I need to do this? Why cannot I have you know data was not delivered let my server send out a NAK.”

And let it send out you know let there not be any acknowledgement because finally all these are reliably connected servers right. So, the transport mechanism is reliable if there was data delivery that did not happen especially when it was an RDMA packet, you have to have upper layer protocols which make sure that data has to be delivered correctly otherwise there will be memory corruption, which is right; 100 percent right, which is

what used to happen in all the other networks right, the network does not do anything it is the end points which actually talk about resiliency.


And if data is packet is not delivered it will say “hey, I did not get it please send it back.” So, data needs to be sent all the way back, but because of these intelligent networks now, because if there was a failure in any of the links, because the networks themselves are intelligent, you do not actually have to go all the way up to the software layer to request for a new packet.

What does it mean in terms of latency? Rather than doing software level retry, which could be of the order of 10s of milliseconds we are able to do this route change in 10s of or maybe 100s of nanoseconds ok.


So, therefore, in a large cluster even link failures do not result in degraded performance because networks themselves are intelligent enough to communicate amongst themselves and create multiple tables so that if there are failures or link breakages packet will still reach. And because these networks do have N number of paths, N multiple paths between any source and destination, because it is a non-blocking network and I will come to that part as well. We will now talk about the blocking factors in a topology.

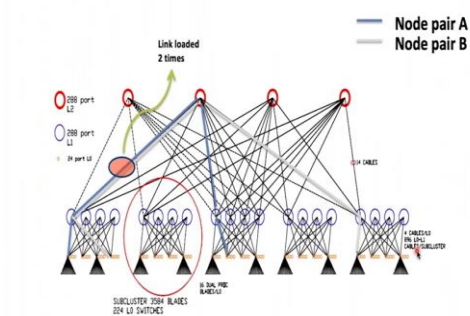
So, we talked about in network computing as one advanced feature. We talked about resilient network as an advanced feature.

(Refer Slide Time: 07:05)




## Over Subscription and Bisection in Static Routing





- Divide cluster into two subset of nodes (random)
- Subset-1 = Tx and subset-2 = Rx
- Trace data-flow path between each communicating pair
- Increment “subscription bucket” if any segment in flow path overlaps
- Repeat n times for statistical accuracy a



Now, look at; let us look at some other one more thing maybe ok. So, showing you the same diagram, showing you the same kind of diagram a three-level fat tree ok which has got again in the bottom most I will show. As you can see in the bottom most you have nodes connected, the orange circles over here are nothing but the level 1 switch. Blue circles are level 2 and then you have red circles which are level 3 switches ok.

Now, think what is over subscription and bisection. So now, in a large cluster, let us say I have a cluster of thousand nodes ok. Now, if I have a cluster of 1000 servers, there can be communication, think about communication when one node wants to talk to there any of the remaining nodes. So, there are 999 nodes that one particular node can talk to, right.

So, there can be a you know there can be 999 ways in which traffic can go from one particular server to a destination server. Similarly, if the second node wants to talk to all the others, there will be again n into you know 999 ways. So, the total number of ways that people or servers can talk to each other is given by  $n * n - 1$ . For a 1000 node cluster, it is  $1000 * 999$ , which is a huge number. Do you think there are those many numbers, there is do you think there are 1 million cables in this? No, there are not ok.

So, definitely everybody does not talk to everybody simultaneously, but you can have any random pair of nodes which will start talking to each other right, depending on how your problem is distributed. So, a 1000 node cluster you can make two sets of 500 communicating pairs and you can then trace the route every pair will take, it is very much probable that many segments may get loaded by communication from multiple pairs.

So now, even though I have a 100 Gbps link, I am taking an example; even if I have a 100 Gbps injection into the fabric and my fabric is fully non-blocking; by non-blocking I mean my lowest level switch have got 20 ports connected down to the servers and 20 ports connected up to other switches. So, ideally 20 servers can get 20 uplinks right. So, it is not blocking, no communication is blocking. If I had 20 ports connected down and only 10 ports connected up, in any all to all scenario I would have actually blocked my uplink two times.

But in a non blocking network ideally everything should work and everything is you know everybody should get that 100 Gbps of bandwidth, but it does not happen. Because



communication pattern in a cluster can be random and because communication pattern in a cluster can be random, there can be segments which will be loaded twice. Because they will be loaded twice, the pairs of communicator which are sharing that particular segment will get only 50 Gbps, if my injection was 100. Basically, they will get half of the bandwidth.

(Refer Slide Time: 10:57)



This is the nature of any statically routed network and this is very important, because now if you look at ok we have not gone into topologies, but there are various topologies right. I showed examples, I named factory we talked about various kind of torus, we talked about dragonfly hypercube, there are various mathematical way in which you can connect multiple switches to create a topology and create a to create a cluster. But all of them are not alike.

In a statically routed network, if you start doing what I did in the previous slide, that is to create communicating pairs and then see how many of those communicating pairs are able to get full bandwidth versus half bandwidth. Half bandwidth because some of the links might be double loaded, versus one-third the bandwidth because there can be some links which are triple loaded ok. So, x axis on my graph over here represents the path sharing, means communicating pair for which there is a particular segment in my network which is loaded 3 times or 4 times or 5 times.

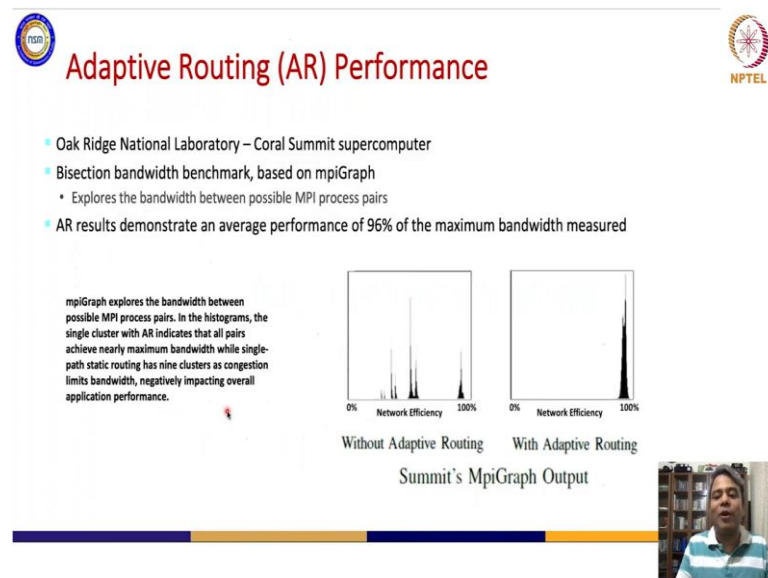


So, x shows the path sharing ok and y shows the number of communicating pairs in my cluster. So, this graph actually plots how a statically routed network will look like for a 2000 node cluster that is a simulation done. So, we have taken and it is actually reality as well because it was done on a real system. We had a 2000 node cluster, where you do artificial blocking versus different kind of topologies and you can see that what is an ideal network.

Ideal network is a network where all communicating pair will be at 1. So, a graph which is a delta function at 1 represents an ideal network that means, I had a 1000 node cluster, I divided into two subsets of 500 and everybody had all their paths only single shared. So, everybody got the same bandwidth as what they were injecting into the fabric. So, delta function at 1 is an ideal network.

Obviously, you do not get an ideal network, but the graphs which are approaching the delta function are the better networks ok. So, these are some of the ways in which you determine a network, characteristic and then comes another advanced topic of adaptive routing.

(Refer Slide Time: 13:37)



I had a fat tree; I had. So, the number of links going into the fabric and the number of links in the fabric, I mean they are if I will always have an ideal communication pattern where everybody should get their bandwidth, because I physically created a non-blocking network. But it is because it was statically routed you know some of them got

paths shared. This is where the concept of adaptive routing comes into picture, where the switches themselves can see whether their uplink paths are getting blocked.

How do they see it? They see it by looking at the data packets which are queued for their uplink path, if that queue starts increasing so that it is not able to sustain the full bandwidth the switch knows that that particular path is trying to carry data worth more than a single injection server. Therefore, it can actually now look at the queues of various out ports and it can try to distribute the uplink packets into these out ports, dynamically so that every communicating pair of servers in a fabric will get full bandwidth.

And this is what is achieved by adaptive routing. So, all the new advanced networks, they contain adaptive routing as an intelligence built into the switch data plane itself. It is not a Ethernet routing algorithm of OSPF or BGP, which is like you know shortest path network or something which our ECMP for load balancing, no.

Because Ethernet it is the control plane that does all the routing, but in high performance network if you had a control plane doing this by the time that the control plane is notified that there is a congestion. And by the time that the control plane is able to make a decision the packets will be dropped, because you are looking at such high bandwidth networks right.

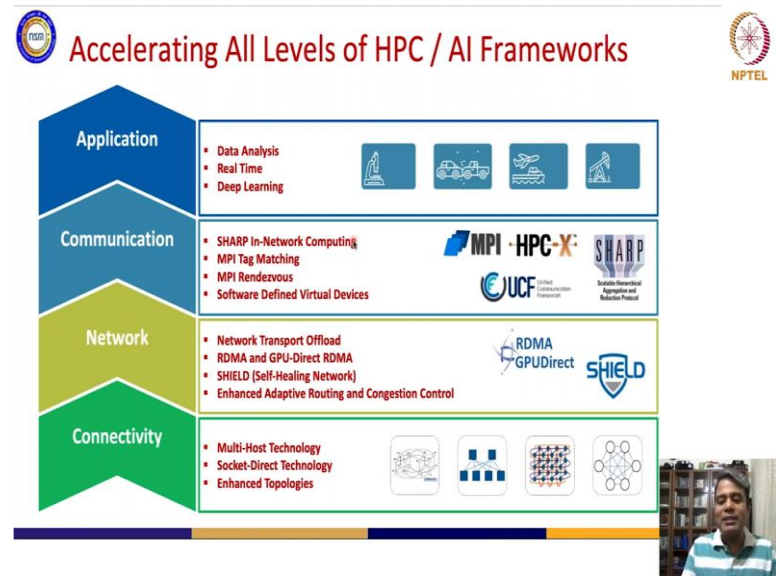
Therefore, you need to have these routing decision making done in the minimum time possible by looking at the data plane itself, which is what all high performance networks do. This example this is a paper from the Oak Ridge coral summit supercomputer, which is a very large computer and where various adaptive routing algorithms were tried and this part shows the network efficiency when it was statically routed that is without adaptive routing.

You can see this this histograms show nothing but the blocking factor, means these many nodes actually were able to achieve near 100 percentage of the bandwidth, there were many nodes which were sharing path. So, they had only 50 percent. Then some of them had 33 percent and so on and so forth right.

And when adaptive routing algorithm and adaptive routing based hardware was enabled on that particular computer, the same computer was able to show near 100 percent throughput, 94 percentage to be precise for all the communicating pair. So, this is again

96 percentage sorry to be precise, for all the communicating pairs. So, this is again one more advanced feature for high performance networks ok.

(Refer Slide Time: 17:19)

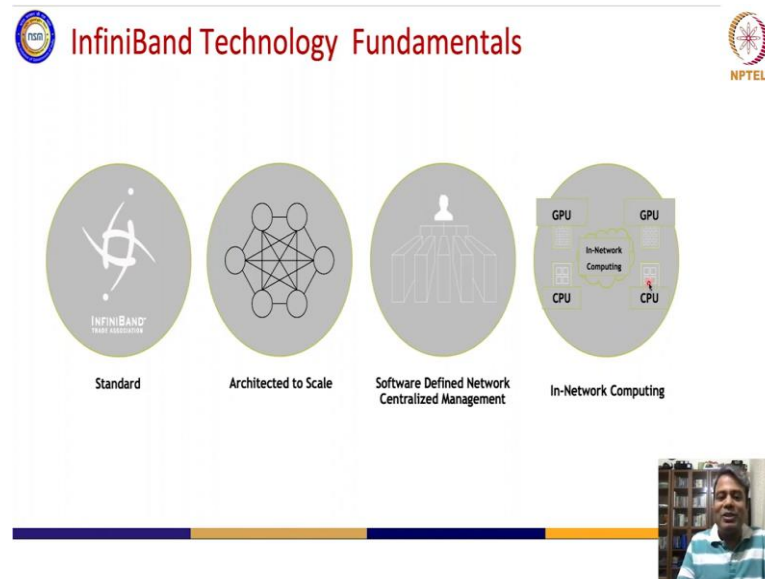


So, I think we talked about lot of points, I really do not want to be you know overloading with many, there are many many other things. What I want to summarize is that we need to accelerate all the levels of HPC and AI frameworks to get a supercomputer working at its optimum performance, from clustering and networking point of view.

You need to look at enhanced topologies, you need to look at you know I did not cover many of these. So, I will not talk about it, but we did talk about the transport offloads these are the basics right. RDMA as a transport offload, GPU direct as a transport offload, these are basics. Then large networks self healing ok.

Congestion in the network enhanced adaptive routing. On the communication framework itself in network computing where the network elements are taking part into computation, they are not only being you know communication engines, but also compute engines. So, all these enhancements are fundamentals of today's high performance networks.

(Refer Slide Time: 18:40)



We look at some specific technologies, as I showed in the previous session, we talked about various high performance networking technologies. But we propagate InfiniBand as a technology because; right now it is one of the only existing technology for high performance networks. As I showed in that graph there is not many proprietary networks, people are not making proprietary networks anymore, because this is a standards based network.

Just like Ethernet is a IEEE standard RFP defined, similarly InfiniBand is IBTA defined a technology standard and there are tons of companies which are part of this consortium. NVIDIA leads it, because right now NVIDIA creates all the elements of this network. It is standards based so it is backwards and forwards compatible, is architected to scale.

We talked about scaling and the various technologies that are put into it so that it can actually scale up to 1000 of nodes. It is fully software defined and centralized managed, as in there is no every switch in that network does not have any control plane processor, like unlike a ethernet switch where you can log into a switch and you go and say; “hey, please run this protocol on it, please do a link aggregation, please do a OSPF, please do a BGP pairing” none of those switches do that.

It is all managed by a central entity called Subnet manager, which discovers the entire fabric, which puts the routing table into each element and so on. So, it is like the epitome of software defined networking.

And then finally, advancements like doing in network computing for both CPUs and GPUs are part of this ok. So, I think with that I will conclude my session; I hope it was informative and you would be interested in looking more into details for folks who are interested in the networking domain right.

Down from you know right up from fundamentals of electrical engineering with you know the 5 layer to fundamentals of networking with the link and network and transport to the maths of graph theory and topologies to you know doing compute inside the network and looking at advanced algorithm for making networks more efficient.

So, again, thank you for joining.