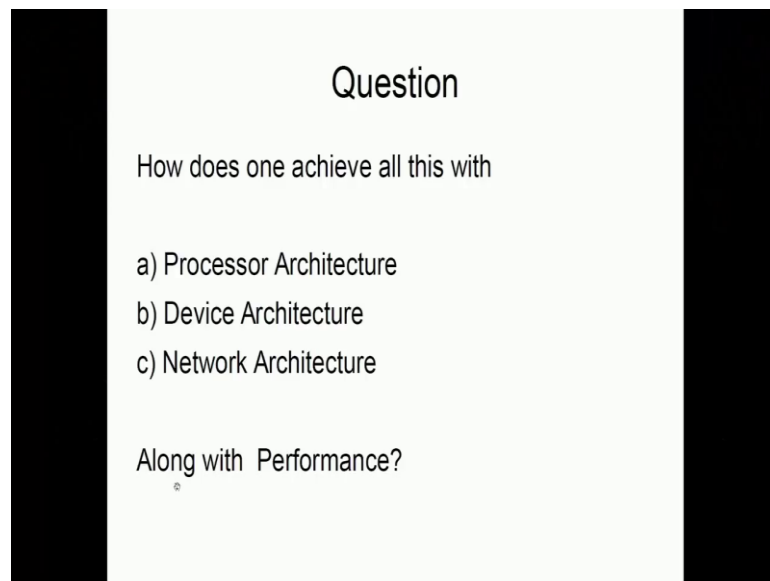


**Information Security - II**  
**Prof. V. Kamakoti**  
**Department of Computer Science and Engineering**  
**Indian Institute of Technology, Madras**

**Lecture – 41**  
**Network Processor Vs General Purpose Processor - Week 8**

(Refer Slide Time: 00:09)



**Question**

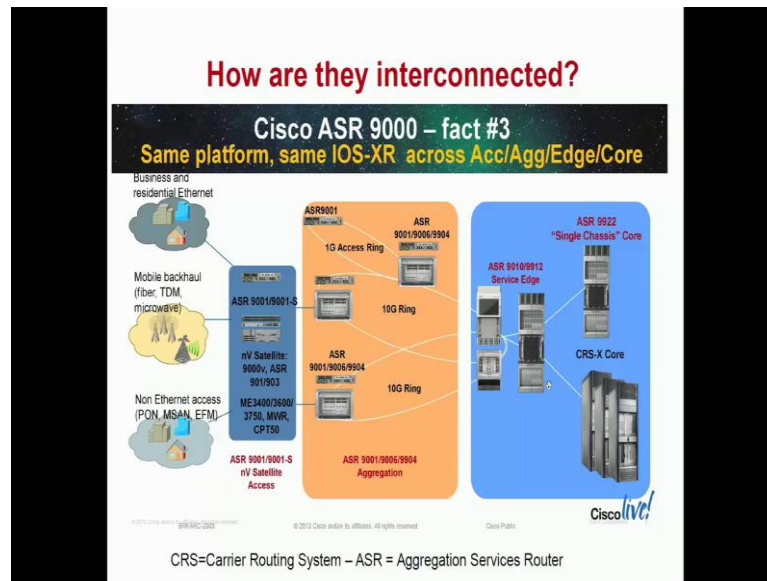
How does one achieve all this with

- a) Processor Architecture
- b) Device Architecture
- c) Network Architecture

Along with Performance?

So, to answer this question.

(Refer Slide Time: 00:14)



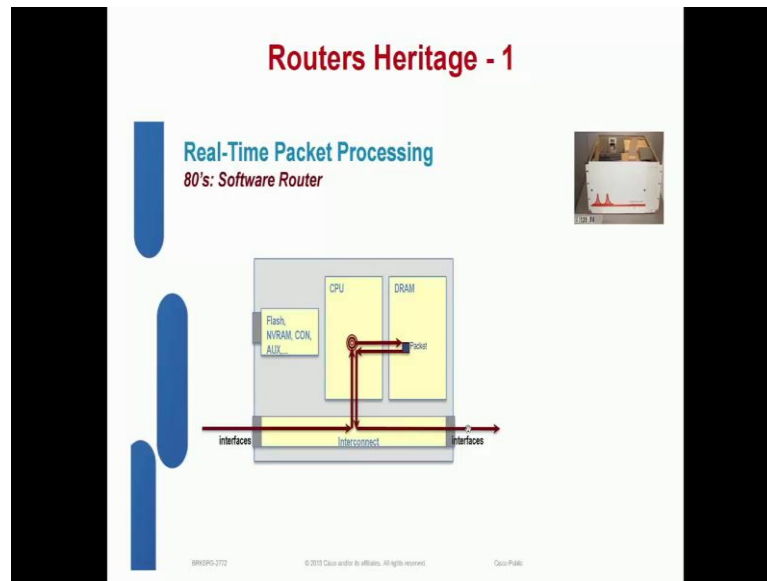
Let us now look at routers. In practice, how they are design, what is there architecture, and once we understand will also look at evolution of a network processor. And finally, after we look at the evolution, we will see the architecture of a network processor, and once we look at the architecture of a network processor then we will see look at how to improve the performance of a network processor along with architecture, that is the ok. I hope this introduction will help you, now understand the slides that are going to come. So, let us look at how the internet is organized you start off with a LAN network that is something like this a business and residential Ethernet that starts something like this and then these network connect to something known as aggregation services router – ASRs, and they connect to bigger aggregation services router.

So, if you look at this these routers are slightly small they connect to routers that are slightly larger and they have more excess capacity among than these routers, and then these connect to co-routers, so this is known as access edge core. And these routers increase in size they consume more power and so on, say for example, and the architecture of these routers are also different; essentially as you the routers increase in size you have more amount of parallelism and pipelining that will come into picture. In essence, any network processor architecture has number of pipe lines and operating in parallel. The advantage is the why we can do that is because packet size more or less in

many protocols are defined because of that you can actually design processors which can cater to that specific packages. So, for example, a word length you know is 32 bits and you know the ip address and IPv 4 at least is 32 bits you know similarly IPV 6 is 128 bits. So, they fix the packet sizes because of that that you are able to say byte bound reason all that. So, you remember one of the things ATM had 53 bytes, but in your architecture class, you always talk about byte boundaries byte boundaries. Now, 53 bytes how did they land up with 53 bytes I mean it was more political, but anyway I mean they land up with 53 bytes and you know processing time actually you would have taken a hit if you process 53 bytes at a time, ok, anyway.

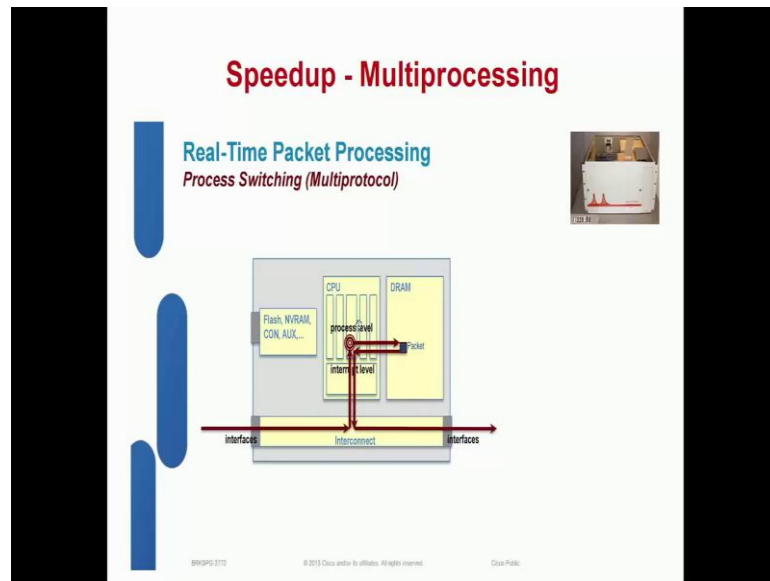
So, coming to this, so if in these kinds of routers, as the router size increases, one of the things that they do is you if you look at this diagram they you see some kinds of these are known as lines cards, those stuff that are here are known as line cards. Each line card has many network processors. If for a smaller router, there will be one network processor, but as the router size increases there will be more network process. Similarly, for a smaller router will be one line card as the router size increases there will be more line cards. And these line cards will be interconnected at the back through a switching element. So, this is in general the architecture of a router. So, the router has you, so from whatever discussion we had until now, you know that a router has to have a regular general purpose processor, it needs to have a specific network processor and all these things must operate at high frequency parallelism and pipelining must be implemented ok.

(Refer Slide Time: 03:46)



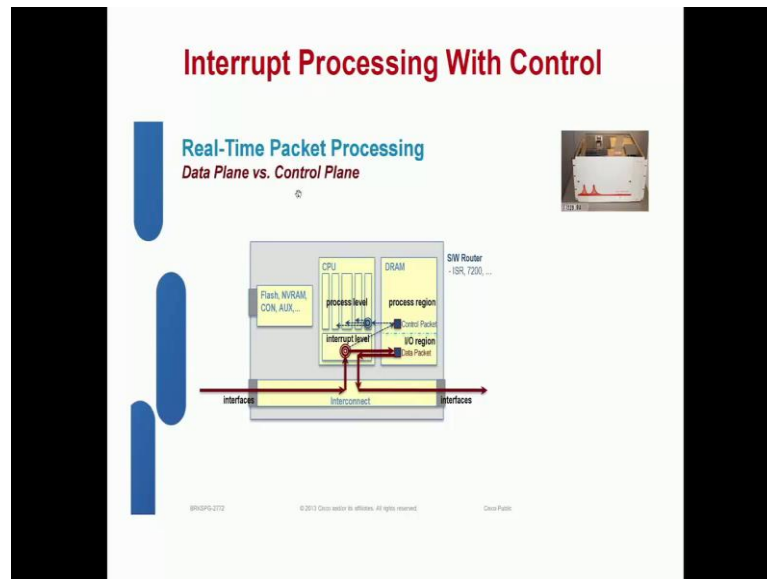
So, let us see how the routers have evolved over a period of time. In the 1980s, when the first router was made by Cisco, if they took a normal a computer, look at this, it had a CPU, it had a DRAM, and then it had flash NVRAM and then console and then auxiliary ports etcetera, which is normally what you get in your regular PCs. The only things extra that it had was the interfaces which connected different computers. So, a data packet used come here it the packet will be given to the CPU, the CPU will do all the operations stored in memory and then do all the operations, and then again it will route back into the interface and the data will go out, this is the first generation routers.

(Refer Slide Time: 04:39)



The second generation routers, what happened was the CPU started having process switching was implementing Linux operating system you know multi processing systems. So, what happened was instead of processing one packet at a time as soon as I start receiving the packet, I can give it to one process, the whole process will do all the operations and then send it to the other machines through the interfaces. So, this was the next improvement that was there. So, it is may need to help in actually real time packet processing, because as soon as the packet is received, you can just forward it to a process and then everything can be taken it. I mean it is not that every packet will be processed a using a process, at least you can and if you more than one Ethernet port, one port can be completely dedicated and I mean any data packets that array on that port can be buffered and given to a process. I mean you can do all those things in the OS ok.

(Refer Slide Time: 05:39)

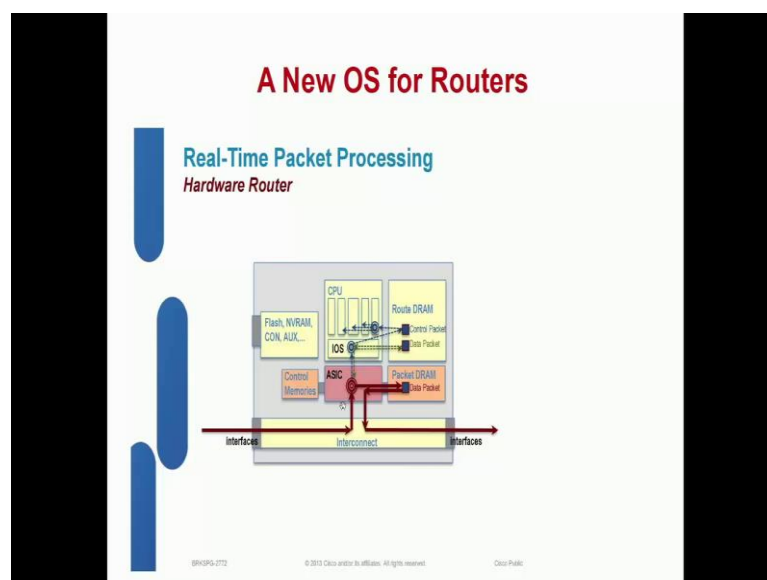


Then the next level, because they wanted increase in performance. So, what they did was whenever a packet used to come to the through the system, they used to actually generate an interrupt. And if you remember in Linux interrupt has a top half and a bottom half that is in once the packet is got, the packet is actually stored in a DRAM and then it the interrupt service routine returns, then a process picks up that packets and then performs the necessary operation. That means, you have to process the packets quickly, so as soon as a packet comes and interrupt will come, now the packet is taken put in memory and then the interrupt service routine will return back; it is up to the process to process the packets. So, in this case, if you look at this, the control packets were taken by the processors and the data packets were retained in the I O region. Why, because if I take from I O region go to memory, again come back to I O region, and then encapsulate the packet and then send it back it is going to take lot of time.

So, what I do is I just split the packet into control and data, the control packets alone will going into the processes for processing because any data packet if you remember the first slide I showed only the headers were added to a data packet, and the data packet did not change. So, why cannot we just store the data packet is one place take the headers alone do all the processing and then attach a new header to the data packet and send it that is the idea. In that way, they were able to get much more speed up and actually the

processor 7200 from Cisco, I am taking Cisco as an example because they actually occupy about seventy percent of internet routers. So, now, in this way, so the memory was divided into two regions; one is for the process region, the other is for the I O region. So, this was the next level of. So, in this case this was the time were the data plane versus the control plane per split into two. And remember the data plane needed a general purpose processor, because it was running algorithms sorry the control plane needed a general purpose process because of running algorithms. The data plane needed fast transmission of data packets ok.

(Refer Slide Time: 08:17)



Now, the next generation. So, what was the problem there, so the problem was that when I take this guy out and then put it in the control packets are put here and then the I O region was here the next level what I see is look at this, this is the data packet was sent to DRAM, there was no problem. And then this control packet had it is own operating system, because the scheduling see for example, when the internet was started people never conceive that voice would be a part of internet or video will be a part of internet, they only talked about FTP, telnet etcetera. But over a period of time what happened voice came and video came, and because of that quality of service became a major factor. Now if you are going to run a general-purpose operating system and then try to get quality of service you are infer trouble. So, they had to have a specific operating system

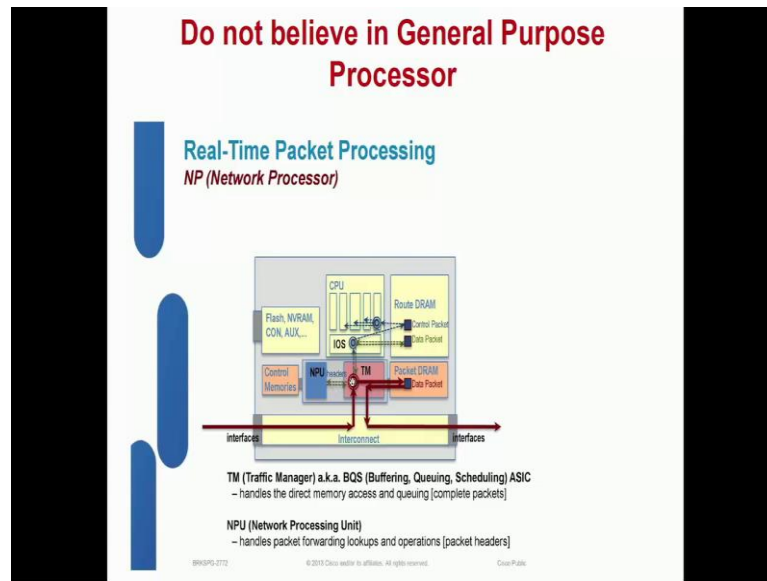
to take care of quality of service. Especially how does quality of service get affected because of buffers and timings. If you have more buffers then it delays the packet because of delay you know quality of service depends on jitter and delay straightforward.

So, now that I have buffers when I take more time to process my control packet, for that much time the data packet has to sit in my DRAM; and not only that if I delay processing my control packet my data packet also gets delay. So, in order to avoid this, they wanted to come with better scheduling algorithm. So, they put a small operating system, next to the CPU. So, then the next speed up that they try to achieve was because this process you have to switch you have to take the control packet out and then give it here and then the data packet out and you have to give it here, they put some ASIC, because the network speeds were increasing. So, they could not process this things at that speed with a general purpose processor, so because of that what they do was that the data packet speeds were increasing. So, they put a ASIC, the ASIC will do the switching very fast.

And not only that it had control memory. What is this memory, see once I calculate the routing there from which port I have to put it into which port then I have to store that information. So, I calculate all those things in the CPU then I stored this information here, so that when a data packet comes I need not send all the data packets to the top I can do some processing here and then put it in the, for example, I get a data packet from in a line card from one port and I want to send it in the to another port in the same line card. Why do you have to go to the processor to calculate it, because the line card has all the details. So, in that way, you have to improve the processing speed.

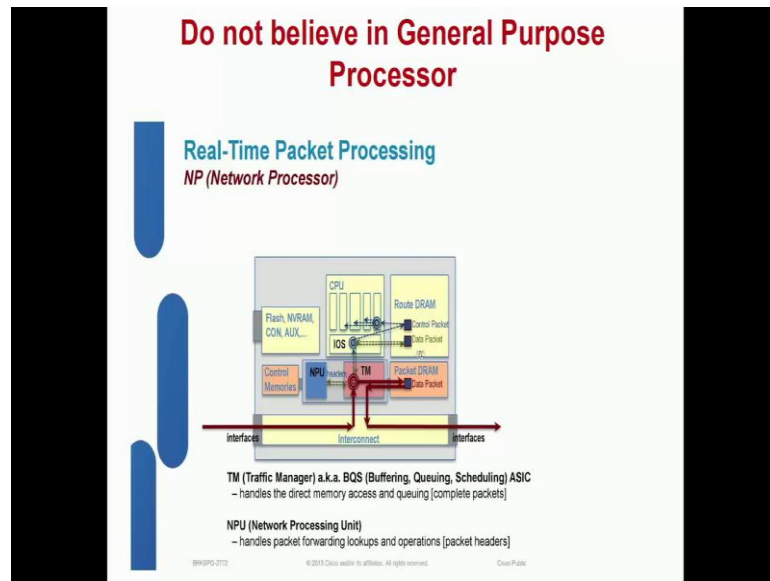


(Refer Slide Time: 11:32)



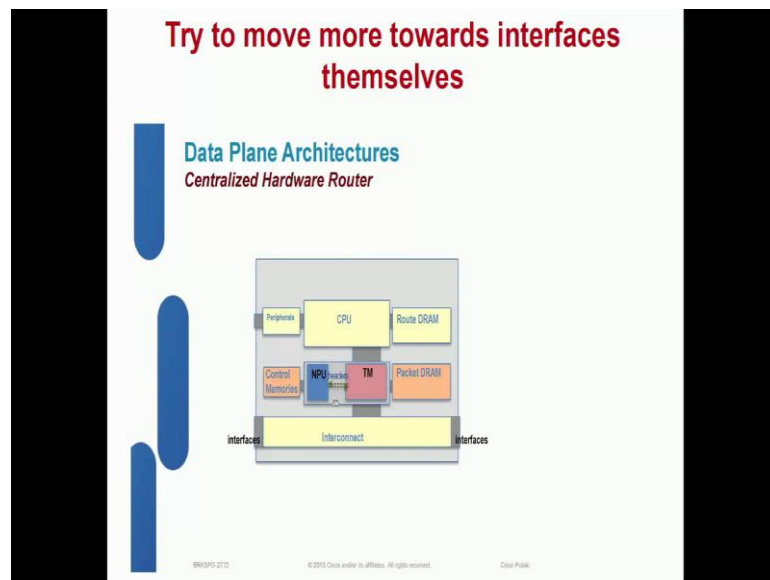
Then now came the requirement for network processor. So, once you started doing all this. So, people thought that now that I have put a ASIC here why cannot I put a processor there which will take care of switching very fast, and not only that this also has to provide a traffic manager as I told you for controlling the buffers, queuing and scheduling. So, three operations it has to do here; one is buffering the packets, and queuing them putting it in, because the packets can have priority and then what to schedule, which packets has to be sent first based on the priority. So, what they did was they put a network processor here. So, they call it as the network processing unit, we will see what goes inside later by then at the highest level let us see what they are doing. So, this did a traffic manager, and this handles the direct memory access and queuing so of complete packets, and then the network processing units looked at the forwarding look up and operations. So, this relieved the CPU the general purpose process of task of that forward look ups and see for example, remember when we got a packet we have to match the destination address and then send it. So, the network processing unit took up that job rather than the CPU itself trying to participate in that general purpose processor. Then what happens.

(Refer Slide Time: 13:10)



So, this was the control plane architecture. So, what they did to the data plane ok.

(Refer Slide Time: 13:15)



So, now what happened was if you look at this the data plane architecture is the centralized hardware router. So, what they did was they integrate the CPU - the general purpose CPU and the network together retained its own ram, look at this, the CPU had it

is own ram and it controlled the peripherals, and the network process had it own packet DRAM and it had control memories. So, if you look at the difference between these two, this is OS that was there actually moved into here, the network part of it. So, instead of running the OS in the general purpose processor, they just moved it to the network processor. So, then the advantage of this is so if you look at the data plane the data plane never goes to the CPU does not need any CPU for any operation right only the control plane needs the CPU. So, the data plane actually was handle by the network protocol part of it network processor part. So, if you look at this only the control plane has to go here, the data plane stops here. So, therefore, what they did was they took the data plane, this is how the data plane is and that is one of the reason they came up with line cards and then they had a route switch processor and so on.

Student: Sir, one question.

Processor: Yes.

Student: Yes, as you were explaining the evolution any specific advantage for NPUs over the ASICS.

Processor: Specific advantage of NPUs over a ASICS, yeah I will come to that I will answer that question at the end, because now I am just introducing you what are the components and how they are split this NPU and all that. Once we complete, when I come to performance then I will tell you why we need ASICS, ok.

Student: I know because your slides where you are showing the slides in a sequences as an evolution, so, that is why I just ask.

Processor: Yes that is right, but we will understand why we go for NPUs, see ASICS are usually use for switching

Student: Yes

Processor: Ok, so I will tell you why ASICS came into see initially, yes they were just switching the packets therefore, they were using ASICS. So, later what happened was they wanted to handle this buffers and other things; at that time switching just may not be enough you need a processor and you need to store the packets before you start sending those packets out that is where they brought in NPU network processor unit that is that answer your question.

Student: Ok sir

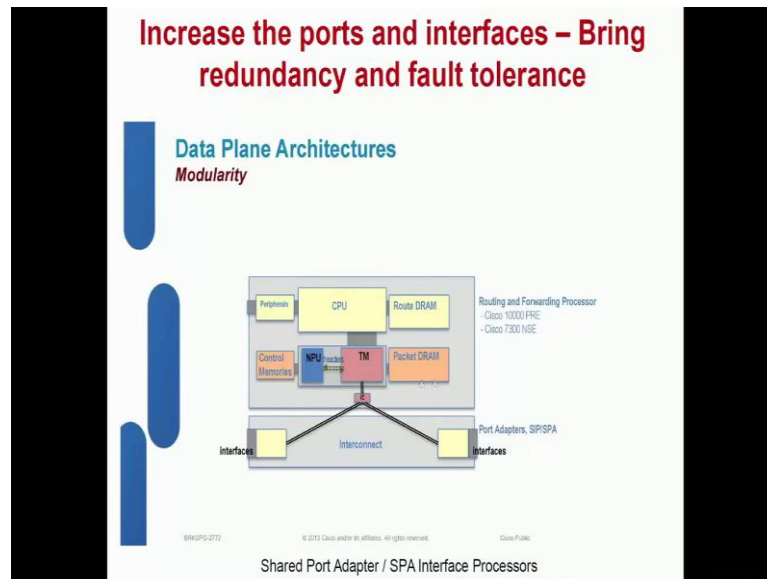
Processor: See initially, it was only for switching

Student: Yes

Processor: then you have to buffer, therefore they brought NPU fine. So, let us proceeds. So, see one of the things that you should see in from this slide is see the data packets, if you look at this, they need not go into the CPU. So, what you can do is you can just take the data packets from the interface and then you can just get it back into the interface again without going into the CPU, so that is and this can be done very fast. Now what happens if you go to the CPU, let us take for example, a traffic manager. So, I told you during the delay and jitter gets affected. Now how does jitter get affected here. So, for example, you are using some sort of a scheduling algorithm in CPU right.

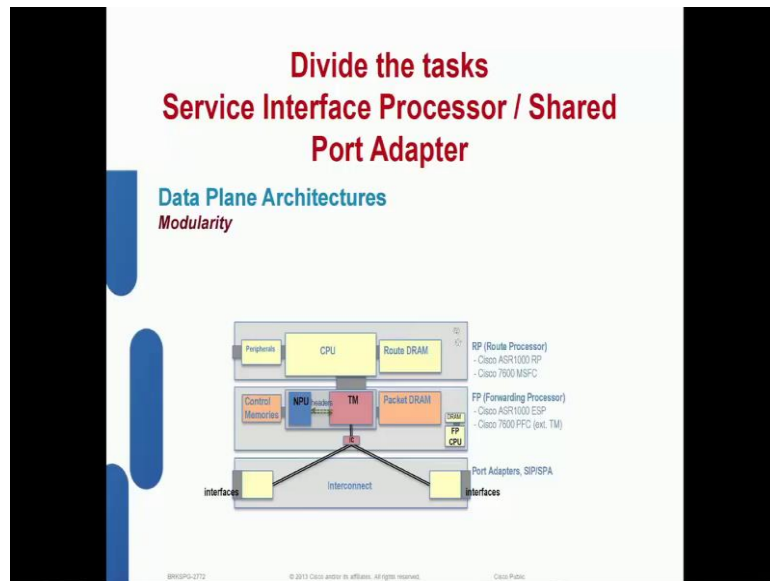
So, if the CPU is involved in transmitting the data packets the CPU will do it based on I mean the OS that is running on the CPU will do it based on what type of scheduling algorithm you are going to use. Now it is never going to be first come, first served; if is first come first served then you can to some extend to expect that the data packets will have the same jitter, but you can use any type of scheduling algorithms. So, that is the reason they really wanted to move these things out of out of the CPU and give it to your special network process. Now one thing that happened was now with a network processing unit and a traffic manager into picture, this device itself was capable of handling more data traffic. So, instead of having just one interface and then using one network processor to handle that interface, what they did was, they did something like this.

(Refer Slide Time: 17:47)



The interface became independent of the processors. If you look at this, here the processor and the interface were all connected together; the CPU, the processor - the network processor and the interface were all connected together. If you look at the next evolution they just removed the interfaces and the interconnect together, because what they were able to do was we were able to receive data packets at a higher rate from different interfaces and then pass it onto the network processes and these network processes did all the calculation and then send it to this. So, these are called as port adapters or shared port adapters or SPA interface processors, they called it. So, a CISCOs products like 7300 and 10000 actually use this kind of an architecture. Now with this the advantage is you can bring redundancy; that means, I can have more than one interconnect then I can also bring in fault tolerance into this network, because if one interconnect the card fails, I can put another card and then the whole thing will work. So, this is the next architectural improvement that they did.

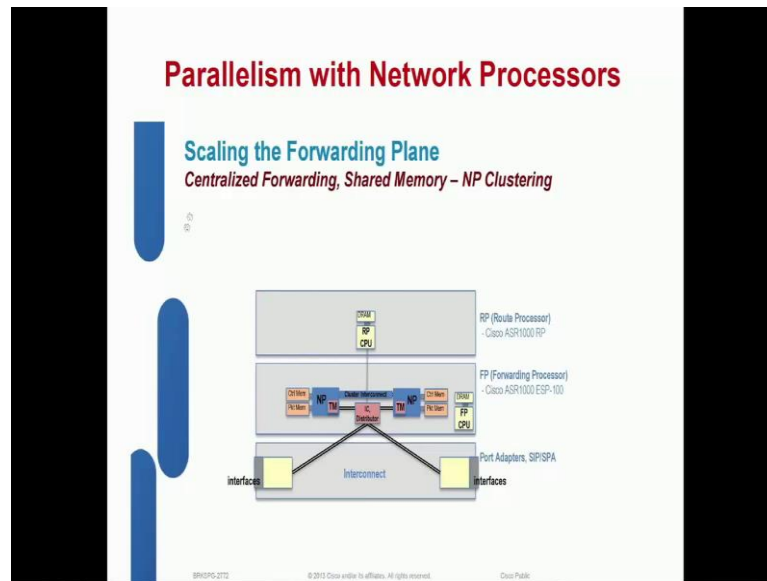
(Refer Slide Time: 19:03)



Now, after that what happened was see they divided, so look at this then they put one more dash split it one more here. I can have one route information route processor, I can have many forwarding processors, and I can have many port adapters that are connected to forwarding processes that is the next improvement. As I told you know this is the way they actually what is this architecture this is parallelism or pipelining, they try to bring in more parallelism - the hardware parallelism. So, in this case if you see the CPU had a route DRAM and peripherals that it controlled, and then the network processor had this was now this was named as a forwarding processor, because what happened was the data packets came here and then it looked into the table, and then it route in the packets through the interconnect.

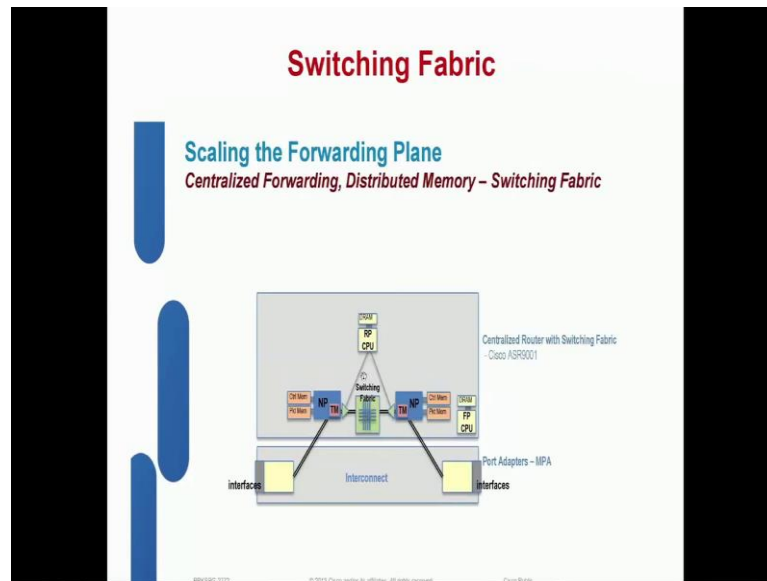
And now they had more than one, so if you look at even to today in hub and all that you have 8 port hub, 16 port hub and so on right. Similarly, these are the interfaces that ports, so you could have say 256 port device, it will just be a line card and that will be connected to a single NPU or a bunch of NPUs, you can do ((Refer Time: 20:24)). So, you can have variations architecture, remember I showed you the architecture in the first diagram.

(Refer Slide Time: 20:30)



Then what happened ok, so what happened here is scaling the forwarding path, parallelism with network process. So, the network process themselves were so if you look at this instead of having one network processor, they had multiple network processor and they were connected by a integrated circuit. So, essentially, if you look at this and then the route processor was relieved of all the burden. So, there was one a small route processor - a single route processor and then has a forwarding process they had and then you had this interconnects. So, if you look at this, this next stage of evolution I had only one.

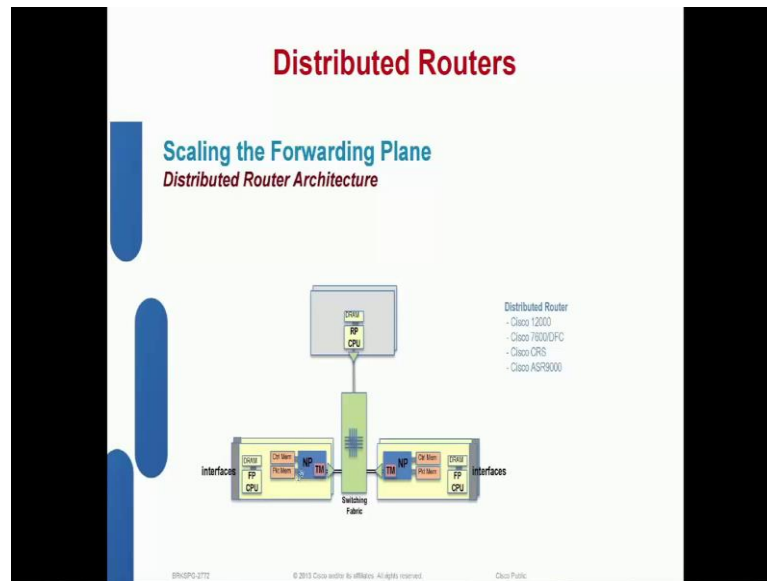
(Refer Slide Time: 21:19)



Now, I have many, you are interconnected parallelism at the next level. Then what happened, sorry, then what happened. Then the introducing because they had to communicate between these networking process and this RPU and these interconnected and so on, they actually introduced switching fabric. And this switching fabric, they called it as a back end, they put it at the back end, so that I can put the these guys in a card, I can put this guy in a card, I can put these people in a card and then I can connect all the of them using a switching fabric at operated at a high speed. So, in this case you have a centralized forwarding, but distributed memory, centralized forwarding because you are using this network processor and route processor etcetera together and distributed memory, because each one had his own memory. So, the CISCO ASR 9001 which I showed you in one of the slide which is the, so had this and then the next level ok.

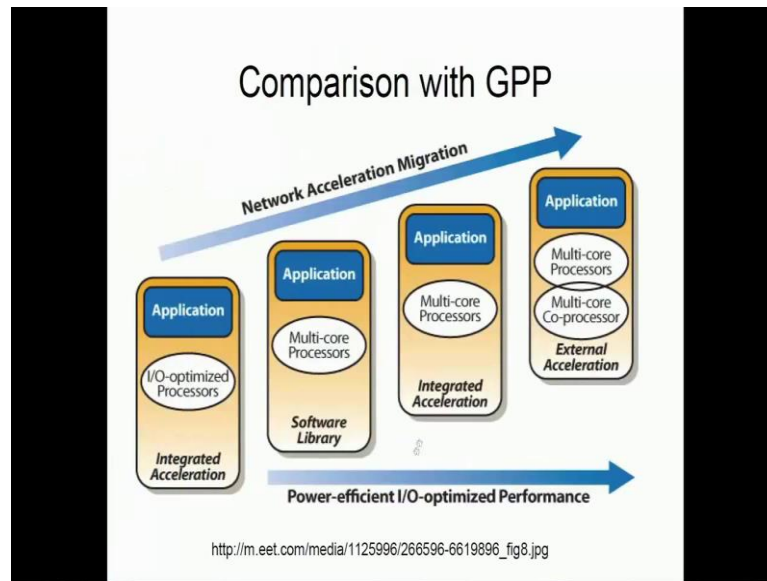


(Refer Slide Time: 22:28)



You had many CPUs, you had many interfaces, and each interface had its own network processor and all of them were connected via the switching fabric. So, if you take this guy alone, he will act as a router; this guy alone, he will act as a router as a switch; this guy alone will act as a switch, and this guy alone he will act as a router. So, each one, so here you see, there are two. So, this is massive amount of parallelism I mean they call it as a distributed router architecture; obviously, because it's like distributed computers; each one was self-sufficient on its own and architecture like CISCO 12000 etcetera followed this kind of an architecture ok.

(Refer Slide Time: 23:11)



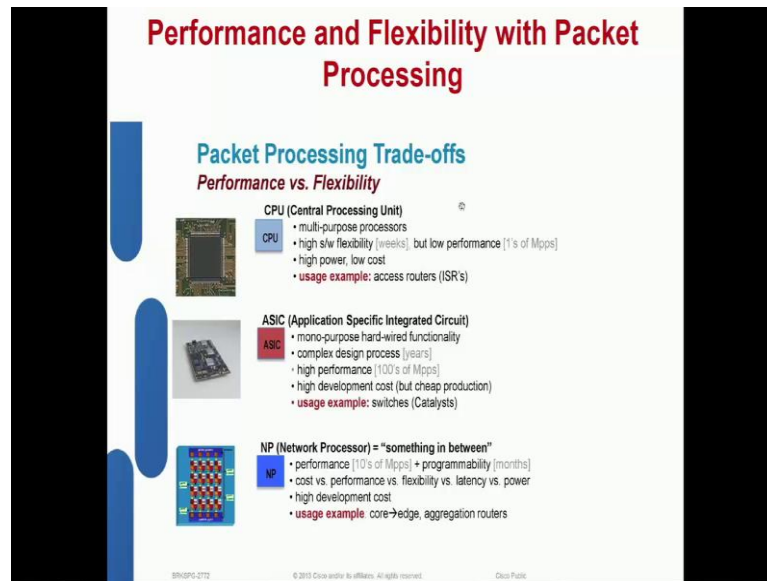
So, when you compare with general purpose processors, I mean you have this application I O optimized process and then integrated acceleration then you add software library integrated external acceleration also. If we go with general-purpose processors this is way they grew. And for networking, they had multi-core and all those things instead of having a single network processor they had multiple network process and so on. So essentially, it is a kind of parallelism, but remember as you increase the number of processors, you have to look out power and security. So, until now you are been only looking at the architecture not security and power.

(Refer Slide Time: 23:54)

## Performance and Flexibility with Packet Processing

### Packet Processing Trade-offs

*Performance vs. Flexibility*



Component	Characteristics	Usage Example
CPU (Central Processing Unit)	• multi-purpose processors • high slw flexibility [weeks], but low performance [1's of Mpps] • high power, low cost	access routers (ISR's)
ASIC (Application Specific Integrated Circuit)	• mono-purpose hard-wired functionality • complex design process [years] • high performance [100's of Mpps] • high development cost (but cheap production)	switches (Catalysts)
NP (Network Processor) = "something in between"	• performance [10's of Mpps] + programmability [months] • cost vs performance vs flexibility vs. latency vs. power • high development cost	core-edge, aggregation routers

BR/SPG-2772 © 2010 Cisco and/or its affiliates. All rights reserved. Cisco Public

So, here is the question that was asked. So, why should I need a network processor why cannot I go with ASIC. So, this answers the questions, this slide actually answers the questions. So, look at these performances versus flexibility. So, if I have a CPU, you can have multiple purpose processors etcetera and these general purpose CPUs are used in access routers. If I have an application specific IC, you have very high performance; whereas a network processor is in between a CPU under ASIC. See ASIC actually cost you a lot. So, in order to achieve a balance, CPU, I mean general purpose processor are actually very cheap. So, this NPU actually lies between these two. So, now, that so if you look at this network processor are used in say edge routers, and aggregation routers and then core routers then the applications specific ICs are used in switches layer two, and the network processor used in layer three usually, I mean, but not a strict rule, if you are a hardware guy I mean you can sell it in that way.