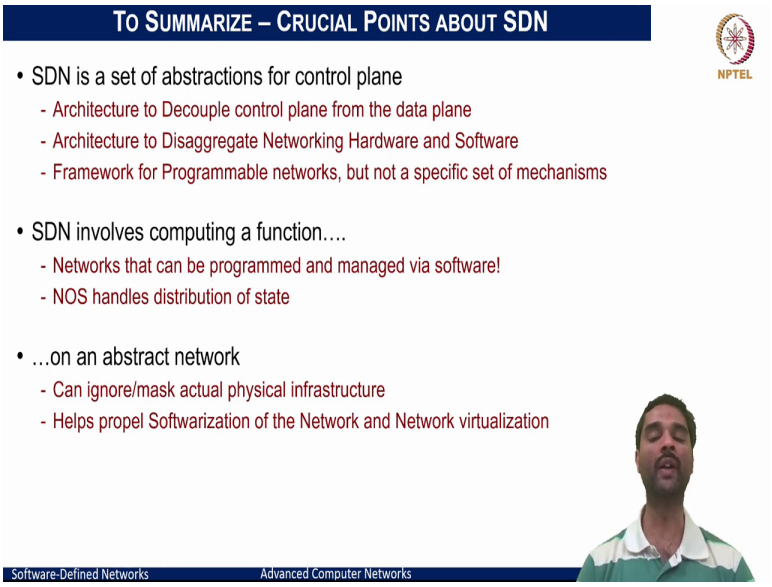
Advanced Computer Networks
Dr Sameer Kulkarni
Department of Computer Science Engineering
Indian Institute of Technology, Gandhinagar
SND Prospects and Challenges

(Refer Slide Time: 00:16)



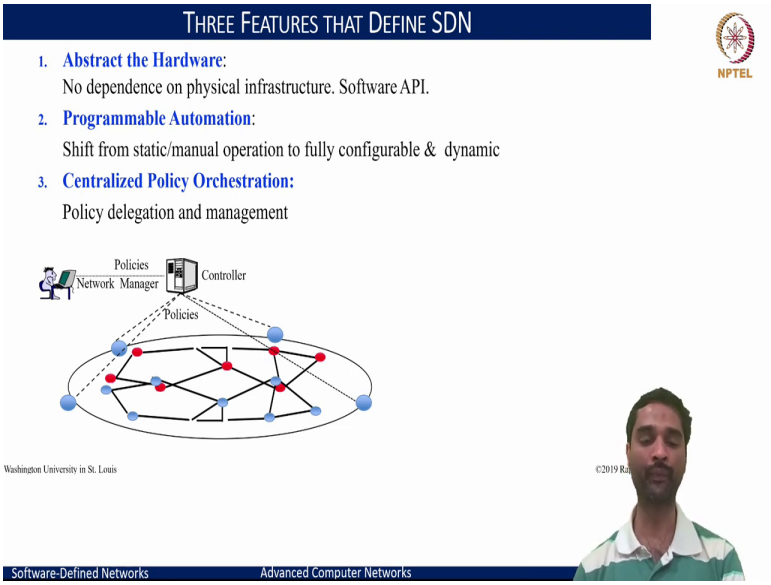
To summarize, SDN can be thought of as a set of abstractions for the control plane. In that, we began with SDN as an architecture to decouple the control plane from the data plane of the networking elements. Then we went ahead and showed that this architecture would also help disaggregate the networking hardware and software, enabling us to mix and match open hardware and proprietary software and vice versa to be run as networking elements.

In a nutshell, what SDN enables is a framework for making the networks programmable. In this sense, SDN can be thought of as a means, as a mechanism, as a paradigm. But I want to caution that it is not a specific set of mechanisms. It is but a framework or a paradigm that enables to build the programmable networks. And when we say programmable networks, we mean about providing the network functions as a computation that can be done on the commodity hardware. And the way the networks need to operate, the way the networks need to be managed all can be done as programs managed via software. And to this end, the network operating systems or this SDN controller would provide the right set of abstractions to ensure that it is easy for the user to interact with and build the applications. And we will also hide the complexities in distributing

the states and taking care of the hardware-specific aspects when you want to write to various proprietary devices.

And such a computation of a function really enables us to build an abstract network to the core. What this means is, as a researcher, as a network programmer, you can ignore and mask out the actual dependencies that a physical infrastructure or devices would bring but rather focus on the abstractions and build the networking applications right based on those abstractions. And this will help propel this authorization of the network and also aid towards building the right network virtualization.

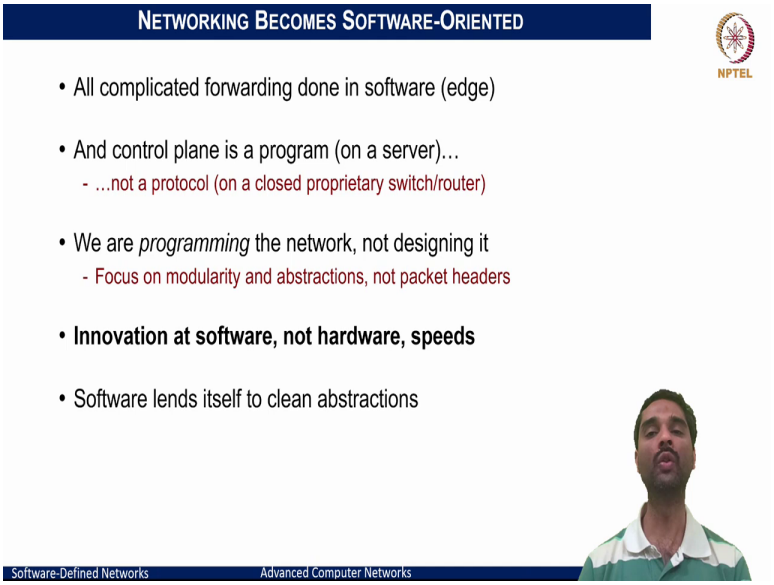
(Refer Slide Time: 02:41)



To remember, you have to think of SDN as three specific features. One is the abstract, the hardware, that is, SDN decouples, the dependency on the physical infrastructure, and provides the right software APIs as the abstractions to build the network functionalities. Second, this concept of network functionalities can help us further to build what we can call as programmable automation or networks that can be self-managed and self-organized. This would be a drastic shift from the manual operator involved or a human involved operations that are static and manual to a fully configurable and dynamic self-organizing network.

And the means to do this would be to have a centralized policy orchestration, where a network manager would once write the set of policies that need to be implemented for a given network. And over a period of time, he would change those policies. But then, these policies would be imbibed by the SDN controller. And it takes care of translating these to the right set of mechanisms that need to be applied over each of the distributed networking elements, that is, routers and switches within the network topology.

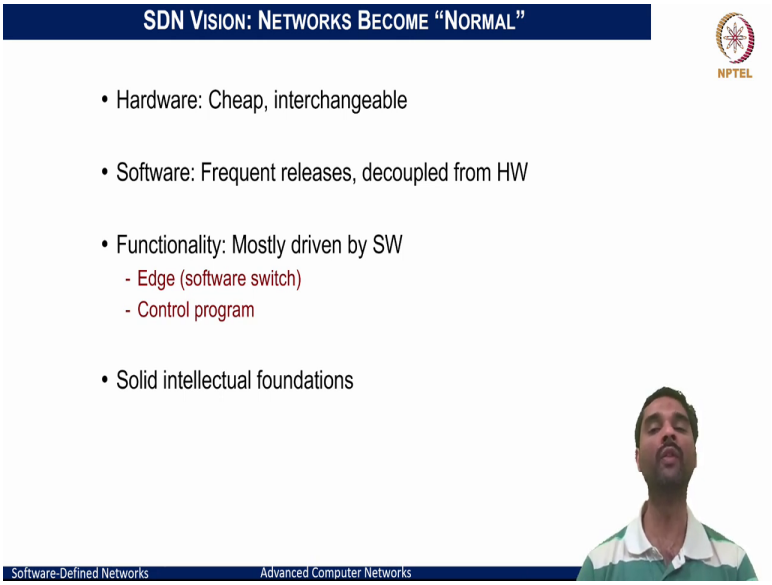
(Refer Slide Time: 04:06)



Thus, what SDN allows is to make networking become more software oriented. Because we have gotten rid of all the complexities and building the forwarding, we have enabled the flexibility in facilitating the need-based forwardings that a programmer can program and control much more easily. And this can be done from a single point. That is where a centralized control plane helps build these aspects. And all of this put together, what SDN has really done is to focus on the right modularity and abstractions, not just worry about the set of protocols, but build the protocols for the sake of providing the modularity and abstractions. In a way, this is an innovation at the software and does not care as much about the hardware or the performance that you get with specific ASIC-based implementations. Nonetheless, this also means that performance could as well be a challenge when it comes to the softwarized networks, which can be worked out in the future.

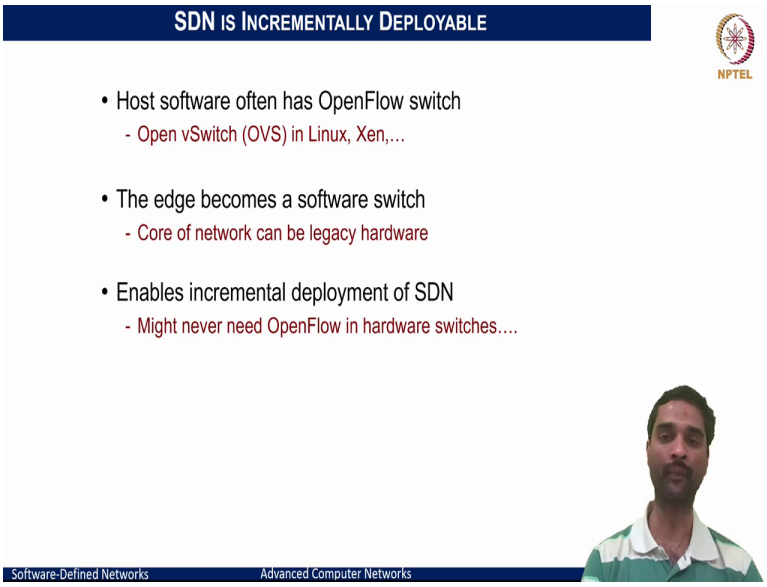
And there have been several books already that are trying to better the performance on how we want to build the applications in networks that are going to leverage the better utility of the hardware. And the summary here is that software lends itself to a clean abstraction, and that is what SDN does to networking.

(Refer Slide Time: 05:37)



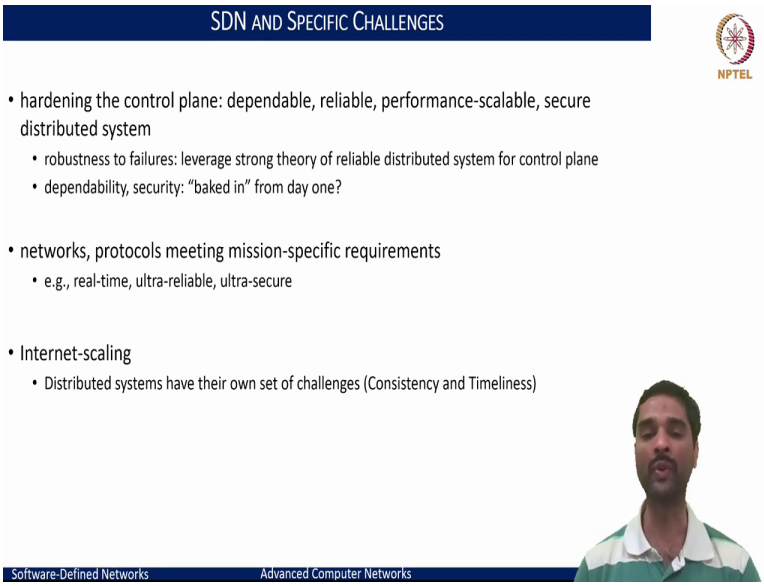
And this way, we can now envision the networks to become much more normal in a way that it is easier economically viable to build specific custom hardware, build custom software's utilize basically the commodity hardware as the network elements, as routers and switches that we want to build rather than purchasing costly vendor-specific devices and making entire functionality being driven by the software that is the control program and the abstractions for the control program, including the control agents that are being put at the switch, make the right foundations for building the networks much more that are agile and flexible and open for innovation.

(Refer Slide Time: 06:24)



Thus, SDN can readily transform the networks, and it has over the last 10 years as SDN has evolved as such. And one of the main reasons why SDN could evolve is that it is incrementally deployable, especially when we think of the internet infrastructure, which is worth several 1000s of billions of dollars; you cannot change in one day and say that I will replace a router with SDN capable router, what we would instead do is have the core as it is and change the edge networks with SDN capable routers and switches And both would then interact. And then, as time progresses, we could think of changing the core infrastructure also to become SDN enabled. This incremental deployment has really enabled SDN can be made in road with the existing infrastructure without any hassle.

(Refer Slide Time: 07:23)



So, we have looked at the green side of SDN, and let us try to see what challenges lurk with SDN. Primarily, we say that we separated the control and the data plane. And what this also means is, in a way, we are hardening our systems around this control plane to dictate what a data plane needs to do. And in a way, this would grow as a dependable and reliable system. This dependency as we have seen ossifies in specific aspects.

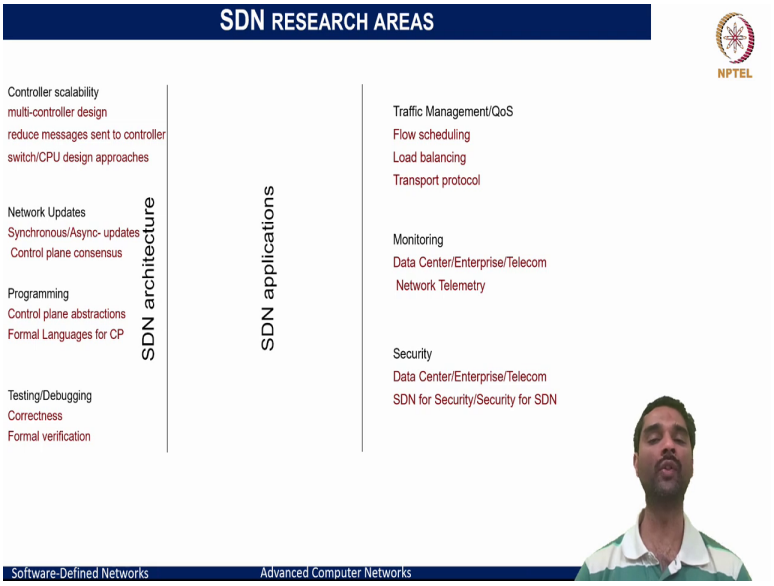
And here, we need right computation to ensure that there are good set of abstractions built and multiples of the network operating systems that are competing in the market and when we do something in software, as opposed to hardware, performance and scalability always become a concern. In one way, it is scalable, because we can spawn software much more easily than buying hardware. But what we mean here by scale is to support the number of users or support the number of flow rules that otherwise an ASIC would support. In a single device, this would be a challenge, and to meet the performance requirements of the carrier grade networks, especially in telecommunications, becomes a major concern. And also, when we build a distributed system, security becomes a major concern in terms of what happens if one of the system gets broken?

And how would this consistency of the data be also managed, and all these aspects bring in as the specific challenges that need to be addressed for the SDN. Nonetheless, over the last decade as the works have evolved, there have been several of these solutions that have also been put forth towards addressing these specific concerns. We think of SDN and then say we have a mission critical systems.

And if we know that communications with the control plane is going to be an additional overhead, then we have to think back whether this is suitable for such mission critical aspects, where latency is prime. And this is where again, specific adaptations or specific mechanisms can be rolled in to say how things have to be worked out, so that we can have real-time and reliable and secure communications with the SDN framework.

And like I said, internet-wide scaling is another aspect wherein if you have a distributed system, the more major concern for the correctness of operation could rely on the consistency of the data that is spread across different nodes and also the timeliness of gathering that data becomes a challenge. Thus, we have several set of things to work on. But we can see the benefits, and we can see the challenges that it brings, and always try to improve as we go.

(Refer Slide Time: 10:13)



And just to point to some of the specific SDN research areas that may be of interest to you, if you get engrossed with SDN could be one to think of how do you want to scale the controllers? That is, if you want to define multiple controllers? How would they coordinate amongst themselves? What is the consensus mechanism that you would want to build, and how you would want to replicate the information so that there are no overlaps or there is redundancy, but with less storage and communication overheads, all of these become interesting aspects to work on.

And when it comes to the network updates, if there are multiple controllers now trying to update, define and dictate the forwarding plane functionalities, how would they be consistent and ensure that one would not overrule the other's updates? So, having these synchronous or asynchronous mechanisms for updating and having the control plane consensus in a distributed system makes another interesting area to work on.

Likewise, when we think of abstractions, we also can question whether the control plane abstractions are, in essence, complete. And what is the way that we can think of the better programming constructs to build for repairing these abstractions or formal languages for ensuring that these control planes can be tested for, in terms of the completeness of providing the functionalities, and all in all, these vital aspects rely primarily on the SDN architectures.

And even correctness and formal verification, provided a given SDN architecture is deployed, is also a concern in terms of how would we test and verify that the things are operating the way they are expected to operate. And this is a challenge in the traditional networks, as well as in the SDN, not a new aspect. But we need to basically think through these aspects to work on and provide better alternatives. And when we think of SDN applications, there are myriads of them, and it is innumerable to put what we can think of what lies ahead in the future. Any innovations in terms of the application space can be readily brought in, and this is as green as ever to build any new applications to suit the smart cities, Smart Grid networks to suit the 5G era or 6G era what kinds of innovative applications would we need, what type of traffic engineering we want to do, what kind of telemetry applications or monitoring that we want to do based on the SDN could all be a new variety of applications that can be built.

The other aspect in terms of the SDN operations would be to think of like how would you want to manage the traffic or quality of service? This includes the standard flow scheduling mechanisms, the load balancing approach, and what transport protocol they should be using for intercommunication between these northbound and southbound, and the NOS and the networking devices, all of these.

And when we think of data centers, or enterprises, or telecommunication networks, monitoring has always been a challenge. How do we want to do fine-grained monitoring so that we are able

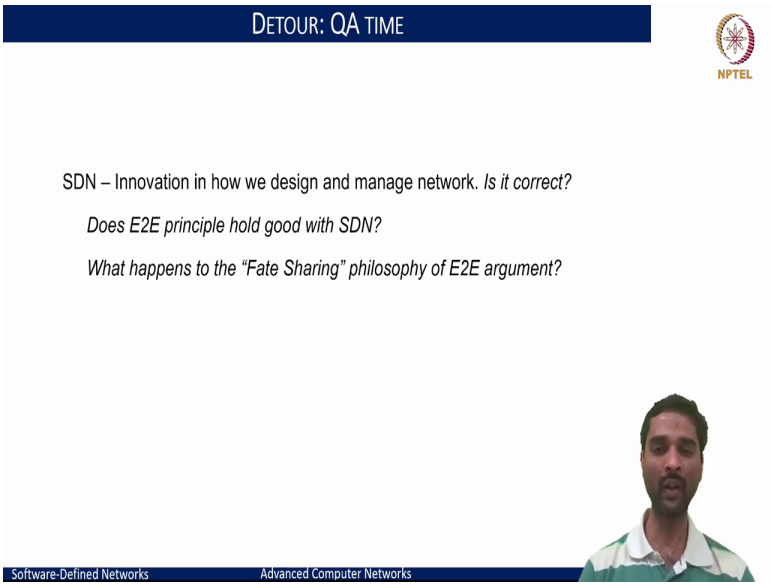
to precisely detect, and even perhaps predict in the future any of the anomalies that can happen in the network and prevent such anomalies before they really occur?

And also, security is an open era. In fact, many have debated that SDN opens a new weaker link in the chain of security, making it much more vulnerable than any other systems. Because if you think of network elements, if I get into one, I may not be able to do as much harm. But if I am able to hack into the SDN controller, I am literally able to control the entire network and manipulate the network the way I would want.

So, this really jeopardizes the security for SDN when we plug the SDN into the networks. But nonetheless, SDN is also seen to augment security because we can now build security specific network applications on top of SDN and enable those to detect, verify, and also ensure that you can predict any of the anomalies that could happen.

So, there is a good research that also happens in the way of saying SDN that you would want to use for security purposes. And on the other side, you would want to secure the SDN and how would you want to make the security framework for SDN. These are some of the green areas of SDN research, which may be of interest to you in the future.

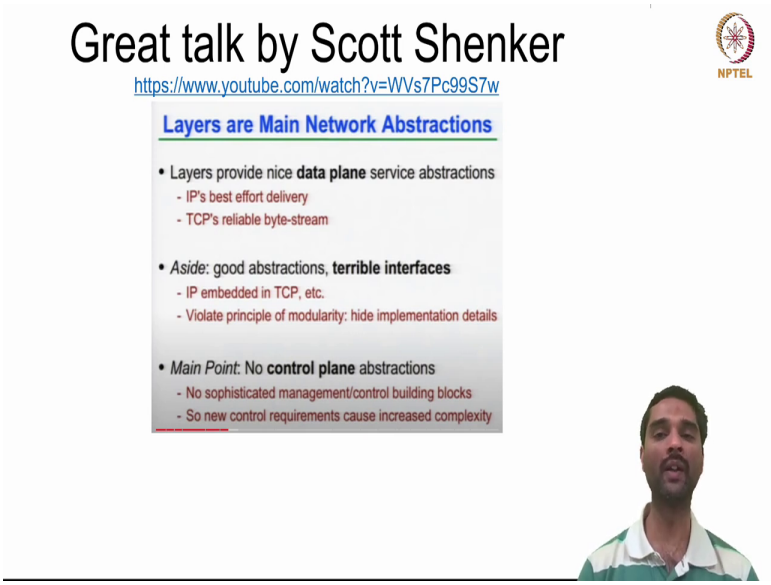
(Refer Slide Time: 15:06)



Lastly, I want to leave with a set of questions. Think of if we have SDN, we said it is an innovation in how we design and manage the network. So, with the entire series of lectures that

we have gone through, think and say whether this is really correct, or are there any missing pieces. Second, we did look at end to end principle earlier. And now we are thinking of SDN, which is changing the paradigm of communication. So, how does SDN really impact the end-to-end principle? And in the end-to-end principle, we also discussed the fate sharing philosophy. And now, does it hold true with SDN?

(Refer Slide Time: 15:51)



But to get to your answers, think through the set of lectures that we have heard, and also, I recommend this talk by Scott Shenker, who really set the tone for what is the set of abstractions? And how and what is the vision that they had for SDN in trying to bring out and trying to hide the complexities of the network? And complexities always mean the inefficiencies and abstractions are the means to overcome those complexities and how they are able to build for the networks. So, this was a very short talk, you would be able to go through it, understand and appreciate the vision that they had long back around a decade ago to see what SDN is all about.