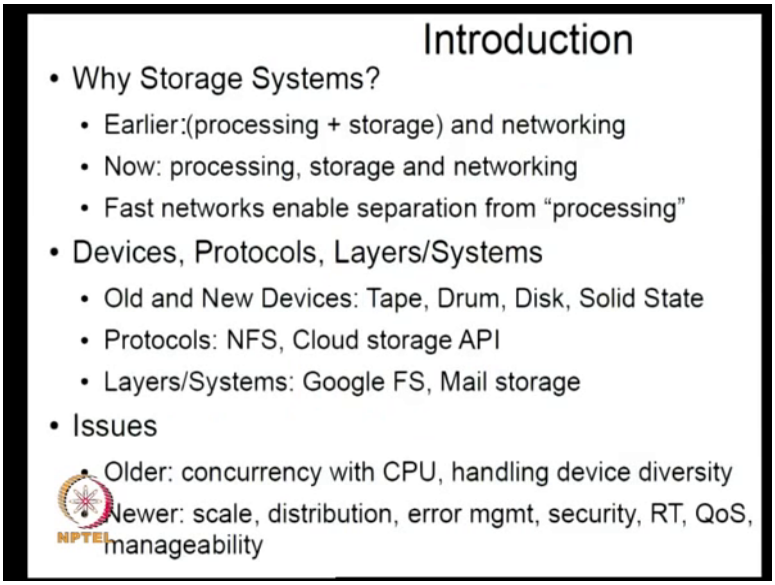
Storage Systems
Dr. K. Gopinath
Department of Computer Science and Engineering
Indian Institute of Science, Bangalore

Introduction to Storage Systems
Lecture – 02
Evolution of Storage Systems, Important problems, How Storage is different from Processing & Networking

Welcome to the storage systems course. This is being offered through nptel and my name is Gopinath. I am from Indian Institute of Science, but I am a professor.

(Refer Slide Time: 00:38)



Today let me briefly mention what storage systems all about. I think most of us have used computer systems and our computer systems visually stores some information, and we try to retrieve it later. Some of it directly what we store and retrieve some of it done by other parties for example, if you write a mail or if you write some document you create it and you retrieve it.

Sometimes other people are doing it for you for example, if you use google file system or a google system for searching, somebody is storing the information and you are using them indirectly. So, storage systems can be small like ones which we create ourselves or it can be very big like the ones which google and facebook those kind of parties provide or create. Now the important thing for us is to know why storage systems have become important. In

the past typically, you will find that the processing of storage used to be together even now if you look at a current pc you will find that the storage comes to the pc all right.

The storage is not outside the pc it is always inside the pc. So, and also it sits on the particular bus and what is called a PCA bus. Which electrical bus so; that means, that it has to be present close to the city because it is sitting on a bus. It is not on a network the storage is not on a network it is actually on the electrical connected to the CPU where CPU is connected. Now So, in the past it always has been the case that storage was an integral part of the computer systems and also why that is the reason why even operating system would consider storage as it is an important component because of operating system you also cannot boot without it is kernel image etcetera available on storage system it was the integral part.

So, it was quite natural that processing of storage would consider together, but with fast networks this has changed. So, because a lot nowadays you will find that people often say that there is processing storage and networking and some of you might already have experience of this in local area networks you will find that your directories may be accessed through networks.

For example, there is a protocol called NFS network fail storage, which you do not have the storage on your machine on which you are working on it is actually being provided somebody else some other machine. So, that is something called network storage that is available to you.

So, essentially the processing is happening on your local pc your storage is coming from what are called NFS servers and they are connected through networking. And this is feasible because fast networks are available. Now for example, in 1980s you had 10 mega bit per second ethernet, 1990s early 1990s we had something called fast ethernet about 100 mega bit per second and about 2000s we had something called fast sorry giga bit ethernet which is thousand mega bit per second and in the near immediate feature we are getting 10 giga bit per second also.

Once you have fast storage it turns out accessing data or storing data is not going to be that costly of an of your it can be done fairly efficiently, in small amount of time that is why what is happened is that this networks have enable separation from processing. So, that is basically a critical issue. So, if you separate them out then the accessing party and the accessed party that is the client and the server they have to understand how to make sense of each other

right? One guy is requesting one guy is responding; that means that there has to be some kind of a protocol in place.

So, in addition to protocols you also need certain way of managing the systems the 2-combined system, because there is a client here, there is a server here and there is a network, here right? In between now anything will happen to any of these 3 things the server can die, the client can die, the network can die somebody has to manage it. So, there is now a system also required. So, in the previous systems it definitely had devices discuss (Refer Time: 05:51) over there before networking there going to be there afterwards also.

So, you will have the same devices as before, but now you are going to have a protocol because you have your client and servers which need to talk if they have to talk they need to have a common way of understanding user that is the protocol. And you need some management in addition. So, that your multiplicity of actors things like client, servers and networks you need to figure out what they have to do how they have to be started in beginning how if something fails how we have to handle them.

So, I given some examples of this if you look at devices you will have things like tape drum disk and now a day's solid-state disks drum etcetera is absolute the tape is still going on for very high-density storage and low cost this still good disk is widely used it is it is being used for a last 50 plus years, but I think there is a now a significantly threat from solid state devices.

Now, there are lot of protocols I mentioned already a networked file storage that is NFS in the windows world there is something called CIFS, and now a day because there is what is called cloud computing systems you also have what is called cloud storage for example, when we use your Gmail system your Gmail client on your system right on the browser right it is actually accessing your mail using some api. Which Gmail has which google has provided similarly, if you are talking about facebook etcetera there is some API by which you are accessing some of the data. In the case of systems we can think about let us take mail storage and there are some protocols you might have heard about something called POP 3 IMAP 4 etcetera.

So, there are the mail is stored each piece of mail is stored as a possibly separate file. You told to tell it which file you are talking about which file to delete which mail to retrieve etcetera all those things are basically the IMAPed essentially it gives you some let us say

protocol by which at the application level now this is happening at application level because this IMAP is being used by your mail client there is one more level below which is the storage client aspect again a storage server site there is one more level below that, but from our point of view we can club both these things together and say that are abstract level from a point of systems you are basically using imap or POP3 as a protocol to name (Refer Time: 08:55) some mail read the mail or delete mail etcetera or move things around all these things are able to do.

Now, if you look at the issues that were there in the previous systems and the current systems, you will see some small changes. In some areas and some major changes some other areas. If you look at what was the issue with old systems old storage systems the most important issue was concurrency with CPU. Why is an important issue? Because notice that the throughput of a system depends on how to clear able to access the storage blocks. Good example is consider booting of the system when you are booting the system you are going to pick up the kernel or the or the operating system parts of it from the storage system.

Now, how quickly you boot depends on the speed at which you are taking it up now while system is booting it can be doing other things it is installing certain things right? So, while installation is going on you can use principle access other blocks required other storage blocks. So, while the booting is going on 2 this can be going on one is initializing of the operating system structures as well as taking up what is might be needed in the future the blocks shortage blocks.

So, if you are able to keep going keep both these things going at a same time we will have higher let us say throughput or you will get lower latencies with respect to your booting etcetera. You can also use the same model for applications also applications what happens is that you could be doing some computation and then you can also be once in a while accessing the storage devices and storage devices are very slow they are about 4, 4 order some magnitudes slower typically so; that means, that you have to find a way in which you can keep the CPU going at a same time as the storage systems also.

So, that is why concurrency is an important issue this is also an important issue even now it has this has not gone way similarly if you look at device diversity. What is device diversity? As I mentioned earlier we had things called drums we had something called tape disks solid state etcetera and there are varieties of them this not one single variety of disk that actually

multiple generations of disks for example, you might have something called IDE disks you might have something called scsi disks you might have something called now a days you have something called SAS disks now we have something called solid state disks varieties of disks are possible.

So, the things is you need to be able to handle the diversity and again these things are very core components of operating system design how to keep systems concurrent. So, that multiple activities are going on at a same time how to handle varieties of things because the operating system is changing all the time with types of devices, newer devices keep coming in we have to figure out how to keep the operating systems structure stable while devices are changing.

For example if you look at it the Linux kernel does not change because a new device is coming you just write what is called device driver, which helps it to access that device the structure of the kernel itself is still about approximately mostly the same the core kernel does not change that much the way it is done by this is being some things called device drivers and that is one way to handle device diversity and that is the same thing even with the case of even current situations. So, these has now nothing has changed here what is the new thing that I have changed in the storage systems one is scale in the past for example, when I first started working in this institute for example, my disk was 5 megabytes later it became 40 megabytes.

Now, I have a 250 giga byte disk. And my 2 x now have the 3 tera byte disk I still I am with that 250 giga bytes and Google is about 100s of peta bytes or more. So, the scale has become dramatic that is distributed also it is not in one single piece. Now a days you will see that I have some stuff on my memory stick here I have something on this system here I have something that my laptop and I have something on my Gmail somewhere it is distributed all over the place that is from personal point of view, but if your talking about big company or organization, there it also distributed for example, if you take Indian institute of science there are lots of departments lots of administrative documents which has started all over the place right?

So, the storage system that is handling this will also be at multiple places the server thing that we had worry about error management why this is an important issue because once you have large amounts of data you will find that there can be multiple ways in which the data can be

become bad one is that lot of people are using it. So, you're in spite of the best training people sometimes get things wrong they delete files wrongly right it must happen to you it happens in organizations also.

So, that can be problematic it could also be because of the hardware itself going bad a good example is some of you not have experience of having cds which are no longer readable. In the past for example, earlier 20 years back we use to work with floppy disks and we used always worry that the floppy disk will go bad and this is basically multiple reasons for which can happen you can keep it on a hot sunny day keep the floppies somewhere else it will bend in some ways and you might find that you know no longer able to read it right?

The reason why cds fall apart is because they use certain dyes and the dyes can go bad. Organic dyes they use which can go bad even disks can go bad, because it turns out disks have sectors and they use some query substituted means of error management that are called ah error correcting codes, but in spite error correcting codes, sometimes you find that certain errors cannot be corrected and a sector goes bad when sector goes bad then you lost the data.

So, why the sector goes bad there are lot of reasons for it we will not go into it, but the reality is that sectors do go bad and we have to work out the things actually it turns out even when you build the discrete cell certain sectors are bad in the beginning itself and there is some mechanism already present on that disk it is basically does what is called the something called error list bad sector list and you map those bad sectors to spare sectors sitting outside somewhere at the edge of the disk.

So, all those things are already present in disks you will find that any big system any large-scale system for example, take any 3 tera byte disk or any solid-state disk. Which is coming out they have a lot of capacity for handling this kind of errors it is part of that (Refer Time: 17:09) that in addition to local disk level error management you need to have ability to handle errors that span multiple devices. For example, you might have a file which is let us say cannot fit into one device that field is going to split into multiple parts. One of the parts so, that one disks one part of that device can fall apart; that means, your file no longer may not be available in one let us say complete hole. So, you will have to find how goes to get around that problem.

Similarly, security right I think people store varieties of data some are publicly readable nobody cares, but some are sensitive you want to make sure that only 7 people can see it for

example, most of us have some passwords or whatever or pin codes that access sensitive information you might write it down somewhere I do not what anybody see it all right? Even if you look at for example, ssh those are few used ssh there is something called a private key and a public key. The product key has to be kept secret from everybody other than yourself only you should know it if somebody has figured out your private key what can you do you can for the message as if it is coming from you and send it somewhere and other person will believe that it has been sent by that party only not not the person who has stolen the private key all right?

The private key has to be kept secret. And there is some security model that has to be available. So, that the private key remains secret how is it done in Unix systems it is done by using it is got something called permissions right? Those of you might have looked at it there is something called read write permissions, right? And it Unix system has it is it is originally deviced this permission systems is deviced in the mid 1960's, and basically they thought of it as the following user has read or write permissions, then there is a something called a group and group now parties can have read and write permissions and then this other everybody outside and that model is quite good, but the other people will have also improve on it there is something called axis control list. Which is there in for example, more recent operating systems and that also gives security of a particular kind.

There are other type of security that can be thought about if access control is not applicable then it turns out that you might yet to use other mechanisms. For example, cryptography means in the case of the earlier model I was talking about the permissions model every access you make for the file is mediated by the operating system, right? The all the file system, but suppose the data it is not present in one system it moves across in one system to another system right this may one single guy who is able to ensure that it remains secure as per the policy this not one single person all right; that means, that intermediate parties can multiple parties can be there right for example, I am trying to access my bank account all right I have my pc there is the server that side, there are various intermediary devices all right I have to make sure that there is security at each level all right?

So, I might tend to use excess control will not work seriously here, but there are now models for that also, but typically cryptography is used here because that is not one single person who is watching all the sections from one side to other side end to end. So, same thing with storage systems if you have a large-scale storage system you might have to use both access

control models also you might want to use cryptography models, multiple security mechanisms. So, that the security policy at high level what has to be observed can be provided.

Now, there also systems which provide which want to guarantee storage systems you need to provide real time, let us say let us real time operation what it means is that suppose you are using video you want to make sure that you are data comes in between the required latencies. It does not come in then your video becomes not. So, interesting to watch audio is for example, much more stringent compared to video. So, there is something hard real time and soft real time. So, you can be a listen to music and you have to make sure that the data is picked up and given to you to your application the music player in the right amount of time.

So, your storage system also has to do it. So, there is a networking aspect the application trying to read the music file and create for you so there is a real time aspect get relating to the application real time aspect relating to the networks there is a real time aspect relating to storage, also all 3 have to be together then only there is some reasonable experience often times it is very difficult. So, what people do is they try to engineer a system sufficiently well that in spite of not being able to manage the real time aspect each of the replaces, but still can give you some reasonable performance that is you make sure that it is the the latencies of smaller f that only if the grow by large amount then you see that problematic aspects.

So, similar to that real time is a quality of service and these things are becoming important, but storage systems still are not current storage system are not very well geared, for this issues this 2 issues are not very well geared for it a good example of why this is having importance in the future, some of you might have heard about something called netflix what is netflix it is a big video it is a distribution consumer system right?

Basically, you can watch movies and they have all the videos all right on some servers and there are going to stream it to you. So now, your experience of watching that movie and typically these are all high definition etcetera; that means, they are much, much higher quality than let us say what has been so far right and; that means, that the storage required to store them is also very huge they are not small files they can be in the reason of tens of giga bytes and there has to be continuously streamed and what is more important is that the streaming is happening on a wide area network for example, the netflix servers could be sitting in one city and I could be watching in some other place.

That means I have to guarantee that in spite of whatever networking elements that come in between whatever software processing elements that can come in whatever storage elements that come in they all have to somehow deliver these packets needed for display at the consumer side right at reasonably good rate;so that the requirement for good video display on the consumer side is not affected.

So, that is our fairly complicated thing so you have to engineer some of the systems extremely well and also other more important problem is that the scale also is important if you look at netflix all right some of you might heard that now netflix is about supposedly 30 percent of all internet traffic, 30 percent of all internet traffic; that means, the it is really become extremely big even web is not has big as netflix.

We thought web is big right we all so many of us use it where web is nowhere whereas, a big as netflix (Refer Time: 25:25) I am sorry I made a mistake. I should said more correctly netflix is 30 percent of the web traffic survey that is, what that is a right direct, what to say it nowhere issue is manageability; what I mean a manageability is that when we have so many things around how do you ensure that things when they fall apart or they are not working well there is somebody watching it and trying to correct it this turns out to be a fairly tricky business.

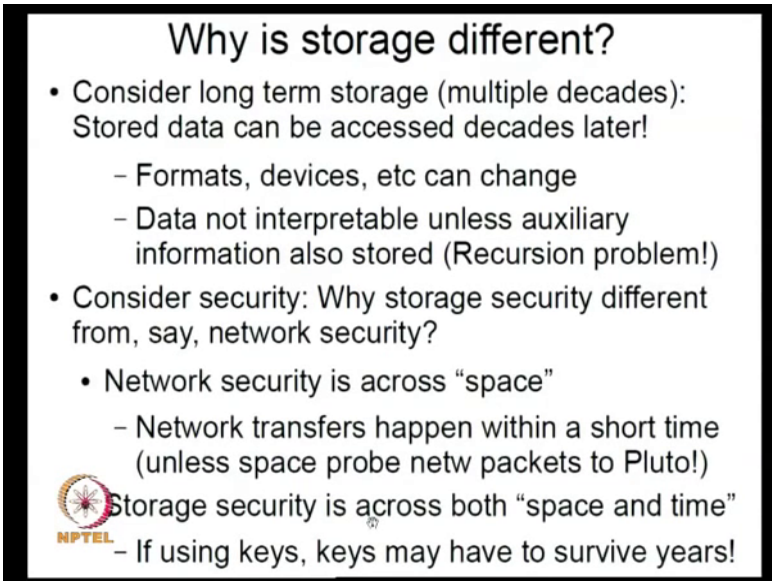
So, for example, you talked about error management all right? So, it may be that one copy of you to make sure that you do not lose anything you keep multiple copies of a particular piece of information if one of the copies becomes bad or it is lost right? You will have to use the alternate ones. So, somebody has to figure out where to pick it up from and ensure that that copies available so that is one aspect of it, the other aspect of manageability could be performance I say that I have a video thing to be shown I want a particular guarantees some body down the line has to ensure that this is given to you some has to manage it. So, that it could be lots of types of accesses somebody might be web traffic who does not care that much about real time aspect people do worry about fitness, but they may not be saying that I have to have it within let us say 100 milliseconds.

Do not be happy if it comes as within 50 milliseconds or 100 milliseconds or 400 milliseconds. I do not notice a difference if it comes in 10 seconds. I worry about it. Where as in the case of video I have to ensure that I get a frame everyone by 30 (Refer Time: 27:14) second I need to get one full frame that whatever stuff is there right I need to get that about

for every 30 milliseconds I need to get 1 frame. If I get it much larger in that particular 100 milliseconds or 500 milliseconds then the frame cannot be used.

So, somebody has to manage the system so thatwhatever objectives who are there with respect to the same is are being handling. So, storage systems also have big issues some scale.

(Refer Slide Time: 27:55)



Now, let us just quickly look at why is storage different now let us take just to you may be aware of many aspects to it, but let me post to you one or 2 things you may not have thought about, there is a ratio of what is called long term storage. What is long term storage? I want to store something from multiple techniques, why it is important? Some may writes some book now a days most books are being written digitally all right and I want to access it about 10 years later 20 years later 40 years later.

Now, with books one of the physically printed books we take it for granted, for storage it is still the unsolved problem I will give you one example, I did my PhD I wrote my PhD thesis in 1998 is written I am not able to because it is a some old form of postscript. I cannot print that postscript file except on a particular printer that is available in 1988, luckily I do not depend on that particular thesis I actually wrote a paper from another conference and that is the only thing I can give you it to you now I cannot give you my thesis whereas, some other parties you might have even wrote it in at least I wrote it in postscript old version of postscript in the 1980s early 1980s there used to be lots of other let us say editors available

both perfect star etcetera. And there was it is not easy to get hold of machines which run at software now.

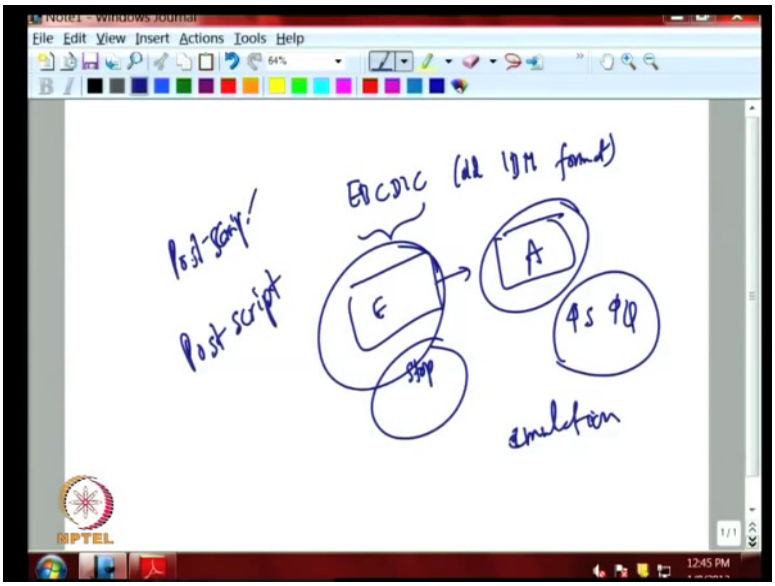
So, if you wrote a thesis using star or whatever it was it is going to be lot of difficulty for you to access them. Now it can be done only to do it is being what is called you can virtualize that application. And run it on a virtual machine using the infrastructure it needed to run star or both perfect in those days and run it as a virtual machine on your laptop current laptop it is possible.

Now, you requires that somebody has done all that work about how to have virtual machine which is able to run that operating system of that time. And with all the devices of that time for examples those days we had 18th floppies. Suppose you had a your thesis was on 18th floppy. That is all we have I want to see it on my laptop current laptop. Let us you can think of it is not trivial.

Basically first of all I need to find a physical device which can take that 18th floppy and be available to read the bits out once I get the bits out then it has to be processed by the various software which understands the blocking the way in which thus or blocks on thus on the sectors on the floppy device right? They all have to be readable right? They has to be partitioned they have to be cut into those pieces and then provided to the current systems then only they can actually make sense of these things.

So, basically our formats devices are certainly changed. And what is most interesting is data is not interpretable unless auxiliary information also stored, at this where a interesting ratio here why this interesting let us think about I think I will try to list at this see, in the past for example, you might have some character set let us something called

(Refer Slide Time: 31:44)



Let us call it EBCDIC. These are old IBM format I should not say old because there are main frame which still uses. So, I might have my this is just like asking this also some 8-bit code. Suppose I have written document in ebcdic with somebody has to translate from ebcdic to ascii somebody has to do it.

Now, if you want as simple as that because it turns out if you look at EBCDIC etcetera. There are various ways of saying stop because once upon a time this character things where being sent on certain telecommunication media. So, they used certain codes for saying stop a transmission, please stop the transmission with etcetera even ASCII has got this for example, in ASCII you have something called control S what is control S, please stop the transmission control q means what continue the transmission. So, there is various characters which have different meanings across this somebody has interpreted these things and map it to the new meanings or try to emulate them in some way. So, that whatever intent that was there here should be captured. Here so it is basically what is called emulation emulate it.

Now, this can be simple sometimes it can be quite complex. Sometimes and easiest thing to do it is to use virtual machines, but even then the complete involvement has to be preserved. Why is that? Because let us take the case of post script right? Now what is the post script? Post script is a a document formatting language and this post script has certain ideas about fonts it has certain ideas about how to let us say rotate something because it you can write some ah permutation this system by which you can rotate an object etcetera.

Now, it can also have certain models like color for example, it might say show me seven types of color because even the color is encoded, some of them might have 206 colors some of them have million colors etcetera right all those things are possible encoded here now; that means, that somebody some 100s of years from now, should know that post script has a certain model of a document. We should know that there is some concept of color which is that it is a 2-dimensional surface it is not a 3 dimensional (Refer Time: 35:08) it is not a rich 10-dimensional thing all right. Or even simple things like if you take a take a book it has got something called content section then it has got something called index section it has got some chapters (Refer Time: 35:21) right there is some structure to it and usually documents will have some information about that.

So, that what I would call as a meta data for the book. Similarly, any document if you take any post script document or pdf document or a dock document, it has a notion of such structure. And the structure depends on how you interpret the document; that means, the software at that time has that information in it is mind because of the code is written in that way and it knows where to look for where the index is where the bibliography is where the contents are so far it knows that where to look for so it can go on end.

But suppose I go to a new model altogether all right I need to have the information saying that this particular document has the index at this place has the contents here etcetera right that information also has to be transferred otherwise I can not make sense of it; that means, that there is some meta data about the document that has to also be transferred across, then only anybody who is looking at that can actually make sense of it

If that metadata is not available then I can not figure out what is going on all right? So, a sense what is going to happen is that suppose you just think that there is a floppy it has got anybody knows that what is stored in the floppy 0 and 1s sosomebody some clever physicist around he takes 18 floppy he is able to shine a laser or whatever on the thing he is able to see 0s and 1s. So, you get a string of 0s and 1s right, but what can we make of it can you make anything out of this 0s and 1s because you need to have know the structure of what was stored there in those days right.

For example you may have what is called a master boot record with a sitting in some place you need to know that there is some master boot record sitting, here the beginning the first set of might have that information, but you also need to know how big it is now a days we do not

know, we now a days we think it is all 512 bytes etcetera who knows it may be 512 bytes than that time also now on a floppy, but one can one can think of it that it could have been 256 bytes there is no reason why it has to be 512 bytes.

Actually, now a days you will see that disks are going to from 512-byte sectors they are trying to go to what is called 4 k sectors that is 4 thousand-byte sectors. It is not nobody has it is not happened completely people have begin to do it for efficiency reasons and other reasons people are going towards that; that means, that once upon a time it was more or less clear that any disk we had a set of servers 512 bytes, but in the future just about 5, 6, 10 years from now probably most of the disk that we will see will be 4 k byte.

So, there is nothing which says that in the future 100 years from now. That also will not change; that means, there is some information about sector size that is critical without that you can not make sense of whatever rest is coming. So, in a sense what is happening is that you want to understand a document it has some suppositions about how to interpret the document that information also has to travel across which it also very difficult.

So, that is what I was referring to data is not interpretable unless auxiliary information also stored what is auxiliary information things like sector size etcetera it is not; obviously, the disk is there I can see the disk and I see the I can see the floppy it is all I know it's all 1 0 0s, but I still I can not make out sense of it until I know what sector size is it used or size of matter whether it is used compression, it was just compression I need to know what method of compressions it was used. It is encrypted only to know what it is secret key was you do not tell many of this things I can seize; I can seize 0s and 1s, but does not help me because I cannot deactivate I can not see uncompressed it I can not do any of this things.

So, what is the real problem right now is that along with data I also need to store auxiliary information now the auxiliary information also has to be stored where is that who is going to store it, because Imy problem is to store the data, but now I have got a component problem of storing auxiliary data also there also I am not solved because it also has to travel across time, but how many days from now both has to be there.

So, it turns out to be a very difficult tricky problem nobody has solved it. So, the only solution is to keep changing from current format to future formats every. So, often keep changing it keep following the technology cycles of this. You had 18th floppy then your data and put it in to 3 and half inch floppies then put on to cds, put on to dvds, put on to blue rays,

put on to holograms. I do not know whatever put on to that put on to this keep on have bit. You do not do it the data will be lost because one day that gadget that reads it may be lost. Of course, you always have a the solution because finally, the bits are physical 0s and 1s somewhere right.

So, somebody coming back from using physics they can do it, they can look at the bit patterns they can see use some laser or whatever light they can use they can see 0s and 1s, but that does not solve the problem that we have to remember the fight that I consist 0s and 1s not solving the problem, why because I cannot interpret the 0s and 1s. You give me 1 million bits 0s and 1s I can not still say what it is, somebody has to tell me that the first few bits are this next few bits are this somebody has to tell me all these things that also some kind of information the meta data unless you also make that happen you cannot relate to those that is why it is a very hard problem it is not a trivial problem.

Similarly, if you look at security you will find that storage security is quite different from network security some of you might have heard about saying that you can use cryptography etcetera right? It is true we can use cryptography, but it turns out cystography is a it is a single mechanism it is not really solving the whole problem I think I already mentioned to you that if you are using a ssh we have to store the private key somewhere securely for cryptography you need this private key and public key.

Now, you should tell me that my private key is going to be again encrypted. If someone is going to ask you, where is that key? For that you have to store it somewhere. If you do not store that you now you can not decrypt it all right you can not keep on saying passing about to somebody else you store it from me safely finally, you have to stop the recursion or a bush the problem and say I am going to store it using non I am not going to cryptography on those kind of methods. I am going to use it some other methods. It has to be physical separation mechanisms access control or a write it in a piece of paper and I only can see it nobody else can see it I keep it in a pocket and I ensured I got it all times nobody can look at my pocket or I keep my pant somewhere I want to make sure nobody can touch my pants when I am taking a bath also I have to do all those things.

Again finally, what is it is a basically physical separation you can not keep saying security we can not keep saying cryptography because the minute you say cryptography, somebody has to tell me how to manage the keys. There are new techniques people are coming with

cryptography call identity base encryption, but even then there are it has not at become completely well-established there may be some methods it look a bit difficult right now to believe that it will happen, but any way people are doing some (Refer Time: 43:18) but what I am trying to suggest is that you need a security for some aspects of data that you use, and it is someone different from network security let us see why it is different from network securities across space that is you are sending it from 1 place and go somewhere place.

So, it happens fairly quickly for example, you want to talk to somebody securely you leave the phone use a secure channel you tell other person something some secret all right. It happens with in a short period of time. And so, the piece of information that has to be protected it is only for short period of time typically within seconds sometimes at the most few minutes etcetera not more than that. I am sending a packet from here to sending a mail from here to some other place in us also that packets will take typically few seconds sometimes hours, then in very bad situations some are probably a day also sometimes it happens, but usually not beyond.

But what is important is that during this 1 day or 1 minute or whatever it is my protocols do not change my machines, do not change that much not of the same (Refer Time: 35:08) whereas, if you look at the previous problem if I do it across 100 years everything might have changed if that also is some sense a type of data transfer I am sitting here in 2013, I want my data to go to somebody in 2113.

That also some kind of transfer that during this period all kinds of things could have changed the liner could have changed all right for example, about 200 years ago many people in india new Sanskrit reasonably well many educated people now a days hardly anybody knows Sanskrit right? So, if you write it in Sanskrit that time nobody is going to understand what is going on all right educated I am talking educated people.

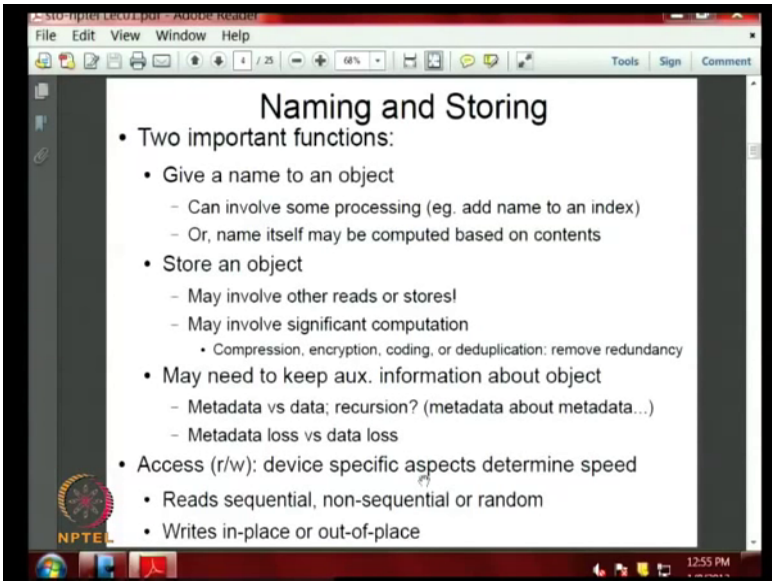
So, same thing about any language for that matter we old Latin old Greek they kind of the current people the Modern Greek people cannot understand old Greek. For example, in some they can figure out a bit, but not much old Telugu a new Telugu all those things languages (Refer Time: 45:39) changes, the protocol change, the machine changes, operating system change, everything changes, their encoding change all those things. So, that is why you will find that this since the securities across not across space in material storage it is actually across both time and space, both across will be there are lot of issues that had to be taken into

account and keys have to survive also multiple years. It can not just like that say that the key is new you have to find the way in which the keys all survive.

If your keys are not available then you can not read it and that was already happened to many secret documents all right? Many secret documents have been encoded and the keys the have been lost and nobody can decode it. Any example of a well-known example of undecoded systems there are many of them if you look at well it is called Indus value valley civilization right they have written something, but no none of us are able to figure out what it is. They have written it very well I can see the if something like in physics right you can see the bit patterns the patterns are visible, but I can not mention what it is.

So, in storage systems they are very complicated problem because the keys also have to travel a long time. So, both auxiliary information, like sector sizes etcetera meta data and keys that also is some kind of meta data about the fact there also (Refer Time: 47:12) if you do not have this 2 things it is not possible.

(Refer Slide Time: 47:15)



I think we will look at the next issues. I know probably that this point at end of this (Refer Time: 47:21) will probably start today that we will see. So, what are the issues out here? If you look at storage system there are 2 issues that important ones what is called naming and storing it looks very obvious, but the ((Refer Time: 47:34) deep things here also. What is the aspect of naming given internal object? This looks very trivial all right.

So, there is piece of information you give it a file name all right file name file contents. So, if you give a name to it that might involve some process because you putting it into a directory; that means, somebody has to put one more additional entry in to the directory right we have to increment a number of files out there in the directory somebody has this in process minor processing, but you have to do it in process. Sometimes your directories can no number of files can be huge especially if you look at a mail files he use something called mail dir format mail dir format. It saves every single file or every email of yours as a single file.

Now, I personally know that I have about 17000 preserved mail as if now since about 2002. I think I have 17000 piece of mail that I were sitting in my directory. So, there are 17000 files in that one directory how many 17000 in one directory. So, that I am going to search something. So, I from searching something what would happen? I am going to search 17000 items. So, I can use fast techniques to find.

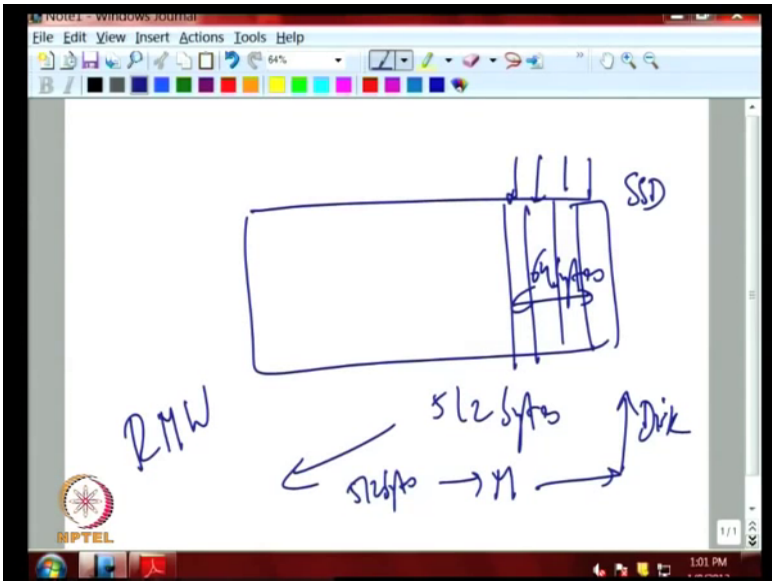
So, if you want to use first techniques typically I need to do some possibly hashing or a some sort the or some index I have to take right. I have to keep indexes they were like an look at if I could use for example, battery then I have to if I add a file I have to change the battery; that means, that just because I gave a name to an object, does not mean that it is a very simple operation it can mean that I need to do additional process. In addition, I can give a name through it contents for example, somebody has a file which is let us say 257 bytes others can do a hash of the whole thing. There are various types of hashing possible unlike an computer hash on the 2 to 7 bytes I get a what is called a summary based on that hash unlike you use that as a index.

If you use a good hash it turns out that they have good what is called anti-collision properties. What is that mean? It means that given any document it is extremely difficult to find another document which has the same hash. There is something called strongly collision resistant; that means, somebody has to really work very hard we have to solve some extremely hard problem. So, that I can get another hash another document which is same hash; that means, that if I use the right hashing model our distinct name for every file all I have to do is hashing. So, some people use this also. So, it is another method and this is very useful in some situations we will talk about it later where it is used.

So, one is giving a name I told you it calls some person storing an object. And storing an object also may not be that simple may to know may involve other reads and stores why

should storing an object require a read it is possible in some situations basically because I will give an example,

(Refer Slide Time: 51:27)



Suppose this is a unit of axis and a disk for example, it can be 512 bytes. I want to write only 64 bytes this part, but for all for me to given the interface the disk I have to first read this 512 bytes into memory put into memory and then update this 64 bytes because memory allows me to write it in units of word size I can write it in for example, so many I can update this part first followed by this etcetera you update it once I have everything in memory then I can put it in back into this category.

So, I get in from disk 512 bytes then update it this is basically what is called a read modify write cycle read modify write cycle. So, even for writing sometimes I need to do reading and this becoming very common in the case of solid state disk etcetera, nobody should find that most of these systems also require this kind of read modify read cycles. So, again it may not be the that simple also because it can even this can be involve significant computation because I want to write something you want to compressed it I have compressed file systems. Which can help somewhere compress it and might have what it is called secure file systems you might have to encrypt it I can do coding or I can even do what is called deduplication.

What is deduplication? I am writing something and it has some connection with other files which I have there is some sharing that is going on for example, suppose somebody is updating the number the states of to this country number of states in a country is has been

changing the last 1 or 2, 3 decades right.? So, I have a new version of the states of the country right. So, it has got some relation to it what was where in 1960s. So, that is some kind of it is similar, but there is some difference. So, it may be possible for me to write the newest information sorry delta of the previous information.

So, when I writing it, I will see if the data I am writing it is it similar to any other data that I have already have, can I write it as a delta of the previous information. That will essentially remove the redundancy; that means, it is a may not be a trivial operation because now I have to find out I am writing this piece of information.

Now I have to check against all the stored objects not one I might have million objects in the system I might have billion objects also I have to go and search all this thing if there is any similarity between the object and storing and what all it exist and then if it exist then I tell myself I will go to divide delta of the difference and then I only store the difference and keep a point into the original information that is also I can do.

So, but this turns out to that this is computationally can be quite expense, but this is being done actually it deduplication is now quite common, if you see current storage systems they do deduplication, because it reduces lot of duplicates in the system and I think some of might has notice it already basically because suppose I send a mail all right? I am attaching some file all right. So, the file exist in my directory and has the file I send it normally when I send my mail I keep a copy of the mail I send right; that means, even attached file is also be stored as part of the mail that has been sent out so there is the copy of the original document in a file system plus the same copy in my sent mail also.

Now, people have found that; as much as 80 percent or close to that is duplicate so, we talking about 100 peta bytes and somebody can tell you that 80 peta bytes is you know duplicate right? People will say what a waste to put 100 peta bytes I will work with 10 parameter that is what I need because the cost of this 100 peta bytes is substantial it is huge it is both in terms of space in terms of energy also it is a huge where we are talking about 100 time 10 times your energy consumption. And you know that keeping computer systems running is a costly business in terms of the energy we are talking about 10 times energy consumption of of course, the storage systems 10 times.

So, I think we will continue from here next time. I come back to this point I think basically I just was giving you some high level ideas about storage systems why they are interesting in

their own right, why they have become separated from processing. So, will continue from here about starting with meta data versus data etcetera will continue from here next time on words.

Thank you.