Non-Volatile Memory Hardware (00:08:10)

The the model that you have there that the design that you have there is appropriate generally when you're storing your data 25 when eventually the SSD will goes to a file right or yes and and then you also need to think about because remember one of the things that we want to do is a remove that constraint of having to always write to a file and change the video and so if I want to directly if I want to directly write to the file or control access to the file myself you need to ask yourself a few questions how it can happen and whether that will impact this the way your SWS represented if you need additional information and then the second question is I think I shared the link puts you on the persistent memory as well and so if we for example decide to use a persistent memory non volatile memory if we decide to use that kind of implementation that means that we're no longer considering disc as a storage horse but we were sticking to right and so if we're sticking to memory what kind of how can we represent the SST so you're in your mind there should be a couple of questions whether they should be the only structure SST will weather changing while you're changing the nature of her medium where you store in the actual data it might change to the structure of the SSD OK yeah I think in mind that compaction that when you're thinking about that change compaction should also apply here because somehow you want to use compaction to reduce the size of your system how to reduce the number of tables and then in there for you also need to think about that

Compaction in LSM: (00:12:45)

Or does it only occur within the estables well generally people curse with the estable and but you could also merge because if you take the member member the memory table demand table uhm and you do a compaction then you have to flush them anything OK see you there the ones to do the practically usually you once you flush the end table the men table you push it as is as an S and other essays table and then you have a bunch of tables cancel the reason we do a contraction is because the same key right appear in several assisted groups so you need to come to to do a compaction to reduce the number of locations where those keys might appear and to have a consistent response to a query OK good it's that's it that's precisely why Ellison trees are not even not really appropriate for reading tensive applications because as much as you can write fast to read you'll have to go through all the SS tables disk where you have key before you can give the response back.

In-place update in Persistent Storage (00:15:35)

Like this segment will appear no this work when you're doing a and in place update in our case we're doing a persistent storage we use that we don't delete we don't delete any version yeah keep older version so it query to the system actually should indicate possession of the data that one wants to read millimeter no vision is being deleted yeah is being removed right we keep all these versions.

Versioning of Data: History of updates and deletions ()

Acyclic Graph To Illustrate Version Of Data Object (00:22:15)

So so I think I think you should come I don't know if I made this comment before but I'm not sure if that's information at binary information 10 really helps OK what I'm doing is to each key a super hit a directed acyclic graph that shows the vision of the objects attached to that key so in your example object one has so far got only one value attached to it so I would that would have one single node yes if you take the key to our doc will have two nodes first you have to value B which has a simple question number and then you have the deletion mark which has certain version number so you request comes in to read the key with the latest latest version then you have to decide since there's a deletion mark on the latest version am I going to serve the previous version or am I going to indicate to the the user that visit that object has been deleted but their previous versions? Then if you if you take 3 you should have the same thing so you have your dad again you should have it load which has the value see women's vision and then that's that's value moved to another value which is the second node CC will be special and so using the deck you can always serve to the node at the end of the the that by default that would be your one and if too if two requests indicates an explicit version number then we go and look for that version.

Distributed Storage Engine Implementation (00:30:25)

Try to adapt it to a multiple nodes on a class are exactly so So what I like to see is how does this work when we have several nodes and if we have several nodes are we going to do partitioning are we going to be shutting the data already and specialized nodes or are we going to have a replication where all of them have the same copy decisions need to be made and in order for us to make that those decisions we need to understand what it looks like yeah marker OK no no idea would be to go back to your under sentry for example for example and then take the actual representation of the object which yes those those objects that you have would be represented now as a dad and the question would be now we have several nodes managing these understands how do we handle group milk.