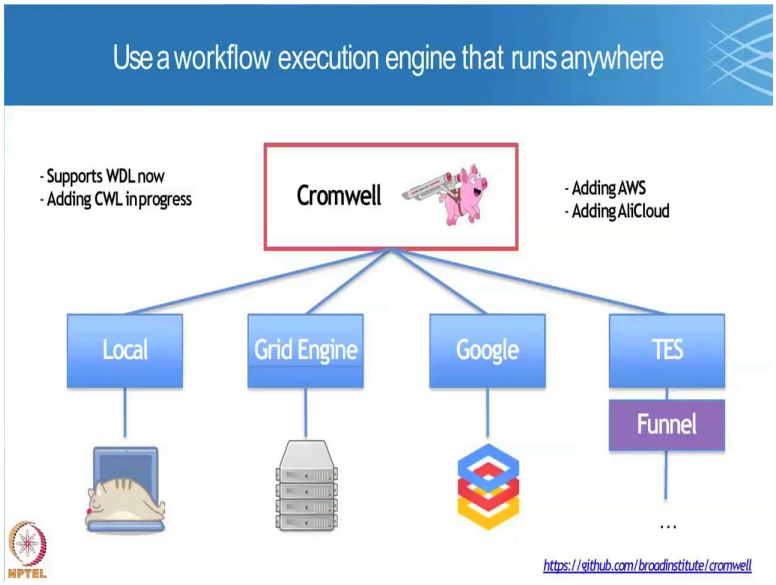
Introduction to Proteogenomics
Dr. Sanjeeva Srivastava
Dr. D. R. Mani
Department of Biosciences and Bioengineering
Indian Institute of Technology, Bombay
Broad Institute of MIT and Harvard

Lecture - 27
Introduction to Firecloud and Data Model

Welcome to MOOC course on Introduction to Proteogenomics. In the last lecture you were introduced to the concepts of Cloud computing, once the workflows have been set up the next step in automated data processing involved in providing input data and collecting output data. Today's lecture by Dr. Mani will introduce you to the steps involved in execution of data analysis tasks on Cloud platforms. So, let us welcome Dr. Mani for today's session.

So, now you have your workflow description language that has told you what your workflow is, what tasks are in the workflow and how to string them together and then I have told you that the algorithm that you are going to use for that workflow is encapsulated in a Docker.

(Refer Slide Time: 01:14)



So, now how do you give it the data; how do you get output; because in the cloud computing thing everything is on the cloud nothing is local. So, if you have a big data set here how do you give it and how do you basically run the whole thing? So, that is kind of

the next part of the presentation where; so, you start with the workflow and then there is something called an execution engine. So, the workflow is given to an execution engine which is generally a flying pig for FireCloud.

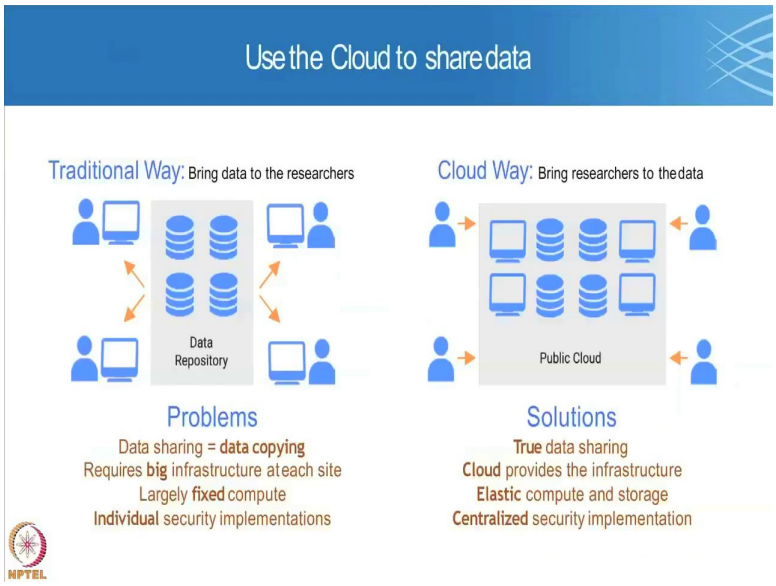
So, it is called Cromwell and I have no idea why they represent it as a pig with rockets on its back. So, maybe it is nice to see pigs fly but the execution engine is called Cromwell.

So, what Cromwell does is Cromwell interfaces to all the actual computing hardware. So, if you are using Google Cloud Cromwell knows how to interact with Google Cloud, if you are doing it on your local computer, it knows how to interact with your local computer or if you have your own grid engine or other local cluster computer that you have, it can run on any of those.

So, this provides like a common platform on which to run your workflow descriptions and your workflows without worrying about what the underlying hardware is and how the underlying computing infrastructure is recruited to do the operation.

So, you basically give your WDL to Cromwell and then it will kind of run wherever. So, I think right now Cromwell runs on our broad grid engine, it runs on any locals you can run it on your computer if you want and it runs on the Google Cloud. They are trying to add Amazon services it is not ready yet, but it is something that is in progress.

(Refer Slide Time: 03:00)



So, how do you use the cloud to kind of do cloud computing? So, I think I have mentioned parts of this before, but I think it is worth going through again. So, here the traditional way is you bring data to the researchers.

So, you do your sequencing, you do your proteomics, you have a data repository, researchers download the data. So, here data sharing is basically you are making copies of the data. So, making copies is usually not a good thing because, if you make a copy and change it nobody else knows that you changed it or if the original people who put the data in the repository, say oh, I should have done this differently or there was something that was wrong I am going to fix and then they put it there.

People who copied it do not know that it happened; so, you need to send an email and then they have to kind of figure out whether they want to get it or not. So, data sharing is data copying. The other thing is it requires big infrastructure at each site; so, to do like genome alignment it takes a lot of computing power and so, if your institution wants to do genome alignment, they need a big computer and the other problem with this these local traditional set up is the fixed nature of the computing environment.

So, you bought like say 500 nodes in your cluster computer. Suppose there is a job that now requires a 1000 nodes, you cannot run it at once, you have to split it into 2; do it 500 once and 500 next time or let us say you suddenly got lot of users that are requesting resources. So, this workshop for example, all of you connected to some the IIT Bombay computer and said I need to run protigy that is it is going to be hard because there is only a limited set of resources.

So, some people will get the resource others will have to wait and the other issue with a lot of this data is when you have genomics data with germline mutations and stuff like that at the US, Europe and I presume India also has a lot of restrictions on how you can make this data available.

If you have identifiable data you cannot just like put it out for people to take a look, you need security and also when you are working on your project you do not want your competitor to figure out what you are doing because when it is on you are on the cloud or you are local you have to have your own security.

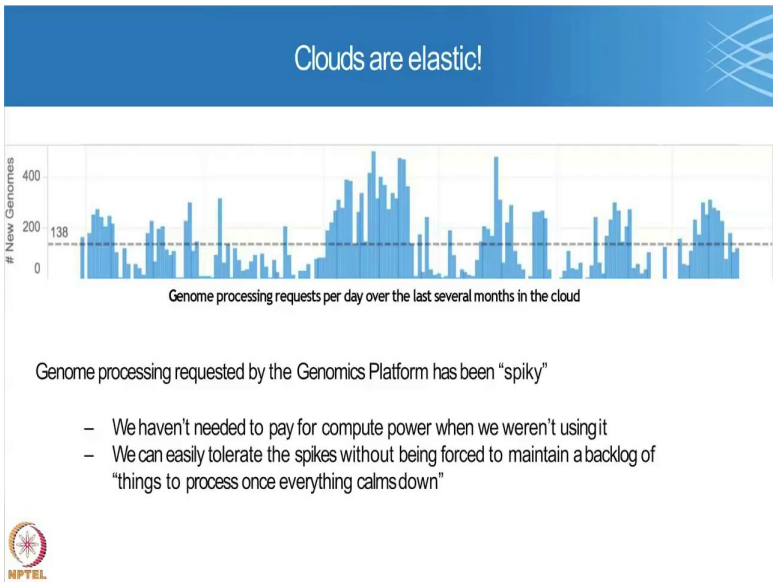
So, somebody has to kind of make sure that your computer does not hack into the IIT Bombay computers and take your data or figure out what is you are doing what you are doing. So, you need a kind of a relatively robust IT department to take care of all this. In the Cloud computing way data sharing so, you bring the researchers to the data.

So, the data is sitting on the cloud and then people go to the cloud to do what they want to do with the data. So, the data is like there is only one copy, you do not make multiple copies; so, you make any change everybody sees it. Look and the next thing is that the cloud provides the infrastructure. So, I think I mentioned this before I will not be label it.

So, you do not need to have your computers and maintain it and so forth, but the biggest advantages that it is elastic. So, for 5 hours or for 5 days when you are running this workshop, you need a 500 nodes you can get it because Google has hundreds of thousands of nodes and they can give you a few more if you want and so, you do not have to now wait for resources, you can get whatever you want instantaneously it is very elastic and security is now centralized.

So, if you your access to this data is now centralized. So, if you say only people who have gone through some kind of a data use certification can access this data, you need to implement that in only one place and it takes care of it for everybody.

(Refer Slide Time: 06:51)

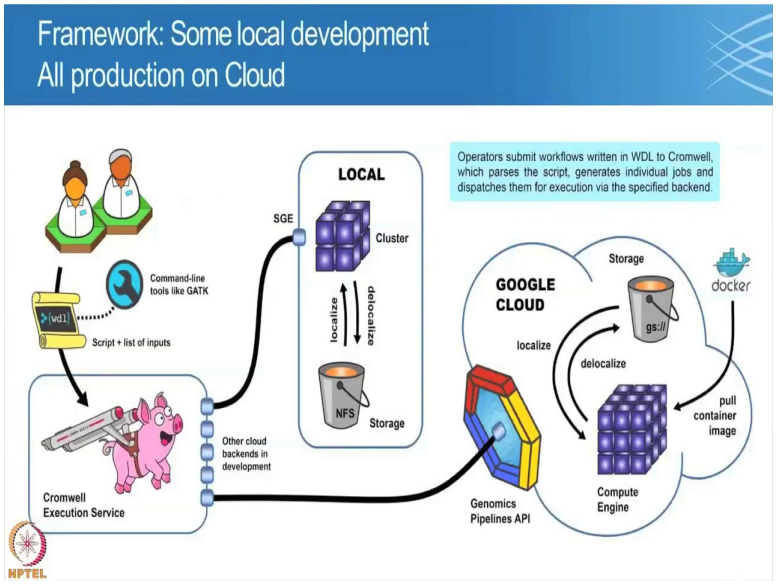


Here you would have to each center that download the data who is responsible for implementing its own data security. So, this is a example where shows the actual genome processing requests per day over several months in the cloud. So, you can see how much it varies.

So, I guess here somebody got a bolus of data from their sequencing experiment and now they want to do alignment. So, they request a lot of resources, here nobody is using maybe it is vacation whatever not much being used.

So, if you are on the cloud then you pay for only the things that you use but if you have your own computer you have already paid for it whether it is being used or not and usually at least at the broad we have found that using Cloud computing resources is like significantly cheaper than using broad IT. We have told them that and we are trying to migrate away from broad IT.

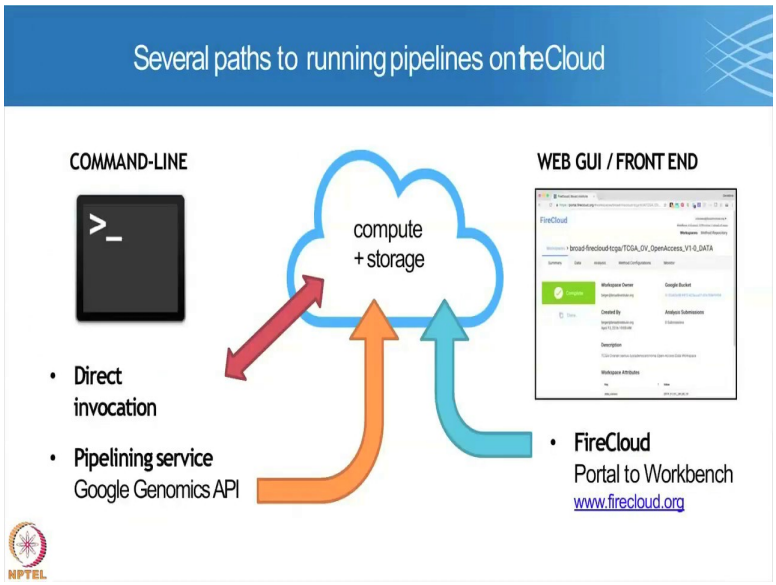
(Refer Slide Time: 07:48)



So, when you are working on FireCloud the way you develop code is to you work on your local computer, you develop your workflow description and then you have your inputs that you want and then you use Cromwell to test your code, your workflows and you can test on your local computer and when you are ready to deploy you can deploy it to the cloud and then it will run on the cloud.

So, that is kind of the model that is being used and so the that is facilitated by the fact that the Cromwell can run either on a local computer, on a cloud computer or on your local cluster.

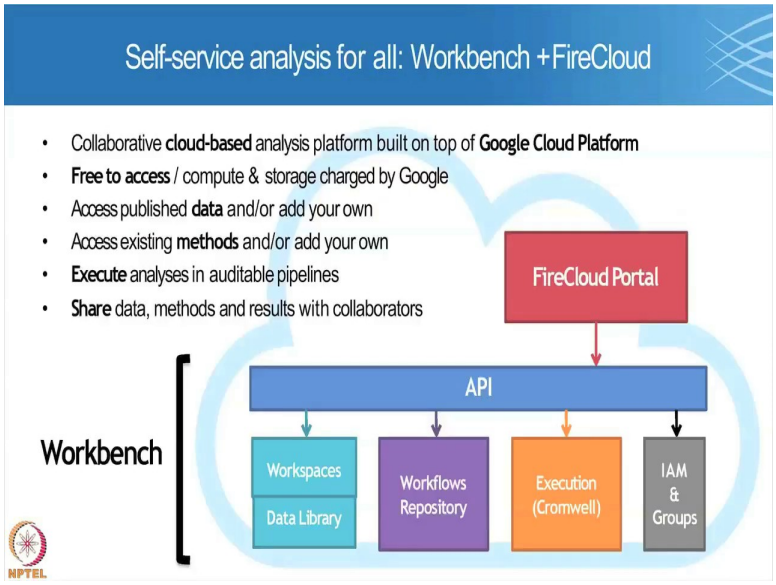
(Refer Slide Time: 08:29)



So, you when you have workflows that you have written, you can invoke them in several ways. So, one is you can directly invoke them using the command line, the other one is you can use pipelining services from Google genomics or you can use the FireCloud portal.

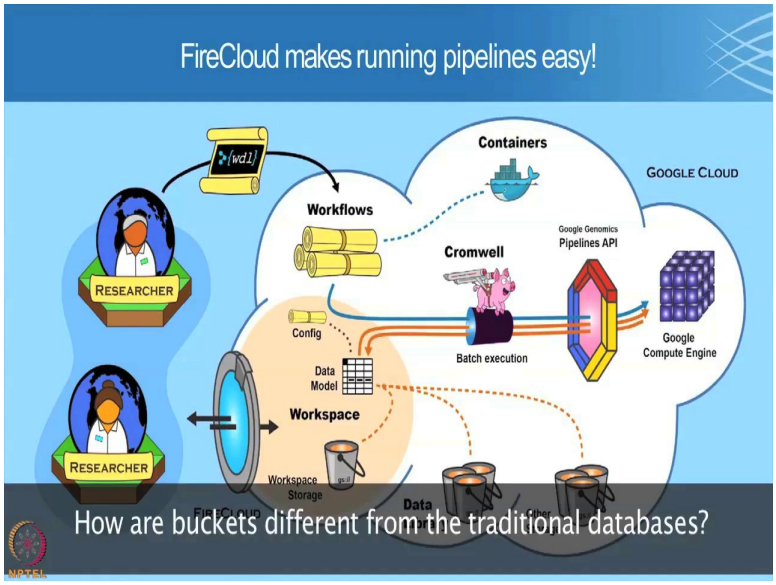
So, I will I will the demo I will give will use the FireCloud portal and it makes it relatively straightforward to kind of take other people's methods, apply it to your data and kind of look at results.

(Refer Slide Time: 08:59)



So, I think this just say it is the same thing in more words. So, the there is an API that you can use to access all the workflows, the workspaces and so forth and, then you can either access the resources using a workbench which uses the API. So, you type a command and then it uses the API to get what you want or you can use the FireCloud portal.

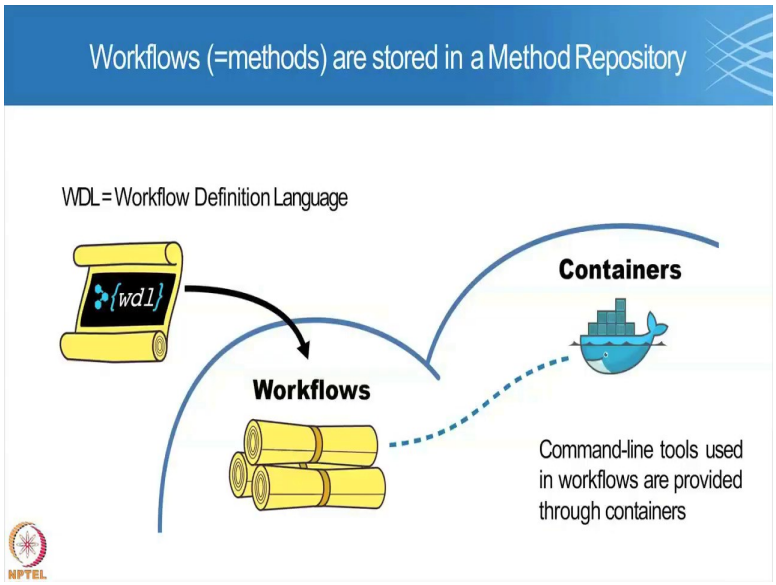
(Refer Slide Time: 09:25)



So, this is kind of the overview of how FireCloud works. So, you have researchers, some just use FireCloud, some actually write workflows. So, when you have a workflow, you

have workflows along with the configuration which says what parameters I have chosen in the for the various options and where the inputs come from and then you use Cromwell to execute it on the Google Cloud and then you have containers that provide the algorithms.

(Refer Slide Time: 09:59)

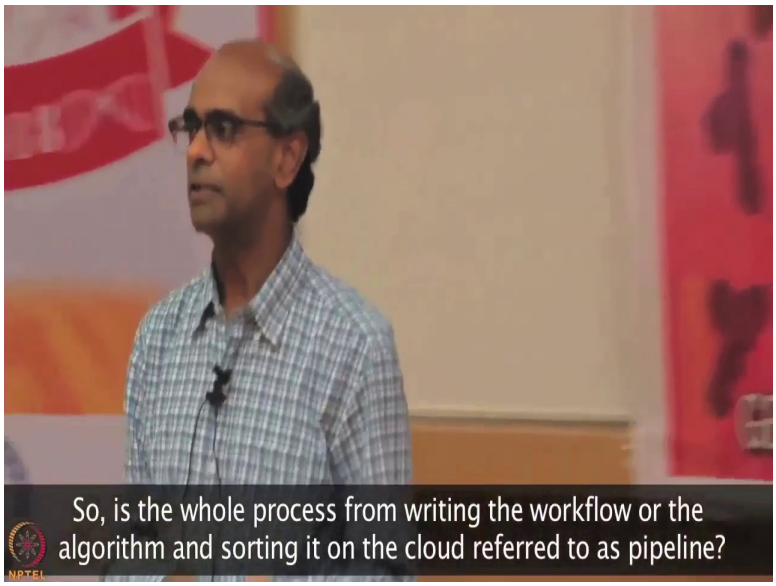


So, the one other thing that is mentioned here is it is the next slide, no, that; that I will kind of briefly speak about is. So, when you are running on the cloud and you have data that you want to apply your algorithm to; where is the data, is it on your laptop or where do you put the data?

So, that is where in the in any Cloud computing environment you will also have a data storage that is part of the Cloud computing environment because if you have data on your local computer or laptop or somewhere else it takes a long time to upload the data to the cloud because, if you have a 50 gigabyte data file it takes a long time to use the internet to kind of upload the things.

So, you so if you did that your algorithms will run very slowly and will spend most of the time waiting for the data to be available. So, you create these things called buckets, I think Google calls them buckets, Amazon calls them something else I think. So, you put your data in the buckets.

(Refer Slide Time: 11:00)



Student: Sir, writing the workflow or the algorithm and then sorting it on the cloud. So, this sorting is called the pipeline.

So, the pipeline is the kind of sequence of algorithms that you want to execute on your data. So, for example, we have discussed about a lot of ways you can analyze data right. So, you say I want to do these five things. So, I want to pre-process my data, I want to remove missing values and then I want to do a two-sample t-test and then I want to visualize the results.

So, there are four operations you want to do on your data. So, can you visualize the results first? You can you start by visualizing the results? You can't, you need to do the other analysis before you have something to visualize.

So, there is a sequence in which this has to be executed. So, basically taking your data and running it through a sequence of algorithms; so, this sequence of algorithms is called a pipeline right.

So, they have to be executed in sequence and so, that string of algorithms is called a pipeline and you can use it for just genomics, for just proteomics, for proteogenomics for all kinds of things. In the in the demo I will show you some of the pipelines we have for proteogenomics and what they do but they are basically a sequence of algorithms that you execute in order to do a lot of analysis.

Any other questions?

Student: Question

Ok.

Student: Sir.

Yeah.

Student: So, when we talk about bucket; so, when basically when bucket we use like a shared as an analysis and we use storing the data in that.

So, your question is we where we store the data? Yeah.

Student: For bucket may be talk about bucket.

Yeah.

Student: So, for bucket it is also like a snapshot of the data.

So, you are asking what buckets are?

Student: Yeah.

So, I will go into that a little bit more. So, buckets are basically like the hard disk on your computer. So, it essentially stores data that is used for your analysis, it also stores results that come out of your analysis. So, any data that you would usually store as a file on your hard disk in your local computer would be stored in a bucket now, because the Cloud computing environment does not have easy access to your laptop and so, basically all the data is used from a bucket and results are put back into a bucket.

So, let us say you are working on like a project where you want to use TCGA data. So, you want TCGA breast cancer data for the 1000 samples that has already been done and you want to combine that data with some data you have and do an analysis.

The nice thing about using Cloud computing environment is that public data like the TCGA data are already available. So, in FireCloud there are workspaces meant primarily to make data available. So, TCGA data is already available in a bucket; so, I think that is

what is marked as a data library here. So, all the TCGA data from all the cancers are available in buckets, you just need to know which bucket it is in.

So, once you know that you can link that bucket into your analysis and then you have your data that you want to include. So, that goes into your workspace sorry, that goes into your other storage. So, you put your data here, public data that is already available in on the internet and other places has been uploaded here and made available to you. So now, you take those two, you run your algorithm and then it generates some results.

So, the results are temporarily stored in workspace storage and then when the pipeline is done executing, it takes that and puts it back into your regular storage. So, that you can log into your bucket and then download the results if you want.

Student: But, how it is different from the traditional databases like buckets or are it is the same?

It is the same concept, it is just same files. So, yesterday David had a file for loading your gene data right. So, that file now if you wanted to do the same analysis on the cloud, you would put it into your bucket because that is the only place the workflow will look for data.

Student: Sir, is it something like the folder you are keeping in?

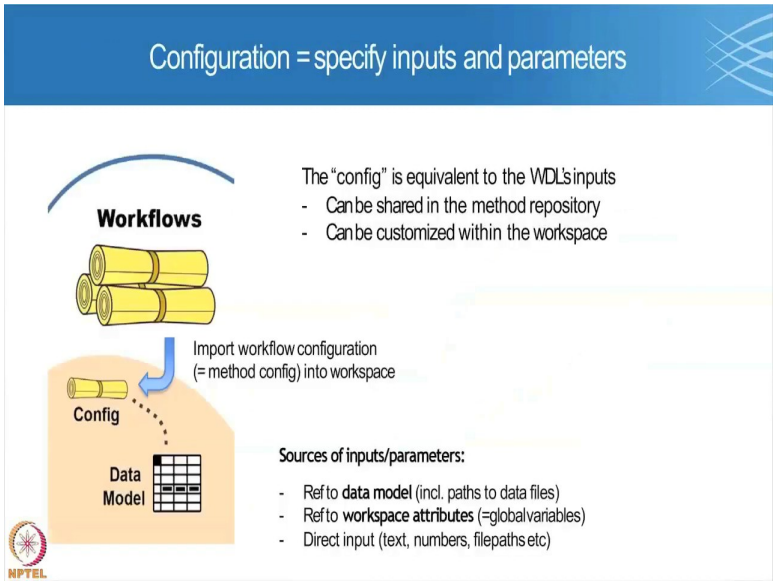
Yes; so, the actual physical way they store data is slightly different but you can think of it as folders with files. I will show you how it looks on when you go into a bucket when I go into the demo, but it is basically folders with files ok. Any other questions? Ok.

So, on FireCloud all the workflows that you create are called methods because, they are usually algorithms or some kind of tasks that you want to do analysis that you use for doing analysis and they are described using the workflow description language. So, all those are stored in a method repository.

So, the method repository has a collection of workflows and each workflow may get its method from one or more containers. You could have three workflows that use the same container, but they use different settings or they use the container in different ways or they may use different versions. So, you can have one container that goes into multiple

workflows or you could have different containers for different workflows so, it is very flexible. So, workflows are basically stored in the method repository.

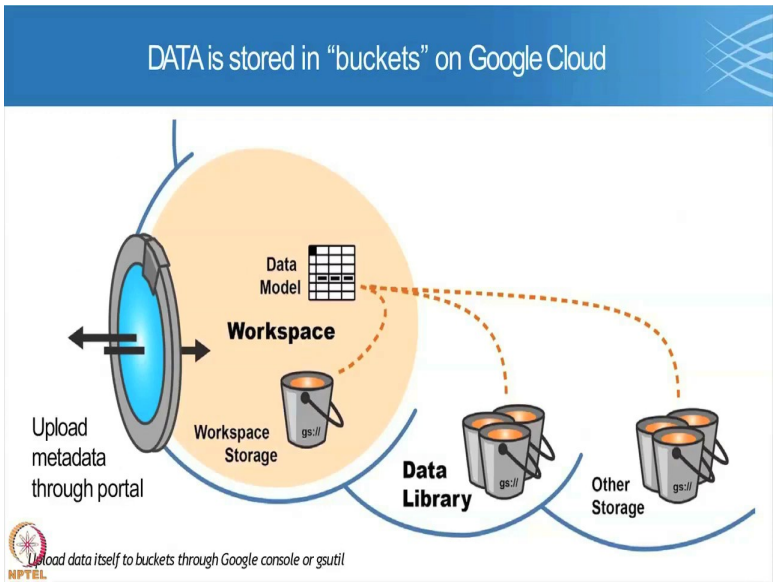
(Refer Slide Time: 16:40)



And, then to make it easy to apply it to like biological samples, in a biological setting there is also there is this thing called a data model. So, the data model specifies like house samples anda aliquots of samples and participants from whom you get samples are all kind of put together. So, that you can apply your algorithm to specific types of data but here I think what we are trying to say is that workflows now have inputs that you want to provide.

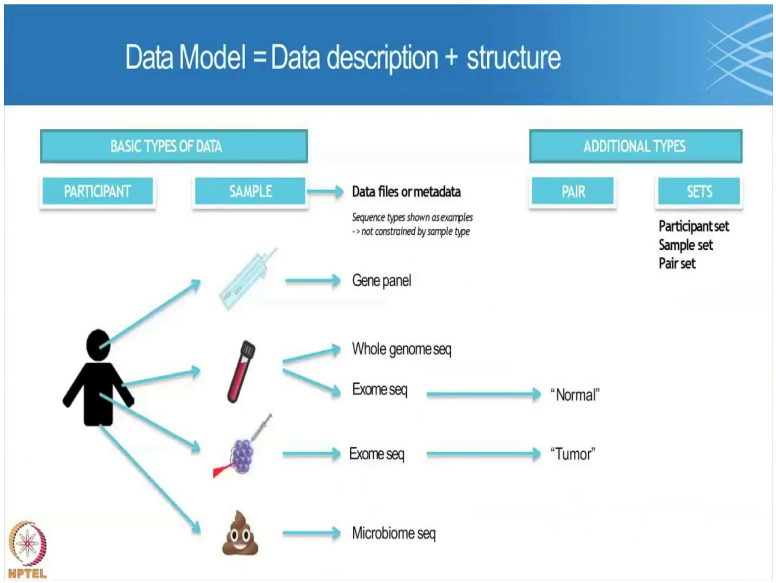
So, you what is the value of your threshold; is it 3 or 5 or 7? What input data set do you want to use? So, those are all configurations for your workflow. So, a workflow add in conjunction with the configuration with knowledge of what kind of data you are working with provides like a configured method and then you can run a configured method using Cromwell.

(Refer Slide Time: 17:44)



So, I think this kind of provides more explicit detail about the buckets. So, you kind of have a portal which we call FireCloud that interacts with the Google Cloud platform. It uses the data model to access the various data storage and data libraries that are there to provide inputs and outputs for your methods.

(Refer Slide Time: 18:07)



So, this is the data model I mentioned. So, there are essentially like two basic types of data you can deal with. One is a participant which is essentially you can think of as a person providing one or more samples and then you have a sample. So, you could have a

gene panel, you could have like actual blood sample that results in a whole genome or exome sequencing. You could have tissue that you got, you could have other kinds of samples and they all result in more data that that are now made available as data files.

And, then there is metadata about those data that is stored along with the information about the participant and the samples, that you can use or update for your analysis and then in order to enable some sort of automated batch processing.

So, you want to align whole exome data for a sample and you want to do this for your entire study, your study has 100 people. So, instead of like invoking it 100 times on a sample, you can create a sample set and then you say I want to run my alignment on the sample set and FireCloud will automatically go, it will create 100 jobs; one for each sample and then do everything in parallel and get things done quickly.

You can also have sample pairs so, suppose you had a control and a tumour from the same patient that would be a sample pair or in sequencing you get a normal blood sample along with the tissue sample to do mutation calling and stuff like that; so, that would be a pair. So, you can set up these upfront in your data in your metadata table and then use that to provide input to your methods. So, that you can configure your method properly and run it.

Student: Sir, sir in this case when we are making sets.

Yeah.

Student: So, if we are having multiple conditions.

Yeah.

Student: Like severity, non-severity entity. So, can we make three different sets?

Yeah.

Student: And, can we compare them like two sets at a time and pair with the third one like that.

Yeah. So, it all depends on how the methods are set up. So, you can create three different sets with different subsets of your data that is definitely possible. You can run specific

algorithms on each set but if there is an algorithm that is that allows you to compare two sets then that would take two sets as input and then you can use that to run it.

So, it all depends on how the methods are written and what they expect as input. So, suppose there is no such method and you want to write one then you would say my method takes two sets as input and then I look at the difference between the two sets to do some sort of an analysis.

So, right now in the proteogenomic pipelines we are not using sets in the right way because, usually when you do some sort of database searching or some proteomic or genomic analysis; we end up with a table that has all the samples and genes or proteins listed in a single table.

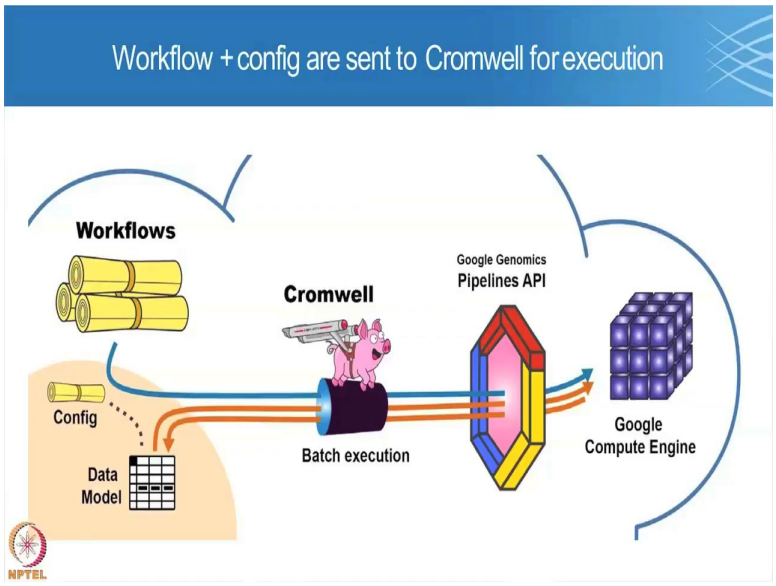
So, as of now we have kind of hacked this thing to say that that whole table is a single participant and then we just use it on the whole table. So, if you had two subsets in your input you would say what your subsets are but the data model would just think of it as one participant.

So, it is kind of not the ideal way to do it, we are trying to use the actual sets and the problem is that when you get output from spectrum l or some proteome or data preprocessing software you get it as one table but when you do genomics and you do like exome sequencing and you get copy number or mutation data you get one per sample.

So, this is kind of tailored towards how data comes out when you do genomics but in proteomics it is slightly different because, if you do a TMT-10plex you cannot separate those individual samples and say each one is one sample but the whole block is one sample.

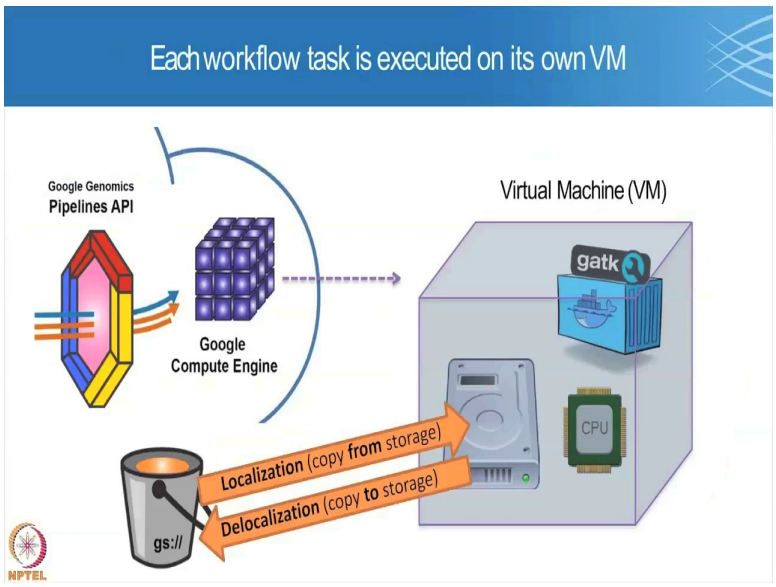
So, we are trying to figure out what best way is to kind of take that model from proteomics and map it to the model that is already here. So, we are not done with that yet and that is part of the reason why this is still not fully public. So, we are still working on some of the aspects of it. Any other questions? Ok.

(Refer Slide Time: 22:43)



So, I think I have mentioned this again previously I will just. So, you have a workflow, you have a configuration that says what the inputs are, you have a data model that provides those inputs and then you send it to Cromwell, Cromwell communicates with the Google Cloud platform. It runs it in parallel to the degree possible on the Google compute engine and then you get your results in the appropriate buckets.

(Refer Slide Time: 23:17)



So, each task is executed on its own virtual machine. So, if you have five tasks in your workflow, each task will get its own computer essentially and if possible it will run in

parallel but if two tasks are related, one task will get its own computer; once that is finished that computer will go away and then you get a new computer for the next task and you take the output of that from the bucket where it was written and then use it for this new computer that is just been created.

So, the whole the Cloud platform works on this concept of a virtual machine which is a CPU for doing the computing, hard disk for local storage.

So, and the methods that you get from the Docker; so, all these go into one virtual machine and then there are buckets that are separate from the machine. So, the buckets are not part of your virtual machine to start with, the virtual machine only has like storage but it does not have buckets; buckets are separate long term storage.

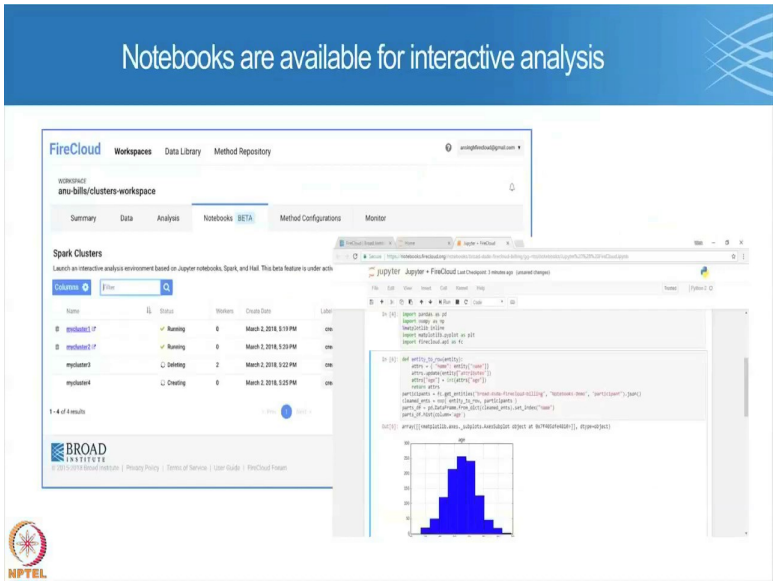
So, when this virtual machine goes away after the task is done, all the data that was in the hard disk also goes away; you do not have access to it anymore, it is wiped out. So, how do you use the virtual machine is you start with the bucket, you copy the data you need to the virtual machine.

So, that is called localization and then you do all your computing in the virtual machine. Once the computing is done you delocalize the data, you take results that was written on the hard disk and put it back into the bucket and then the task is done this virtual machine goes away and any data or any settings, any results that was on this virtual machine is no longer accessible.

So, you have to set up your things in such a way that thinks that you want as output or properly marked as output. So, that at the end of the task it will copy all those to the bucket during the delocalization process. If you would have a temporary files that you do not care about, you just do not say anything and they will go away.

So, I think this is a combination of everything and a repeat of a previous slide I think. So, it is just to recap that that how the whole system works.

(Refer Slide Time: 25:26)



So, now off late I think FireCloud also has notebooks available; if people are familiar with Jupyter Notebook in Python. What you do is you have a notebook that basically knows the programming language and you kind of type commands and as you type commands the results will appear at the bottom and then if you want to give it to your friend to look at what you did, they can see the code that you type, the results you get, you can do plots, you can also add explanations.

So, for example, the R program we wrote yesterday, if you had done it as a notebook, if there would be an opening section saying this is code to do x y z and then there will be the actual code and then the results would be below it, if you want to plot it you can plot it and if you want to run it on a new data set, all you have to do is change the data set and then click enter and the whole notebook will run and so, all the results will be regenerated with a new data set.

So, that is kind of the new recent concept to make computing easy and to kind of be able to share your computing with others or when you publish your paper, if you make a notebook available that has your entire methodology encapsulated.

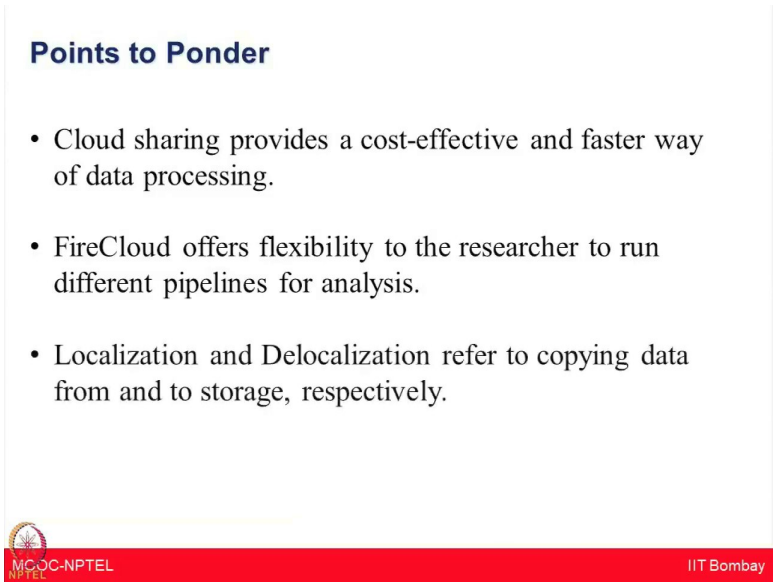
Then you do not need to field queries about how did you normalize the data or what did you do to filter, everything is there and the results are right there. So, following up on that; so, that that concept primarily started with Python programming and now has kind of percolated into pretty much any kind of programming you can think of and so,

FireCloud is also to starting suppose support that; I have not used it much, but it is available.

(Refer Slide Time: 27:00)

And, that is where you would find the actual FireCloud environments.

(Refer Slide Time: 27:06)



In today's lecture you were introduced to the concepts like Cromwell which is an execution engine that executes workflows across multiple hardware. Several paths can be used to run pipelines on FireCloud. The results once obtained are stored in buckets. These can be retrieved and accessed by the user by simply logging to the cloud. A data

model specifies all the information relevant to the patient, sample as well as when needed for creating a configured method. In the next lecture you will be given a demo of FireCloud and how it can be used to set up data analysis workflows for large datasets.

Thank you.