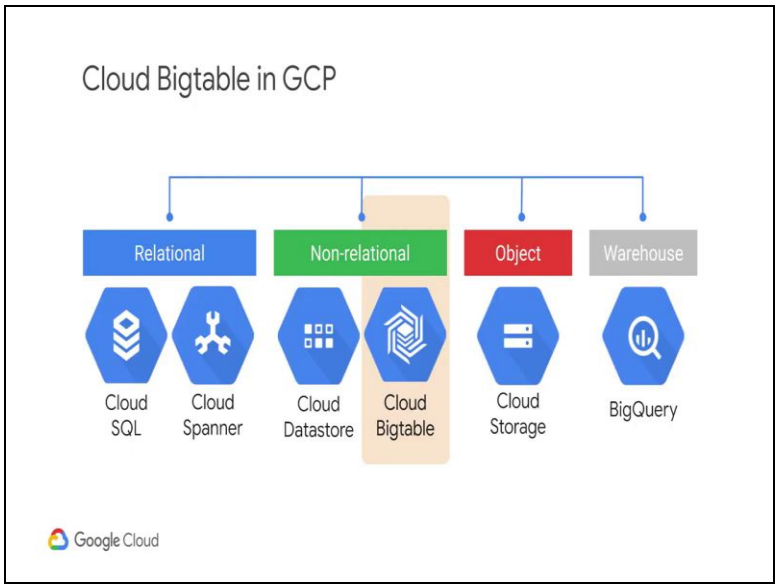
Google Cloud Computing Foundation Course
Priyanka Vergardia
Google Cloud

Lecture-34
Cloud Bigtable as a NoSQL Option

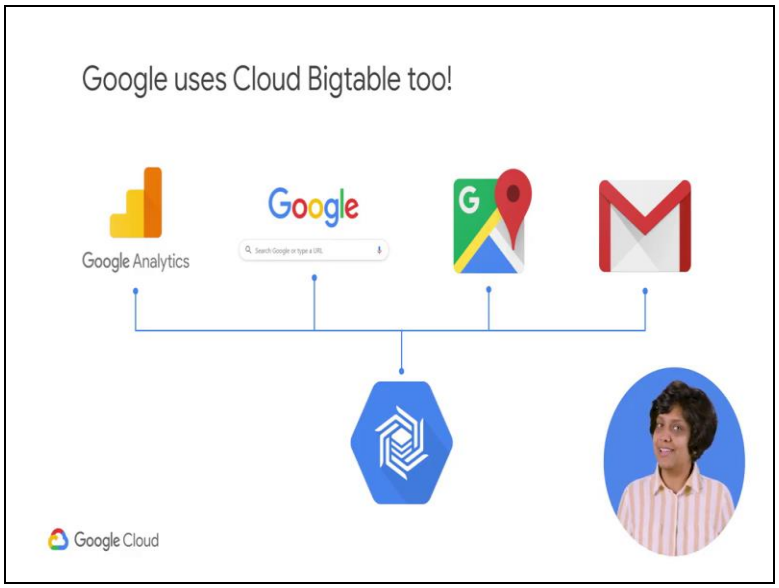
(Refer Slide Time: 00:07)



Final topic of this module describes how to leverage cloud BigTable as no sequel option. Cloud BigTable allows with non relational database requirements and it is high performance and no sequel database service for large and pinnacle and throughput intensive operational workloads. It is designed for very large mine of data it is a great for internet of things user analytics, financial data analysis, time series data and graph data.

Cloud BigTable is also an option if support is it not required for acid transactions or if the data is it not in highly structured.

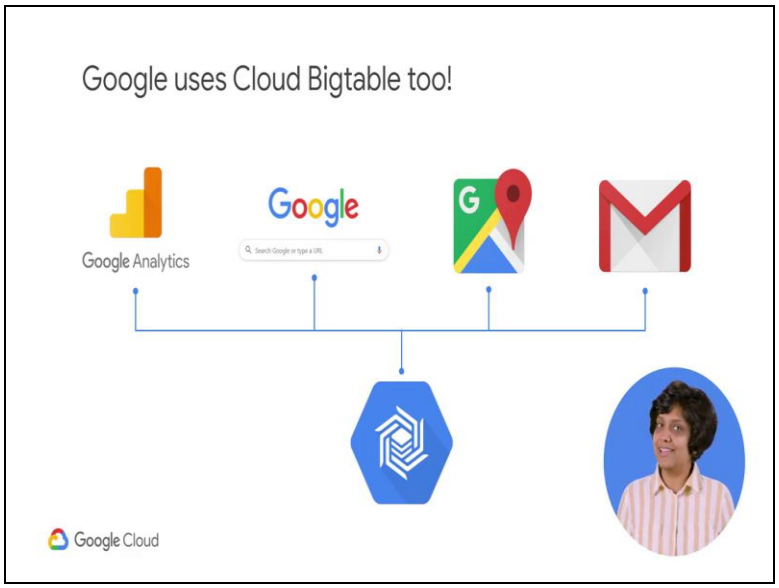
(Refer Slide Time: 00:40)



Cloud BigTable is the same database that powers many of our core services including analytics, Search, Maps and Gmail. BigTable was an internal Google database system. It was so revolutionary that it is kick started the no sequel industry. WE wanted to build a database that could deliver real time access to terabyte of data. The result was BigTable and in 2006 we released research paper describing it.

This was later awarded as been one of the most influential papers of the previous decade. This gave people outside the Google ideas that led to the creation of popular no sequel databases. In 2015 cloud BigTable became available as a service. So you can use it for your own application.

(Refer Slide Time: 01:29)

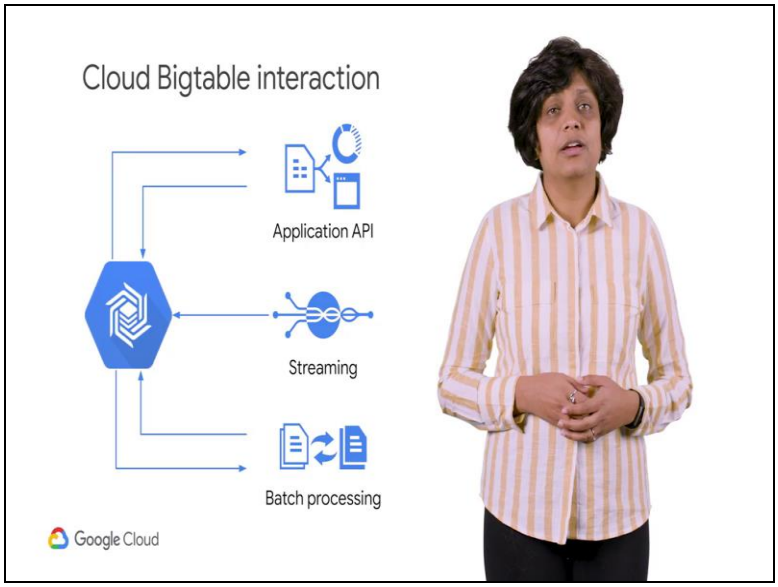


Cloud BigTable offers high performance under high loads for that reason large apps and workloads are faster and more reliable and more efficient running on cloud BigTable. Cloud BigTable is ideal for storing large amount of data with very low latency. Databases can automatically and seamlessly scale to billions rows of thousands of columns allowing you to store terabytes of data.

Changes to the deployment configuration are immediate. So there is no downtime during reconfiguration. Replication as high availability for live serving apps and working those isolation for serving versus analytics. Because BigTable is a fully managed service there is no need to worry about configuring and tuning your database for performance and scalability. And they also create data backups to protect against catastrophic events and allow for disaster recovery. You can use cloud BigTable for range of applications from real time on analytics and all the way to tracking millions of readings from thousands of internet of things sensors.

Because cloud BigTable is compatible with industry standard like ports like HBASE and Hadoop big query and cloud dataflow it is easy to put all the data that works for your apps. In terms of security all data in cloud bigtable is encrypted both in flight and at rest. While access to cloud bigtable data is easily controlled through IAM commissions.

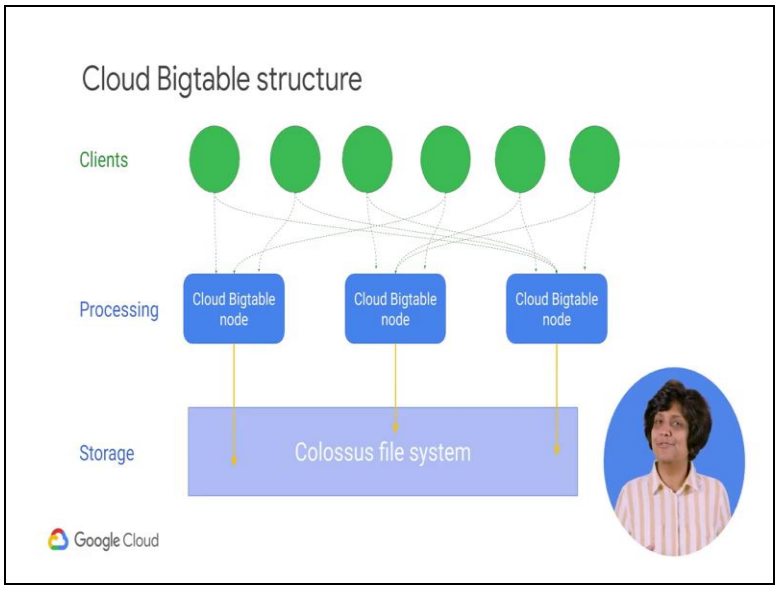
(Refer Slide Time: 03:03)



As cloud bigtable is part of GEC process system it can interact with other GCPs services and third party clients. From and application API prospective data can be read from and written to the cloud bigtable through data service layer like manage VM's, ASP Page server or a java server using AG based clients. Typically this will be serve data to application dashboards and data services. Data can also be streamed in through a variety of popular stream processing framework like cloud data flow streaming, Spark streaming and strom.

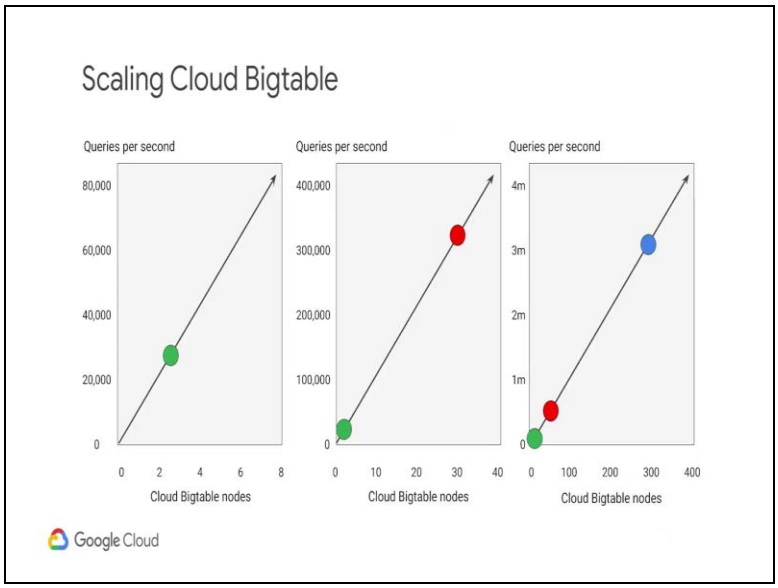
If streaming is not an option data can also be read from hand written to from cloud bigtable through batch processing like Hadoop map reduce, cloud dataflow or Spark. Often summarized on newly calculated data is written back to cloud bigtable or to a downstream database.

(Refer Slide Time: 04:00)



This diagram shows a simplified architecture of cloud bigtable. It illustrates the processing which is done through a front end, server pool and nodes it handles separately from the storage. A cloud bigtable table is sharded into blocks of continuous rows called tablets to help balance the workloads of queries. Tablets are similar to age pace regions, tablets are stored in colossus, which is Google's file system. In a sharded a string table or SS table format. And SS table provides a persistent ordered immutable map from the keys to values.

(Refer Slide Time: 04:43)



Where both keys and values are arbitrarily by straights, this chart shows that as the required query per second increases the nodes required will increase too. The throughput scales linearly so for every single node that you harm you are going to see a linear scale of throughput performance up to 100’s of nodes.