Company Participants

- Carl Olofson, Research Vice President

Other Participants

- Anurag Rana, Senior Technology Analyst, Bloomberg Intelligence

Presentation

Anurag Rana {BIO 7440273 <GO>}

Good afternoon, everyone. Welcome to the Bloomberg Intelligence, the State of the Database Market and the Cloud Webinar today with arguably the smartest existing databases that I have come across (inaudible).

Just to start off with some housekeeping items, today's presentation will be recorded and available for playback. You can access the replay via the link sent to you in the email from Bloomberg webinars. At the bottom of the slide window, you can adjust the volume and maximize your screen. And then you can ask a question by submitting one to the right of the slides. We will address questions at the conclusion of the presentation.

Also just a quick word about Bloomberg Intelligence, we provide in-depth research and data on industries, companies, credit litigation, etcetera. We have about 300-plus people that are in our department. We cover close to 2,000 companies. So with -- and you can access all of our research on the terminal be BI .

The agenda for today is to talk a little bit about the overall state of the database market. How much is it going to grow? What's Oracle's cloud strategy? Comparing Amazon to Oracle and Microsoft. The alternative database technologies from NoSQL to Data Lakes and Carl will just present his summary and conclusions.

With that, let me pass the presentation on to Carl that can go over the discussion.

Carl Olofson {BIO 21744954 <GO>}

Thank you very much, Anurag (ph) and before I get started in our topic for today, let me just give myself a little bit of an introduction for you, so that you know who I am. My name is Carl Olofson. I am Research Vice President at IDC, covering Data Management Software. I've been in the IT industry for over 40 years. Spent part of that time as a programmer-consultant, much of that time as a software development, building database management software and as a Product Manager and as a Marketing Product Manager and for the past 21 years, I've been working at IDC, analyzing and researching the market.

So over that time course, we've seen a lot of change and you'll be seeing some of those changes reflected in my remarks today. Why don't we begin by talking about the overall state of the market?

So in general terms, we're just going to -- just some high level revenue observation of the overall market growth for relational databases, which include products like Oracle database and Microsoft sequel server IBM Db2 and so forth, grew from 2016 to 2017 by 6.9%.

We saw most of that growth happening in the cloud and so, no surprise the strongest growth was shown by Amazon Web Services with their products RDS Aurora and Redshift and Microsoft with their product as your sequel database, whereas there was tepid to declining growth for the -- sort of the big mainstays in the industry Oracle, IBM, SAP and Teradata.

Public cloud services share of the market is approximately 11% at present and the drivers include overall data growth. Just to say the amount -- the volume of data out in the world and how rapidly it's growing; Edge and IoT Data Analytics, management of analysis of Edge very important and data science.

In the non-relational space, this includes legacy products that run on the mainframe and some many computers that are hierarchical or navigational in nature as well as end user databases and multi-value databases. That market grew by 2.2% from 2016 to 2017, in that, it's a very stable market. Most customer relationships are decades old. Their applications are built-in with the database very dependent on the database and those people are very reluctant to change.

The most interesting vendor in the space is InterSystems, which has modernized their database into something that has some completely different characteristics and that product is called Cache. It's mostly to be found in the healthcare sector in the United States, but it's in a variety of sectors in other countries, especially in Western Europe and in Australia and New Zealand.

The dynamic data management software market is the market that includes the things we typically refer to as Data Lake or Hadoop Software as well as no sequel database. It's called dynamic because unlike the other two products in this market do not require scheme (ph), in other words, they can ingest data that is not defined to the system.

What that also tells you is that most of the knowledge about the definitions and rules of the data exists in program codes rather than in any form of definitions within the database management system.

The overall market growth for this space from 2016 to 2017 was 62% with the dominant vendors being Microsoft with HDInsight and Cosmos DB and Amazon with EMR and DynamoDB.

Most of the new development is on-prem at present, but cloud revenue drives growth and now why is that? Why would most -- you would think if you were doing most of the development and implementation on-prem, why would cloud be driving the growth and that's because most of these products are open source. So people are developing and gradually implementing these things in production using open -- the open source model.

The open source model is a slow growth model because it's subscription-based and it's based on people using the product first, getting to a certain point of maturity and the use of the product and then acquiring a subscription. So the growth rate is slower.

Whereas in the cloud, you have to buy a subscription as soon as you go in because you have to pay for the use of the technology in the cloud and that's why cloud revenue is growing faster than on-prem revenue, even though the user base is larger on prem than it is in the cloud. And that will eventually all sort of normalize out.

Drivers in this space include new operational data models, such things as managing data in smart apps that you have in handheld devices and so forth. Handling and routing data that maybe coming in from sensors, also general online state management applications and gaming applications and very specialized applications that are typically Internet based and require a lot of state data to operate as well as specialized data analytics that require functionality that's not native to a relational system.

So if we look at the competitive situation amongst the vendors in the three markets that I just described and the relational area, Oracle continues to dominate with 42% of the market, Microsoft has 24%, IBM has 13%. What is interesting here is that Amazon Web Services has 3%, which looks really small and it's really small, but it is the fast -- it is one of the faster growing and is that -- so that piece of the pie is getting bigger, whereas IBM at 13%, that piece of the pie is getting smaller. The SAP is fairly stable.

We really expect it had a lot of growth when they were driving so many of their customers to Hanna. It sort of tapered off a bit in the past year, but we expect that growth to continue, not as strong as before, but fairly strong nonetheless. A lot of the damper is not because people are not moving to Hanna, but because of people moving off of the legacy SAP products SAP ASE and SAP IQ.

In non-relational space, which is actually on the right hand side of your screen, the dominant vendor is Microsoft with access, followed by IBM with IMS -- which is a mainframe hierarchical database and then you see InterSystems at 14%. That company has been growing steadily. In other words 10 years ago, it was sort of next to other and now it's got 14%. It's actually become the third largest vendor in that space. So important one to continue to watch.

In the dynamic data management software market, which is in the middle, we see Microsoft has the largest share by a little bit with -- as I've mentioned before, Dynamo, or sorry, Cosmos DB and HDInsight and Amazon following with EMR and, DynamoDB, that's going to continue to be a battle or a while.

Google with -- has Bigtable, but Google is going to become an important player in the future and Cloudera and Hortonworks interesting that they recently announced their merger. We are not sure what this is going to mean for the company. So making long-range projections would be a little bit irresponsible. However, I think they're going to continue to be important in the space.

Okay. So foretasted database market growth we expect the relational database market to grow by 7.7%. Compound annual growth rate to 2022 and the drivers include the continued need for analytic -- the need for analytic transaction processing, that means blending in analytic queries to transaction processing.

Most transaction processing today involves just a few simple background queries such as you know, if you're processing an order, you look up the customer and that sort of thing.

These are analytic queries, that are doing things like looking for patterns and data, not previously possible because of the overhead and demand that's placed on the database, but with these new models that involve (inaudible) and memory-optimized data management. They're capable of doing transactions and some fairly sophisticated, not full-on multi-table joint type query operations, but fairly sophisticated analytic operations against the data and so analytic transaction processing, which you can think of as a merger of OLTP and OLAP, is going to become increasingly important.

Also higher quarter analytics and machine learning, one of the things that we're finding is that there was a lot of experimentation with using Hadoop and Spark for deep analytics machine learning and they are still going to be use -- they're still being used and will continue to be used for those things, but in many cases, people have found, they need the discipline of the relational database and they're moving that functionality back into the relational sphere.

The cloud will be a significant factor in relational database in the last couple of years of the forecast. In other words, it's a five year forecast. A lot of the companies are preparing to move to the cloud, haven't moved much -- that much to the cloud. Some smaller companies have moved production data to the cloud and a few leading-edge customers have moved data to the cloud -- their production data to the cloud, but most have not. So that's a thing to watch out for.

Non-relational database is expected to decline by 2.2% compound annual growth rate by 2022. That's a five-year compound annual growth rate by the way. The customers -- the users are looking to move to the cloud -- users who are looking to move to the cloud who are using mainframe, hierarchical and navigation databases are considering rewriting their applications.

For years, they wouldn't consider this, this is what kept them on those systems, but now they're seeing that the total benefit of moving to the cloud supersedes the cost and risk involved in rewriting those applications. So they're starting to look at that.

Multi-value databases will probably be flat, but not decline, which is important to know. So these are like the PICK databases and so forth and end user legacy mainframe will decline. A lot of the end user database functionality is now made available by other means, various kinds of in memory, analytics sandboxes and things that people used to use, things like Microsoft Access for and so forth. So we think that those are going to, in general decline.

The dynamic data management software market system is expected to grow to -- by 30% CAGR over 2022, again a five-year CAGR that's following a fairly normal growth curve for this market. I'm sure there are others who have more aggressive forecast from this, but because we're dealing with a space that's mostly open source, the revenue forecast estimates need to be necessarily conservative.

The rapid growth of open source usage has resulted in lots of -- lots of users, but not lots of revenue, but we think that most of those accounts will subscribe over the five-year period and that's going to be a major growth driver in the space.

The document databases are also becoming more business-friendly. They're doing things like they're including support for asset transactions, which never existed before in document database, well except from MarkLogic, but not in the other document databases and they're also adding enterprise-level security features. Obviously data security is a major concern for everyone these days and if you're going to be putting data, especially personal identifying information in document databases, you need to make sure that they're secure and can't be hacked.

In the Data Lake space, they're settling into a supporting role. Years ago people thought that Hadoop is going to take over everything. Didn't quite work out that way. Hadoop of course is designed as a data collection space, driven by MapReduce. However, MapReduce today appears to be adopting a very limited role of initial ingest processings, sorting, filtering and formatting or organizing the data and then the rest of the work, all the analytic work it seems is being handed over to Spark.

HDFS survives as a file -- standardize file format for this environment and will continue, but it's unclear as to how much of the rest of the Hadoop cluster management, file management technology will actually come into play, especially as people move their Data Lakes to the cloud and look to see the data on cloud native object storage and run Spark directly against that, which seems to be a raising model for use.

This also means that the Cloudera Hortonworks combination will need to basically rise above the general den of confusion with respect to Data Lakes by offering very comprehensive enterprise-friendly packages of capability that make it straightforward to deploy and manage.

Okay. So for forecasting these are the five-year forecast for the three markets. I know the print is small. I think you have access to the slides after this. So you can look at that at your leisure. As you can see, the growth rates I mentioned are reflected here. The interesting thing to notice about these forecast is that the top part of each column is the cloud forecast and the bottom part is the on-prem.

So as you can see, for relational database, the on-prem is actually about flat, it's growing, it's growing by a percentage point or 2 less than 2% I think. Whereas in the dynamic data management software space, the cloud portion is the part -- portion is doing most of the growing.

There is gradual growth on-prem and that's because people who have deployed various document database like MongoDB and Couchbase and so forth are purchasing licenses to protect their investment, their effort investment in putting corporate information or at least business important data into those systems.

But as they grow, we expect that most of them will be deploying in the cloud and a lot of new development will be going on the cloud for the very simple reason that the cloud is non-demand environment and we are doing development.

This way you can just sort of say, I want to have a server of a certain size, of a certain amount of storage to do my development work on and immediately it's there. Whereas if you're operating in -- on a non-prem context, you have to make a requisition for the server, for the storage, you have to reserve the time for those systems, etcetera, etcetera and that slows down the development process and we are living in agile times when slow is no go.

The non-relational database picture is on the right. As you can see it's declining. The only part that's showing some growth is cloud. There'll be a few elements, probably mostly in the end user space, things like file maker and so forth, that will -- that will be growing in the cloud the way that picture looks.

Okay. So as far as Oracle's cloud strategy is concerned, a lot of questions have come up about this. It's important to understand that the cloud is a critical success factor for Oracle. They have as an absolute priority to move their as customers as many as possible to the Oracle cloud, not just to the cloud, but to the Oracle cloud.

It's important to maintain account control, which has always been the key to Oracle success and it's largely for this reason Oracle came out with the autonomous database service versus the Oracle autonomous data warehouse service and now they've been talking a lot about our transactional service, which will be -- which we'll learn more about at Oracle OpenWorld.

These things are necessary in order to do two things. One is to demonstrate the value of the cloud as a place where you no longer have to assume responsibility for the physical management of the database, but also because you don't have to apply your own patches. You don't have to do any of those things, which means that the database is going to be available all the time. It's going to be running at top performance levels and your DBA and some technical people can work with developers to create applications and adjust systems to match enterprise needs rather than constantly tuning with database.

So that's the value proposition of Oracle, the kind of database in a nutshell. And to be honest it is the most clear demonstration of why not just the cloud, but the Oracle cloud is important, because of course, the autonomous database is being managed by Oracle personnel, who are presumably the most knowledgeable and expert in managing that environment and they can also enjoy economies of scale because they are managing multiple customer instances at the same time and therefore their efforts are -- their efforts are spread across those and so it works out well for them and for the customer.

This is really important to Oracle because it means that if they're successful with this program, they can retain the bulk of their database customer base, which is as I said, it's key to them. If customers move their Oracle databases to other clouds, such as AWS or Azure or Google or something else, that would be bad for Oracle because now they don't have that level of account control any more.

They have to split the customer's attention and interaction between themselves and the cloud provider and that could ultimately mean not only that the customer spends less with Oracle and is less influenced by Oracle. But that they even could contemplate moving to another database systems such as Aurora or Spanner or something else and that would be a disaster for Oracle. So keeping Oracle customers on the Oracle cloud very important.

Ideally those same customers will also if they are in the Oracle cloud, use Oracle infrastructure as a service for the non-Oracle workloads. In other words, they'll choose to keep all the data and operations together and that's obviously -- that's what they're pushing forward as well and you hear this, and if you read their quarterly reports and the management remarks, you'll see the infrastructure as a service coming up all the time for that reason.

They also want to attract new customers. Now this is important because Oracle has not historically been really good at attracting new customers and the reason is pretty simple. Oracle has a large and expensive direct sales force and that direct sales force in order to get -- to justify the cost of sale that's embedded in using such a sales force, that sales force is concentrated on the largest users, which means that smaller users really don't get the attention.

And therefore Oracle has had a hard time attracting them. Oracle has tended to respond to this problem by working with third parties with ISPs and with consultants who can bring in smaller customers, but that sometimes gets in the way of their own product strategy and historically, they haven't done a real good job of managing all that together.

So moving to the cloud alleviates the essential problem because the Oracle cloud provides a direct means of ordering up services and so people can sign up online and adopt the pay as you grow pricing model, which means that when they're just using a little data, they pay a little and few processors and when they use more, they pay more and that is attractive on several levels.

One is that the Oracle customers don't have to deal with the sales person, which a lot of people are glad for. They know exactly what they're getting and they are in control of how the account progresses.

In fact, it can be said that even for larger customers, this model reduces the amount of interaction they need to have with the salesperson, once they move to the Cloud because they can see exactly where they stand. There will be no more audits, which everybody hates, but instead they will see where they stand online and they can make adjustments accordingly and it's -- everything is transparent and that should improve customer relations because customer relations between Oracle, typically what you find is the technologists really like Oracle. The DbAs really like Oracle. The program really like Oracle. The business people don't, because of the friction that's created by the Oracle audit and having to deal with the Oracle sales force, which is powerfully motivated to increased sales wherever possible. So this could help them with customer relations.

There was a question about BYOL or Bring Your Own License, largely motivated by customer demand that wasn't originally part of the Oracle cloud plan, but Oracle found that a lot of customers wanted to apply the value of their existing license to the cloud and so in order to get them to move to the cloud more swiftly, they adopted this policy, removes that friction so that accounts can apply their licenses in the cloud and get credit for them.

And there is another reason why customers want to use their licenses or at least use perpetual used licenses in the traditional form and that is that they're license management people or typically lawyers or contract management people. There are uncomfortable subscriptions who are then dealt with them before.

They understand the perpetual use license contract and support contracts very well because they've been working with them for years. And they're not sure of all the implications of switching to a subscription.

We think that over time, they will switch to a subscription model because it gives them more flexibility. Obviously, Oracle likes it because it gives them more of a guaranteed revenue stream, but for the time being there is some reluctance and so the BYOL policy in general, the policy of reporting perpetual used licenses in the

Cloud are necessary to overcome the objections I mentioned and get more people on the cloud sooner.

So in comparing Amazon, Oracle and Microsoft in the relational database sphere, they really represent very different -- their backgrounds are very different. Amazon has been the -- has been sort of a darling of new application developers. These application developers who come to AWS to build applications that they make available to users in the cloud.

They really like everything about Amazon and from relation, when they need a -- transactional relational database, they naturally turn to Amazon for that too. Now RDS -- which is a relational database service, has a number of options in it, including a number of databases they don't control such as Oracle and Microsoft and my SQL and PostgreSQL.

However, Aurora is the one that's not only managed by Amazon, but is optimized by Amazon for the AWS environment and so customers like that idea, it has had a MySQL service layers so that you can take a code that's from MySQL and run around it and it now also has a PostgreSQL service layer, so you can go with either of those depending on how you've chosen to write your application.

That's gaining a lot of popularity within the AWS community and Redshift is their analytic database. It's their sort of data warehouse platform if you will and it's just those there -- it's a fairly frictionless adoption process for people who are committed to the AWS environment.

With Microsoft, the situation is different because Microsoft has a lot of legacy obviously that makes stuff spin around a long time. They have a lot of existing customers. Their main challenge has been to move Microsoft SQL server users to Azure AQL database because if they don't do that, it's a lot harder to move their customers to Amazon -- sorry to Azure, because to get the most out of Azure, you really need to have a platform that is optimized for Azure like any cloud environment it's highly dynamic, the resources are virtualized, the system needs to be able to take advantage of that.

And SQL server of course is designed to run on a defined server, in a conventional environment with storage and all that. And so it's not going to give you the dynamism, the resource dynamism that you would need in the cloud. So -- but they had a problem, which was that the two -- they represented a code for a critical upper level, which meant that applications written for SQL server would not automatically run as your SQL database. They would have to be changed.

So Microsoft is taking steps to change that so that the same applications can run on either platform. It seems like that should help in moving Microsoft-based applications on to seek Azure SQL database and therefore into Azure, which is what they hope to do. And of course they have a lot of client, I'm sorry, they have a lot of third-party vendors, providing software. Most of those existing Windows-based vendors have been reluctant to move to Azure until they feel that it is a frictionless process.

So that hasn't happened a lot. There are lot of new vendors who have been developing on Azure and been quite happy with that. But -- so it's quite possible that we could see Azure, which has been growing fairly strongly anyway, gain even more momentum as a result of this shift.

Emphasis on Microsoft, we do everything for your model. A big difference between Microsoft and Amazon is that Microsoft is heavily prescriptive, meaning that they put together the elements that you need for various used cases and say here are the tools and elements that you need to do these things whereas Amazon is more liked in as a Chinese menu.

There are (inaudible) Amazon you can say, you can use Azure, or you can use Oracle, or you can use Microsoft, or MySQL any one of those, there are several MySQL variance also available on AWS. There are all these options that you have to choose from. It's very confusing and I've heard a lot of people complain that it's a little too hard for them to figure out how to configure their environment there and Microsoft is more cookbook oriented. So that's a difference there.

Oracle is they're coming from the perspective of large enterprises well established. You need to move those enterprises, who are looking to go off -- get off the datacenter and go the Cloud. We need to move them to the Oracle cloud. So Oracle's first job is customer retention in a big way and of course they have some of them most -- the largest and most demanding customers with the most demanding workloads and requirements and so that's been their entire focus.

Now people don't -- a lot of people may not realize it but Oracle actually has three relational database systems. They have Oracle database times 10, which is in memory database that was developed separately and then acquired and MySQL which was also acquired.

The role of times 10, the role of times 10 is as a pure in-memory database that's used for cases where in memory, the pure memory capability is critical where latencies are very, very low and a lot of data processing these we handle in memory and where the sizes aren't to huge. MySQL is basically its open source, it's a means of attracting new customers. They have a large -- there's a large MySQL community and even though many other people in that community are now using alternatives such as perCona and MariaDB, Oracle is still actively involved in them -- that MySQL community because it is an important prospect base for them.

So as I mentioned here, the one issue that Oracle has in moving customers to their cloud is that many customers are reluctant to increase their dependence and investment in Oracle because of past experience. Oracle's hoping that the changes they've made will address that.

So at alternative database technologies, there's been a lot of excitement over the past 10 years, regarding so called Big Data Systems and no sequel databases and the result -- there are a lot of -- there are a lot of breathless explanations of how these systems are going to make relational databases absolutely as we've seen that hasn't happened. However these new systems do address new and emerging workloads very well in fact better than relational databases. Relational databases continue to address the workloads that they have traditionally addressed well.

Relational database model was actually developed as a means of providing application neutral query data, prior to the relational concept, which came out in late 60's offered by Dr. Ted Codd. The only way to report on data was to write complicated programs that could navigate the database, find the database, look for and then produce a report and Ted Codd's aim was to provide a system that you could dynamically query.

And as a result its strength has been reporting and they still are because relational database is the only architecture designed so that if you do it right, if you fully normalize your data, you don't have to anticipate any of the questions are going to be asked.

It will have answer any mathematically valid question that you pose to it along any line of query. You don't have to know about the structures. You don't have to collect certain structures to do it. You just ask the question and it would and it will give you an answer. It's the only system works that way. All others are dependent on structures that you have to retrieve and sift through to find the details that you're looking for.

The reason we have transactional database is that relational databases that customers didn't want to use one architecture for transactions and another for query. So it was necessary to provide a database that does both and it works reasonably well for classic two dimensional data structures such as lists and ledgers and most of the things that we do is sort of back office accounting things and maintaining customer list and stuff like that.

It has a tough time with different shapes then two dimensional ones, such as precursor structures, graphs, embedded structures, graphical objects and so forth because these demand -- these demand or a style organization that either breaks the relational model from a perspective of how these databases work or if you try to impose relational model on them, they just don't perform, they just can't perform.

And another aspect of this is that relational databases do require fixed schema. In other words, the relational database is schematic database, which means the database needs to know the organization of the data before you put data into it.

Whereas the new class of Dynamic Data Management service that are out there including no sequel and they do operate on the basis of the idea that you put the data in first and then discover how it's organized. So that capability is supported by things like graph databases, in some cases people use key value stores, document databases and object-oriented databases, which are actually also schematic, but different, can also serve some of those jobs and is used, especially when their relationship is very close with a language such as Java.

So the alternative technologies include Data Lakes and what has been called no sequel databases, which originally have actually met no sequel and then they decided, well that's sounds too restrictive. So they call it not only sequel, but it really -- what it really means is that well we can support a sequel interface. So if you go through and you define data in just the right way and you mark it in just the right way and put in some metadata, you can execute sequel, but it's really not a relational database.

So, Data Lakes serve three basic functions. The first is initial ingestion, which includes all the filtering, sorting, regularization. So for the also do discovery as to what data needs to be secured and so forth and then after that, you may either export it to a relational data warehouse or you may write it out to a object file system for analysis by Spark or you may run Spark directly on it and now when you do that, that's largely for these sort of deep one-off analysis conducted by data scientists who are looking to discover patterns and anomalies in data.

So initial ingestion, deep analysis and then the third function is long-term data retention because it's cheap. So the related technology to Data Lakes includes Hadoop and all the sort of elements of the Apache Hadoop constellation of projects and also object storage, particularly native cloud storage such as AWS, S3, all operating or mediated through HDFS protocol.

Some people are starting to use Parquet files and access those, which can enable user manage data in common format and spark, which has become the sort of pay factor standard for most analytics in this environment.

And as an emerging role data, receiver data, organizer data router, which means it becomes a kind of operational data hub in which data comes in both from external sources maybe streaming sources in some cases and also from other structured, managed internal sources and it gets combined in different ways and moved out.

This is as I say an emerging role, but this I believe and I'm not speaking for them, but I believe this is a direction that Cloudera, Hortonworks combination will be looking to pursue.

In the no-sequel space we see for general benefits, that the lack of schema support means that developers can just go ahead and do their thing. They can build and make changes to their applications at will. The applications right, documents out to document database and the database will support it and so they can address constantly shifting requirements, they can fulfill the agile development model. And the operational constraints are not there, which gives you a greater -- required by asset are not there, which gives you greater scalability and flexibility in their cases where because the data that you're putting is unrelated to the other data that you're putting in the database, consistency isn't really an issue.

Now, when it is an issue when there are formal relationships, then you need to be a little more worried about those things. Document databases are key on the operational side for no SQL databases that their contents are defined by application code, but they are self-defining meaning that the field names actually occur in the documents.

The documents can become inconsistent, which makes analytics difficult, because remember there's nothing controlling this. You can put rules in some of these -- some of these systems while you put rules in to force documents to be more consistent, but it's not a primary requirement.

And there, most commonly used for online session management maintaining the data, session data as users use in online environment, which is important, not only for recoverability of user session, but also for later analysis and tracking.

Also for IOT tracking and which is IOT data tracking and for gaming applications, very common and popularly used for gaming applications. Other no SQL systems include Viacom stores such as Cassandra, which is also Agile. You can add new comps at will -- but it doesn't do great at complex query.

It's not that optimal for operations where performance is an issue, but it's pretty good for analytics, particularly aggregation analytics against very, very large collections of data. And the other key value stores, which are mainly used for the shared persistence of application internal data and also key value stores underlie a number of document databases that are out there as well.

So you have sort of like a two layer system. You got the document, management layer on top and then you got the key value store underneath. And actually key value stores are usually being used by some emerging relational database systems as well.

Now I am going to wrap up the formal portion here and so we can get onto the questions. Relational databases remain core for back-office data processing enterprise data warehouse, all the traditional boring things. Well, a lot of the developers that you're going to encounter at Strata or at Dreamforce or whatever we call boring, the relational databases will move to the cloud, but whose we don't know.

Oracle, Microsoft Amazon, Contend; there are specialized providers for data warehouse functionality like Snowflake out there. Interesting to see how that all works out and in fact Teradata has a cloud called Intellicloud as well recently developed. Data Lakes and NoSQL are primarily served new workloads and those for which a relational database is historically ill suited. Oracle needs broad acceptance to the Oracle cloud to thrive long term critical requirement/.

Microsoft jumped out in front with this -- with Azure and people -- when they first came out of Azure, people were scratching their head and said why do they even need to do this, but it turns out to have been a really smart thing for them to do.

Amazon Web Services is as solid because they are like it doesn't matter to us. You can buy, you can use our stuff, you can use other people's stuff. Will deploy it manage it and they have Amazon's retail mentality. We have our brand products on the shelf, but we have other brands on the shelf as well. You buy those, we still win, because we're still doing business with you.

So Amazon -- that's Amazon's approach. It's going to continue to be their approach, but it puts a lot of the burden of making decisions on the consumer.

MongoDB and Horton Cloudera plus Hortonworks are growing pretty well. But the open source model makes it difficult to track and predict. I mean the thing about Open Source is that in order to know where it's going, you really need to have a really strong sense of who's using it. but without revenue account, which is what we do at ITC.

The only other way to do is through surveys, which we've done from time to time, but we need to measure the user basis of these products in order to give ourselves a better view. but we think we've got a pretty good handle on it. And we think that MongoDB, CouchBase and others are growing pretty well in their user communities and eventually that will translate into revenue growth as well.

So those are -- that concludes the formal portion of my presentation. I'm going to hand it back over to Anurag and -- questions to answer.

Anurag Rana {BIO 7440273 <GO>}

Yeah, we have a lot of questions Carl. Let me start off with a couple of them. Then I will read them out from the ones that are coming from the listeners and I would again request if you have more questions, please fire them ahead.

So one thing is very clear as you've mentioned, for Oracle to be successful long term, Oracle Cloud has to succeed and we have obviously seen slowdown in that perhaps largely because of the pace of migrations of these larger databases or their legacy clients to cloud in general. What do you think Oracle can do to accelerate the shift to Oracle cloud?

Carl Olofson {BIO 21744954 <GO>}

So part of the problem, is this. They are largest customers and they really need their largest customers to come to the Oracle cloud. Also I have the most complex situations in their data centers and they just can't move -- they can't turn on a dime.

In other words, they need to -- not only do they have complex Oracle installations, they have a lot of other technology in their data centers. They need to make a lot of decisions. Once they've made those decisions, some of that technology can't be moved for various reasons.

So what they are doing now is they are implementing hybrid cloud strategies and we're going to continue to see that going on for a while. As they formulate their plans, the other thing is that they have a lot of technology in the data center that is on depreciation schedule. They're not going to stop. They're not going to turn off expensive servers and storage systems until they've hit their depreciation limit. So that means that timing becomes important as well.

So we really didn't expect a large number of those customers to have moved by now. I think what you've seen is that there is a sort of initial rush of people with the pent-up demand. We're past that now. And now we are into the more serious long term -- how do we get there kind of thing and it's just going to take several years. I don't know what Oracle could do to speed that up. One thing that they have -- there are two things they have done.

One is the Oracle Autonomous Database Technologies that they've come out with is attractive enough that once it's mature shown to be mature and right now we don't, -- we've got a lot of people, we've talk to are beta customers. They need another year for us to get the experience of the people who have these technologies and production particularly because the transactional one isn't even available yet. But those are things that can help accelerate the process of moving to the cloud because it makes it easier.

The other thing -- the other thing is this thing called Oracle Cloud Customer and this is a box that they have that works like a -- that works like a little piece of the Oracle cloud in your datacenter. So for people who say, well I can't move most of my systems yet. I'm not ready to do that, and they have to interact with these Oracle systems on a very low latency basis and so that we need to keep them connected to our internal network.

You say -- okay, no problem, put in Oracle cloud customers. They run the Oracle database, Oracle is managing the database for you and right now Oracle clouded customer doesn't actually support Oracle at times database, but we think it will in the not too distant future.

You put that in and that way you can sort of have your cake and eat it too because you can start the process of Cloud icing parts of your data center while actually keeping those elements in the data center physically connected through your internal network to the other systems. So that -- those are two things that Oracle is doing to try to accelerate that process.

Anurag Rana {BIO 7440273 <GO>}

Fair enough and we have seen now your stock works in a similar way and use a hybrid methodology. But let's talk about those legacy enterprises. So if I'm a very large customer that I'm an Oracle shop that I've been using Oracle databases for several years, and I'm a fairly big customer. I am one of your the most high paying customers for Oracle.

If and when I decide to move my database to the cloud, what is the probability that I would go with either Oracle or Amazon or Microsoft like, which way am I more skewed to given my legacy investments?

Carl Olofson {BIO 21744954 <GO>}

So I don't think there's a one size fits all answer that question because it depends on what you're using the database for and how many specialized Oracle features you're using. If it's an in-house developed application and you have lots of PLC sequel, you're probably going to stick with Oracle technology.

Now that doesn't mean you go the Oracle cloud. You could deploy Oracle and AWS rents and I think, believe you can even deploy it in Azure, but people who are going to go to Azure are people who are Microsoft-centric, I believe, and there Microsoft-centric people who have some Oracle systems and there are Oracle-centric people who have some Microsoft system. So if you're Microsoft-centric, you're going to be -- the appeal of Azure is going to be pretty strong, but if you have a lot of PLC sequel, you're not going to move to another data.

On the other hand, if you don't, if you mostly just use sort of generic sequel functions and third party applications that support multiple databases, then you might consider switching database systems as part of your cloud conversion process, which then means go anywhere and we see people considering all of these things.

I tend to think at the end of the day that most of the large Oracle customers are more conservative in that and will stick with Oracle just because it's what they've used. They're not sure what would happen if they try to move their databases to another system. The mid-sized ones are the ones that might go to say an enterprise DB based post solution or something like that or they may even switch altogether.

Every vendor as you probably know has their Oracle conversion tool that is supposed to help you move Oracle databases to their platform. So, but for those with a heavy investment in Oracle, I really don't see them moving off of Oracle and the appeal of the Oracle cloud, especially if you're also using Oracle applications pretty great.

So the questions, I would ask are how heavily customized is your Oracle environment to use lots of PL sequel? Two, are you using lots of Oracle applications? Are you dependent ib Oracle applications? If either of those is the case, you're likely to go with Oracle and to go the Oracle cloud. If neither of those is the case, then there are -- there is a variety of options you might consider.

Anurag Rana {BIO 7440273 <GO>}

That begs the second question and in the case of the second example that you gave, how easy is it to migrate a large database off of Oracle to let's say Amazon or somebody else?

Carl Olofson {BIO 21744954 <GO>}

Well the first thing you would need to do is look at the characteristics of the database or maybe it's a data warehouse. I don't know and ask yourself can the proposed target system handle it? And the answer to that may eliminate a lot of what you would consider potential options.

If I'm looking at a large complex database data warehouse for instance, I'm probably more likely if I want to move off of Oracle and probably more likely to consider another database system that's designed to handle large complex data whereas such as Teradata rather than something like RedChip couldn't handle it.

But the feedback that we've gotten is that data warehouse users tend to be more comfortable with those larger more highly scalable environments. Now, if that's not the case, if the databases what we might say fairly prosaic and not too large, then there are a number of possible choices and I've talked to customers who have migrated such databases to other platforms and they seem quite satisfied.

So it's all a question of what's there, but if I'm moving a large complex Oracle database, transactional database, I would do a lot of due diligence in examining the potential targets as a moving -- something as moving to the Azure sequel database for instance.

You'd have to do a proof of concept project to determine whether or not it can handle the load, but if you decide to go that route, as I said before, there are -- everybody has their Oracle conversion tool and there are also third party Oracle conversion tools out there.

So that there are ways and means of doing it, but I would also assume that it is not a risk free and effort free exercise. There is. There are going to be catches. They're going to be a few things. We didn't anticipate this. So that's all part of the if you will, the friction that presents itself in making such a conversion.

Anurag Rana {BIO 7440273 <GO>}

Fair enough. One last question and we'll call it off, given that we've almost close to 1 o'clock. One of the clients -- one of the questions is about is Google Cloud platform even in the running or it's a completely irrelevant in your view?

Carl Olofson {BIO 21744954 <GO>}

Oh, no. I don't think it's a relevant at all. Google is of course a late-comer in terms of presenting a cloud platform that is considered suitable for enterprise data management but -- and I think that spanner is too new a product to build a formulated opinion about it. The white paper looks good.

But we have to see real customer cases before we -- before we know how to interpret it, but they're going to do everything they can. I think that most of the deployments will be new applications initially at least on the Google cloud platform, but a number of vendors have very good relationships with Google and have committed to GCP support.

And also the other thing to bear in mind about GCP is that not everybody is all that comfortable with the idea of totally committing to say AWS or Azure. They like to have a third choice out there and maybe some people will distribute some of their resources across both platforms simply in order to give themselves a little more flexibility, so that they feel like they're not completely dependent on one vendor and they're putting all their eggs in one basket.

Anurag Rana {BIO 7440273 <GO>}

Fair enough. Carl, as always is a pleasure talking to you. There are a bunch of questions and I will try to take them offline and get back to them. So with that, thank you so much and look forward to talking to you again.

Carl Olofson {BIO 21744954 <GO>}

You're very welcome and I appreciate everyone's attention and time in attending this event. Thanks.

This transcript may not be 100 percent accurate and may contain misspellings and other inaccuracies. This transcript is provided "as is", without express or implied warranties of any kind. Bloomberg retains all rights to this transcript and provides it solely for your personal, non-commercial use. Bloomberg, its suppliers and third-party agents shall have no liability for errors in this transcript or for lost profits, losses, or direct, indirect, incidental, consequential, special or punitive damages in connection with the furnishing, performance or use of such transcript. Neither the information nor any opinion expressed in this transcript constitutes a solicitation of the purchase or sale of securities or commodities. Any opinion expressed in the transcript does not necessarily reflect the views of Bloomberg LP.