WHITEPAPER

PERSISTENT

Analytics in the cloud: Mapping requirements to technology

Fernando Velez
Chief Data Technologist
Corporate CTO Office
Persistent Systems

Sunil Agrawal
Chief Architect
Corporate CTO Office
Persistent Systems

Outline

1. Introduction

After having been enclosed during its lifetime behind the firewalls of an organization, the center of gravity of the BI/analytics market is finally moving to the cloud. Recent market research surveys reveal that cloud analytics has now reached parity with on-premise analytics in terms of adoption for new projects, with most organizations hoping to expand their cloud implementations going forward.

An analysis jointly conducted between the Analytics practice and Corporate CTO identified three drivers that explain why organizations are increasingly turning to the cloud for their analytics needs. First, *more and more data* is in the cloud. Not only has data become available for analysis via public web services and public data sets, but also a significant subset of an organization's operational applications is now cloud-resident. Second, on the *demand side*, the business drivers for analytics are becoming more complex. With the help of analytics, organizations are seeking overall efficiencies and new opportunities by providing visibility into key business processes, which explains the appetite for predictive insights driven by data from multiple sources. Established efficiencies provided by cloud computing, e.g., operational cost reduction, also apply to cloud analytics. Third, from the *supply side*, cloud platforms have reached a level of maturity where organizations can develop flexible solutions using a large variety of data storage and management options; they provide the ability to scale computation up or down elastically as organizations need to, and they effectively lower the total cost of ownership.

The newer capabilities available in the cloud are raising the maturity level of organizations and their approach to analytics. At the same time, they need to evolve the processes and skills to support it. Most clients who engage with Persistent do so because we are regarded as innovators who can help with decreasing the time to develop and deploy their solutions. Our analysis advocates that we also should help them with a key part of the process, namely, identifying the technical and business factors relevant for cloud analytics, and helping them with the choice of provider and technology. In this respect, our experience in cloud analytics is key, so available technology and best practices needs to be shared as widely as possible. This is the goal of this document, which is written for an audience of architects, developers and technical presales people that help customers with developing new analytics solutions or moving legacy analytics solutions and operations to the cloud.

This document is structured as follows. Section 2 starts by summarizing the concerns and challenges that explain why the industry has been slow in moving to the cloud. Section 3 lists the technical and business factors that must be considered when developing or moving an analytics solution to the cloud. We then show how to materialize customer requirements from these factors, and we group them in categories: for instance, requirements on the *data* used to derive analytics insights, on the type of *queries* of this data; there are other such categories. Section 4 provides technical input intended to help with mapping these customer requirements, to cloud provider technology and services. This mapping may be used to ultimately determine the choice of provider and the overall environment with the provider. This section is written as a series of decision points that the customer must go through, each decision focusing on a specific area; it points out the choices to be made at each step, based on these requirements. Section 5 illustrates how we can apply these decision steps to a real customer case.

The reader should keep in mind that these steps, albeit generic, are still confined to the cloud analytics domain: this is not about moving or developing arbitrary applications in the cloud. We assume a generic architecture with the following mandatory components:

1. One or several "cloud databases" (a data warehouse, a Hadoop data lake, a NoSQL database, etc.),

2. Managed deployment on cloud infrastructure (at a minimum, hardware, OS, network and storage)

3. One or more processing analytic applications running on top of the cloud database: the computing power to run analytics is in the cloud.

Finally, section 6 provides a product to product comparison of the 4 main cloud platform provider competitors: Amazon Web Services, Google Cloud Platform, IBM Bluemix and Microsoft Azure. An overview of the services and technologies provided by these providers is given in Appendix 3.

2. Why analytics industry is slow in moving to the cloud

Deploying applications in the cloud have become normal in the enterprises, especially for CRM and customer facing applications. However, analytics applications, which require data from various data sources, have not been deployed in the cloud frequently. We believe this obeys to several reasons:

1. The first and foremost concern is **security** and **privacy.** Enterprises are worried about their customer's data privacy and related regulated laws around it. This has been identified as the top obstacle to implementing analytics in the cloud, according to market survey [1]. Before moving to cloud they would like cloud service providers to provide assurance on data security and privacy.
2. While there were guidelines and **compliance regulations** governing data, issued by standards bodies and governments, it was not possible to have data reside outside your premise. However, regulatory agencies and standards bodies have recognized the value and popularity of cloud services. New guidelines and compliance updates are spelling out safe use of the cloud. Adoption of cloud analytics will certainly increase as these guidelines get implemented at cloud platform providers, and as these providers offer co-located data centers in the country.
3. Due to queries running over large volume of data, **performance** is critical in some business analytics scenarios. Cloud deployments can only add to the latency.
4. Even after accepting all above points, some enterprises may raise eyebrows on **data movement**, as it is really a big challenge to move tens and hundreds of terabytes of data from the on premise to cloud environments. Cloud providers have come up with offline as well as online data integration tools to help solve this issue. But the real challenge is to make data movement to and from the cloud a seamless part of the enterprise data flow. For cloud analytics, enterprises must think about two-way data movement using streaming data pipelines.
5. There is also a belief that **cloud products feature set** does not match the features of their respective on premise products. While this may be true for few products, most cloud products currently provide better product features and support than on premise version.
6. Enterprises call cloud implementations as **black boxes** as they lose visibility and control over it. At times, they find it hard to tune or configure parameters which they could easily setup when everything is in their control.
7. Last but not the least, don't forget the **investments** already made by enterprises in on premise deployments. There must be real ROI before them moving to cloud. Usually these should be planned along with major hardware/upgrade cycles.

3. Factors to consider for Cloud Analytics

This section lists a variety of factors that must be considered when moving an analytics implementation to the cloud or deploying a new implementation. Factors are grouped in four categories:

(i) Requirements on the *data* used to derive analytics insights (beyond big data's popular "three V's", the types of data and the needs for integration, and quality);

(ii) Requirements on the *queries* needed to derive such insights (whether they are known or not, their type, analytic workload, response times and user scales issuing these queries);

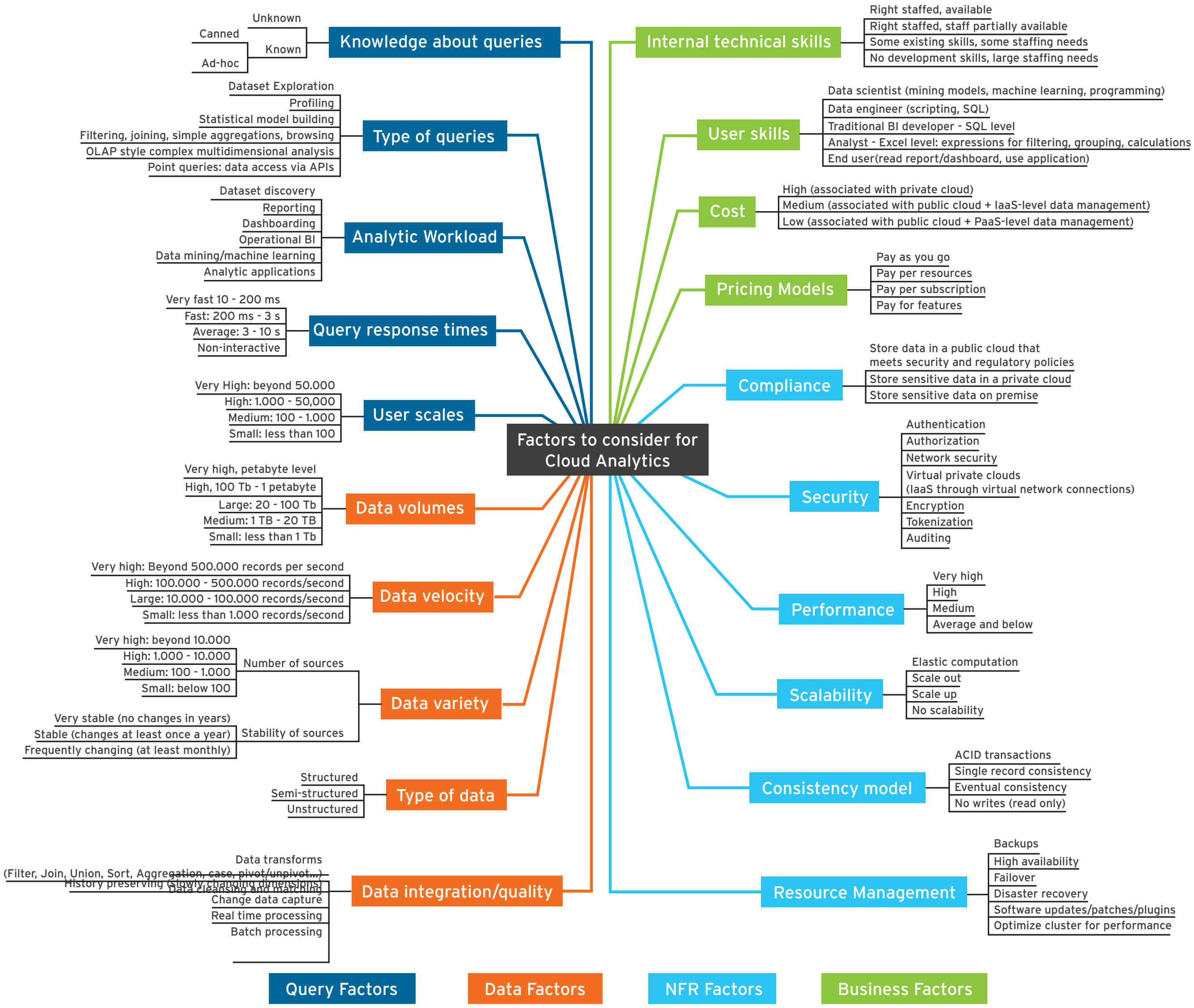
(iii) *Non-functional* requirements such as security, compliance, performance and scalability, management of resources in the cloud, and consistency model under update failure (if the analytics solution is embedded in or deeply related to a read-write application); and

(iv) *Business* requirements such as internal technical and end-user skills, cost, and pricing model.

Please refer to the figure below where groupings are color-coded for easier understanding. These factors should be self-explanatory (except for performance and scalability, on which we comment below); they are presented as a collection of “requirement dimensions”, each with possible value points, or members. Sections 4 and 5 below will be constantly referring to these dimensions (which will be written in **bold** when referring explicitly to them) and their member values.

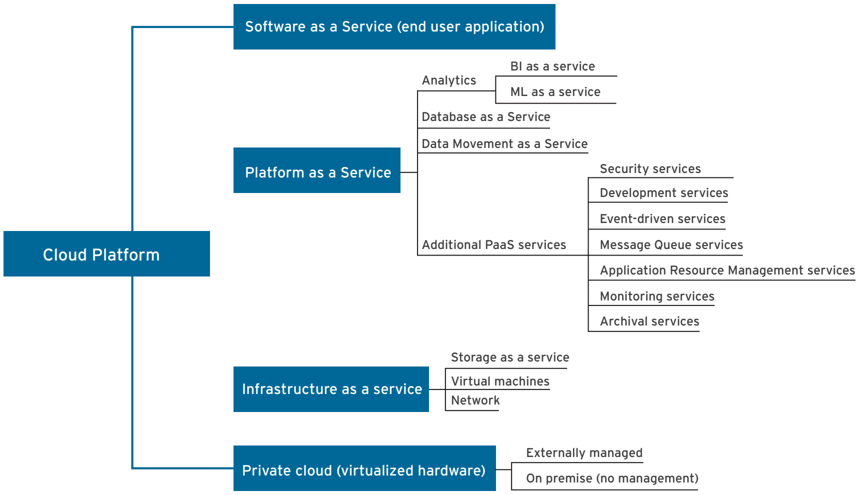
Customer requirements should be accurately expressed using members from these dimensions. Some dimensions take unique members to describe requirements: for instance, the **Data Volume** dimension, a single value such as “small: less than 1 TB”, or “medium: 1 TB – 20 TB”, etc., is generally assumed to be chosen (although not absolutely required if there are several datasets on which a separate analysis is required). On the other hand, some dimensions generally take several values: an example is **Analytic Workloads**, which may take “Reports”, “Data Discovery” and “Dashboards” for a given customer.

Performance is a complex, derived requirement that can be viewed as a function of query and data requirements. It generally refers to sustained query response time during a time interval, on a well-specified workload under control where type of query mix, user scales and data volumes are precisely defined. The same can be said about **scalability**, a related concept which corresponds to the ability to overcome performance limits by adding resources, so cost is involved.


Factors to consider for Cloud Analytics
Knowledge about queries
Unknown
Canned
Known
Ad-hoc
Type of queries
Dataset Exploration
Profiling
Statistical model building
Filtering, joining, simple aggregations, browsing
OLAP style complex multidimensional analysis
Point queries: data access via APIs
Analytic Workload
Dataset discovery
Reporting
Dashboarding
Operational BI
Data mining/machine learning
Analytic applications
Query response times
Very fast 10 - 200 ms
Fast: 200 ms - 3 s
Average: 3 - 10 s
Non-interactive
User scales
Very High: beyond 50.000
High: 1.000 - 50,000
Medium: 100 - 1.000
Small: less than 100
Data volumes
Very high, petabyte level
High, 100 Tb - 1 petabyte
Large: 20 - 100 Tb
Medium: 1 TB - 20 TB
Small: less than 1 Tb
Data velocity
Very high: Beyond 500.000 records per second
High: 100.000 - 500.000 records/second
Large: 10.000 - 100.000 records/second
Small: less than 1.000 records/second
Data variety
Number of sources
Very high: beyond 10.000
High: 1.000 - 10.000
Medium: 100 - 1.000
Small: below 100
Stability of sources
Very stable (no changes in years)
Stable (changes at least once a year)
Frequently changing (at least monthly)
Type of data
Structured
Semi-structured
Unstructured
Data integration/quality
Data transforms
(Filter, Join, Union, Sort, Aggregation, case, pivot/unpivot...)
History preserving (slowly changing dimensions)
Data cleansing and matching
Change data capture
Real time processing
Batch processing
Internal technical skills
Right staffed, available
Right staffed, staff partially available
Some existing skills, some staffing needs
No development skills, large staffing needs
User skills
Data scientist (mining models, machine learning, programming)
Data engineer (scripting, SQL)
Traditional BI developer - SQL level
Analyst - Excel level: expressions for filtering, grouping, calculations
End user(read report/dashboard, use application)
Cost
High (associated with private cloud)
Medium (associated with public cloud + IaaS-level data management)
Low (associated with public cloud + PaaS-level data management)
Pricing Models
Pay as you go
Pay per resources
Pay per subscription
Pay for features
Compliance
Store data in a public cloud that meets security and regulatory policies
Store sensitive data in a private cloud
Store sensitive data on premise
Security
Authentication
Authorization
Network security
Virtual private clouds (IaaS through virtual network connections)
Encryption
Tokenization
Auditing
Performance
Very high
High
Medium
Average and below
Scalability
Elastic computation
Scale out
Scale up
No scalability
Consistency model
ACID transactions
Single record consistency
Eventual consistency
No writes (read only)
Resource Management
Backups
High availability
Failover
Disaster recovery
Software updates/patches/plugins
Optimize cluster for performance
Query Factors
Data Factors
NFR Factors
Business Factors

We have chosen to expose performance and scalability also as separate dimensions belonging to the non-functional requirements category. We distinguish performance levels as being Very High, High, Medium or Average, as well as types of scalability such as Scale Up, Scale Out and Elasticity. Please refer to Appendix 1 for a discussion about what is meant by these terms. Our performance scale will be applied informally throughout this document, for the most part ignoring workloads (so it should be taken as a very rough indicator).

Customer requirements must be mapped to cloud platform services (see below), and a choice of provider.



4. Decision points for selecting cloud technology, services and provider

Whatever the drivers may be, whether cost reduction, IT staffing difficulties, or business drivers (e.g., enhancing business processes or improving customer experience), we assume in this section that the organization that Persistent (PSL) is servicing is convinced of the benefits of moving its analytics operations to the cloud, or developing a native cloud analytics solution.

This section provides technical input intended to help with the process of mapping the customer requirements to cloud technology, services and platform provider. It is written as a series of decision points that the customer must go through. Each decision focuses on a specific area, points out the choices to be made at each step, and considers the relevant requirement dimensions out of the 20 dimensions introduced in the previous section. We will be referring to this organization as "the customer" (we might also be referring to it in second person, as in the expression "your requirements").

4.1 The type of cloud (public, private or hybrid)

The first decision enterprises must make when moving data to the cloud is to choose the right environment. While public, private[1] and hybrid clouds all have benefits, determining which model best meets their needs is a crucial step on the path to the cloud (both migrations and new deployments). Organizations must make this decision based on various dimensions described below:

a. **Security** and **compliance** – As mentioned in section 2, these are the top two obstacles for moving the data to the cloud. At the same time, security has also been recently identified in [1] as a technical driver for adoption of cloud analytics. Indeed, as cloud platforms mature, fear of security issues have lessened: products from platform vendors such as Azure SQL Data Warehouse, Google BigQuery, Amazon Redshift as well as other infrastructure offerings (e.g., Virtual Public Networks) have swaths of security features to guarantee the safety of customer data at every point in its journey.

[1] It should be understood here that we refer to a dedicated deployment (thus, for a single tenant) on virtualized hardware managed by an external provider –otherwise we are talking simply about an on-premises deployment.

This calls out for a strategy based on checking with cloud service providers. Public cloud providers may report that your specific industry regulation guidelines (e.g., HIPAA for healthcare) have been implemented. Security and privacy guarantees (e.g., virtual networks, encryption, authentication and authorization) may also prove to be sufficient. On the other hand, if for whatever reason one of these two hurdles is not cleared, the remaining possibilities are either a private cloud deployment or a hybrid cloud, where the more sensitive data is stored on premise or in a private cloud, and the remaining data is stored in the public cloud, and a virtual network connects the two environments securely.

b. **Performance** and **scalability** – Based on the performance requirements of analytics queries, again customers might decide to go public, remain private, or even opt for hybrid.

There are certainly more guarantees for predictable query performance for single tenants (this designation includes both private cloud and public cloud IaaS providers) than on public multi-tenant clouds. While auto-scaling technology in public clouds may scale resources up or down elastically, noisy neighbors with heavy spikes in demand can still be an issue[2]. Even though there are ways to mitigate this (e.g., by throttling mechanisms implemented by public cloud providers to prevent heavy tenants from consuming too much resources), single tenant implementations are more of a fit for mission-critical workloads.

Some private cloud providers are also hardware vendors (e.g., HPE, Dell, IBM which are in the top 5 spots of this market), so customers can buy cloud hardware and services from them. Some also let customers bring their own hardware. Thus, with private cloud providers, customers have more control to choose the appliance or scale-up server that is known to suit their needs. On the other hand, technology running on virtualized hardware offered by both private cloud and public IaaS vendors is generally horizontally scalable to accommodate growing data volumes and numbers of users consuming analytics services.

When requirements for query latency are in the sub-second range, transfer times for data from the cloud to the client and vice versa will be higher than acceptable. In such cases, customers may want to investigate direct network connectivity services offering more reliability and lower latencies than typical Internet connections (all major public cloud vendors offer such products for lease or purchase –of course, this adds to the cost). For queries on datasets demanding very predictable, very fast response times, leaving these datasets on premise may be the better solution. As with security and compliance, performance may also be a driver for hybrid clouds.

c. **Pricing models** and **cost** – Private cloud offers large cost benefits when compared to on-premise by eliminating heavy upfront costs: no hardware must be purchased (necessarily), no data center costs, and substantial operational costs are saved, as less IT personnel is needed. Nevertheless, there will be some upfront costs as well as an ongoing cost. Public cloud pricing models can be either subscriptions for a predefined period or "Pay as you go", and it will be generally cheaper than ongoing costs of private clouds. Of course, the cost related to technology used at a given level of service in public clouds is very relevant; we will examine this in the next section below. Beware, pricing structures are not standardized and may be complex. Hence, depending on expected timeframe of your analytical solution, whether it is PoC or production deployment, compute the TCO in both scenarios, and set up trial accounts with different cloud providers before deciding.

d. **Internal technical skills** – This is also one of the deciding factors: if the IT team does not have the skills to manage infrastructure, going with public cloud will make more sense. Or they might have the skills, but some may feel challenged or threatened by this new way of working. However, as with previous technological shifts, cloud solutions open new opportunities for IT staff, so it may turn out to be a positive change. Based on this dimension, enterprises can also choose to go with hybrid cloud.

[2] In PSL we have experience of an extreme customer case where, because of cost considerations, the architecture that was decided was to allow 40 tenants in an Amazon Redshift single node cluster. Beyond 5 tenants issuing heavy queries, the system could not scale and overall query performance suffered.

Because they try to combine the best of both worlds, hybrid clouds deserve to be described in a bit more detail. Hybrid clouds allow you to get the following benefits:

- Security, compliance, predictable performance and reliability from private clouds, and
- Horizontal scalability and elasticity, and lower costs of the public cloud.

Uniformity may still be achieved in hybrid clouds, as platform services from a cloud provider may also be installed on premise: this way, your developers can build applications using a consistent set of services and DevOps processes and tools, and then deploy data to the location that best meets your security, compliance and performance requirements. However:

- Hybrid clouds are more complex to implement (so total cost of ownership may be impacted),
- Overall security and compliance is not guaranteed, as organizations must reshape their governance to ensure data is still properly protected, and
- Success may depend on the maturity of management and maintenance.

4.2 The cloud database management level

At this point we focus our attention on the cloud database at the core of our cloud analytics high level architecture. There are two different aspects to explore: the service level at which to manage the database, and the specific type of database technology. At this stage of development of cloud computing technology, these two aspects are largely independent, as we can now find different types of databases offered both as a service, or available to install and manage on the cloud on top of infrastructure managed as a service.

This section tackles the level of management aspect, and the next section delves on specific database technology for your requirements. This first aspect will be useful as well in exploring other two options: hybrid clouds, for obvious reasons, as well as private clouds, given the similarities that exist between public IaaS services and private clouds. In what follows, what is said of public IaaS applies to private clouds as well, unless explicitly pointed out otherwise. The dimensions below are relevant to evaluate this decision.

a. The choice of service level when managing your database seen as a resource –this is part of **Resource management**, which may depend on your **internal technical skills**.

   With *IaaS*, you get close to the flexibility of an on-premise environment in terms of storage, networking and computing. You set up your environment: number and size of virtual machines (in terms of number of cores and size of RAM); then, storage, which is separate from processing power, where you choose capacity and disk throughput; and finally, network bandwidth and isolation (standard internet, direct connectivity or virtual private network[3]).

   On top of this infrastructure, you then install your database software (your own license or an image you pay as you go), and you manage it almost[4] as usual, assuming responsibility for backing up, configuring for high availability, monitoring, patching and scaling up (and down) as needed. IaaS services provide automated features to dramatically simplify these activities. The Appendix 2 complements our running example in section 5 with a cloud platform provider (Microsoft Azure) by providing detail on the management of the infrastructure layer and the tools available to IT administrators to manage a DBMS running on VMs.

   Cloud infrastructure providers only provides availability SLAs for VMs and not on the software installed or services provisioned on top of infrastructure. Typical SLAs are in the 99.9% uptime range, which

[3] VPNs are used to connect to public clouds. Private clouds are usually accessed from on premises through a physically isolated network, so this does not apply to private clouds.

[4] Management is not exactly as usual when compared to on premise: for instance, reasons for enabling backup no longer include protection against media or hardware failures, as the IaaS VM service provides this already (private clouds also do).

correspond to monthly downtime in the 45-minute range. With some work to configure redundant resources, you can get higher availability when installing a DBMS on top of VMs. For disaster recovery, the picture is the same: at the IaaS level, administrators must understand the tradeoffs between the existing options offered by the DBMS vendor working on IaaS hardware, and configure and manage them.

On the other hand, at the *PaaS* level, you have database as a service (with several technology variants including data warehouse and Hadoop as a service, described below); what follows immediately applies to relational DBMSs and warehouses, and we refer to it as DBaaS.

Your database will share infrastructure resources with other tenants. DBaaS forces you to relinquish control on the infrastructure: your visibility of the hardware and the virtual machines it runs on disappears, so it is a departure from the traditional way of managing your database. The underlying database is, for the most part, managed: you do not have to spend time managing upgrades, high availability, disaster recovery, or backups. Your developers use directly the service to provision data, optimize queries and user workload, and develop an application. As we will see below, computational and storage resources can be scaled out automatically, and they have state of the art fault tolerance features.

With DBaaS, you get out of the box much higher availability SLAs from the cloud platform vendor. For disaster recovery, the picture is similar: DBaaS come with built-in disaster recovery. For instance, Azure offers out-of-the-box geo-replicated full and differential backups. Finally, rapid failover [5] to another site in planned events (e.g., maintenance) or in failure events with minimal data loss can also be configured much easily with DBaaS as compared to DBMS on IaaS.

b. **Knowledge** and **types of queries** – Legacy database and data warehouse environments (where queries pre-exist, so they are completely known) are best migrated to the cloud layering the DBMS on top of IaaS services. Legacy code may be difficult to port it to a database-as-a-service offering. For new developments, the choice between DBaaS and DBMS on IaaS depends of the requirements discussed in this section.

c. **Performance** and **scalability**, depending on **data volumes**, **user scales** and **cost** – Per our volume characterization, large volumes start at 20 TB. Below this level, traditional SMP databases are now able to handle loads of relatively complex queries for medium size concurrent number of users and average to fast response times. SMP DBMSs running on IaaS services can be a fit, especially if there are there are no drastic variations on volume or user scale, i.e., when the lack of elasticity is not really a problem. DBMS on IaaS gives you the advantage of predictable performance, as you will be the only tenant –but, at a price point above DBaaS, as discussed below in more detail.

At larger volume scales, or larger user scales, with queries still needing average to fast response times, it is worth considering MPP data warehouses as a service, which have a parallel distributed capability that can take advantage of scale out architectures[6]. This type of databases, very expensive when purchased on-premise, are much more affordable on the cloud especially when licensed as a service, and they outperform SMP databases by far (several times faster, up to 10 times faster on query intensive workloads, depending on volume; the larger, the bigger the difference). Database services on MPP differ in terms of their architecture: some, as Azure SQL DW, separate and scale independently (and elastically) compute nodes and storage nodes; some, like Amazon Redshift, don't perform the separation and scale compute and storage nodes jointly, so may be costlier in some cases. Volumes in the price list go up to the low/mid hundred TBs of compressed data which, uncompressed, is about 1 PB; DWaaS do not go past petabyte size yet.

[5] This in fact applies more to transactional databases on the cloud, not so much to cloud analytics databases.

[6] There are also (SMP) databases as a service in most cloud platform providers as well. And there are in-memory, columnar databases on scale-up servers which are also a solution for a large category of analytics workloads.

In any case, data architects still need to design carefully a partitioning schema in the cluster to optimize your most important queries. Another important aspect is right-sizing for concurrent user access. Both Google and Azure have schemes based on concurrency slots, which are units of computational capacity –and the more slots you reserve the more you pay.

At the intersection of the cloud and big data you now have the fast-growing *Hadoop as-a-service* (HaaS) market, which you should consider when volumes are petabyte size, query response time demands are in the average range (several seconds[7]), and you want to rein in costs.

The biggest drivers for HaaS are reducing the need for technical expertise and low upfront costs, the former being more of a driver than with the more traditional databases, given Hadoop's management complexity. Amazon Elastic MapReduce (EMR), the HaaS service by AWS, is the largest player (described in 10.2.2.9). Microsoft, Google and IBM also have their own offers described below. Other vendors include Cloudera, EMC Pivotal, Qubole and Altiscale, now part of SAP. Most of these competitors initially provided *Run It Yourself* deployments: they were hosted Hadoop on top of IaaS, really. Some are starting to propose complete running and management of the Hadoop jobs. Managed Hadoop typically provides no real multi-tenancy (no sharing of cluster nodes among tenants), but provides elastic, auto-scaling clusters where nodes are added or removed depending on SLAs for jobs estimated in advance by IT users.

d. **Pricing models** and **cost** - The combined cost of data management[8] tools supported through public IaaS services is generally costlier than the corresponding public PaaS data management services, so being a single tenant is, well, a bit of a luxury. The running example in section 5.2.2 will give you a more concrete idea but, of course, this depends on your use case and platform provider. Pricing schemes for DWaaS are widely different from one provider to another: some separate storage from querying activity costs while some bundle it; some include concurrent usage concepts; some offer prepayment options to lower pay-as-you-go pricing, which makes sense in stable production deployments where you are sure to use the service for a term measured in years. Pricing for HaaS is simpler and typically include fees for its service, based on usage on top of storage costs, the latter being IaaS storage costs, which are cheaper than those that DWaaS charge (at least, when fees for storage is visible in the cost structure).

e. Cloud Computing ecosystem: Data management and other PaaS services –it almost goes without saying that the ecosystem of a cloud platform provider in terms of tools, services and partners will be crucial in determining the final choice. If your organization has already settled on a specific tool or service, this reduces the choices (it could be argued in this case that the presence of such a tool in the cloud platform ecosystem is a customer requirement). Beyond infrastructure and data management, today's platforms offer a very wide spectrum of platform technology services which include data, server and VM migration tools, security services, developer tools, IoT and machine learning services, application services (e.g., API and workflow management), mobile services, media services, cognitive services, monitoring services and more. Of this long list, the most relevant category might be migration tools if you have resources to migrate from on-premise to the cloud (for detailed steps in migrating data warehouses to the cloud, please refer to [9]); and security, given it is a common deterrent for organizations considering both cloud migrations and new cloud deployments.

4.3 The data model for the cloud database

This section is just a refresher on the type of data engine one would use on-premise based on the data and the query category requirements introduced in section 3 (for a longer treatment of the subject, please refer to [15]). The cloud nevertheless does open possibilities in this area: as analytics use cases are becoming more complex (given the appetite for insights on data increasingly available from so many sources), they may necessitate more than a single engine to do the job, and it is much easier and cost-effective to architect these kinds of "polyglot persistence" solutions using services in the cloud as opposed to installing different engines on premise.

[7] On Spark, the fastest Hadoop technology, which is still about 1 or more orders of magnitude slower when compared to MPP data warehouses.
[8] This term refers here to databases, BI and data movement tools and services.

In a nutshell, here are the broad guidelines for the database technology to use given requirements around the data requirements category and the query requirements category.

- Use traditional SMP relational databases when data is structured, volume, velocity and variety are mild to medium level, queries are known and concern mainly operational data with low requirements on aggregations, and you want a minimal lag on availability of the most recent data. If you have concerns with disturbing users using your operational application with the analytics load, consider replicating operational data to a separate instance. Most replication solutions work in real time these days.

- Use MPP relational data warehouses (or in-memory, columnar databases on scale-up servers) when data is structured, volumes are large but up to several hundred terabytes, velocity and variety are mild to medium, queries are known in advance and need complex aggregations and calculations, and workloads are for a low to medium user scale (under a thousand). When queries are multidimensional in nature, need additive, semi-additive or non-additive aggregations (e.g., sums, averages or ratios) based on flexible hierarchies, and predictable performance is needed, then you should consider using an OLAP engine.

- Use Hadoop (which includes both Hive and its recent variants, and Spark) when your data is very large, has a lot of variety or comes in different types (e.g., semi-structured web logs or text files, but also structured data from many sources), has large velocity or beyond, and/or when your queries are not (entirely) known in advance. Perhaps the main characteristic of Hadoop engines is that the data precedes its schema, so it can easily accommodate big volumes and big variety. On the other hand, query complexity, performance and high concurrency is not their forte: In Hadoop for analytics, or data lake, you must start by discovering the structure of data, layering a schema on files and eventually transforming data before querying. If query complexity and/or performance is a concern, this architecture still allows optimizations, such as overlaying an OLAP model for querying from Hadoop data with OLAP-as-a-service offers [10], [11], or exporting key/value data from Hadoop for fast querying with tools such as Cassandra or ElephantDB; the choice depends on the type of queries you might have –of course, this adds more cost to the overall solution. Hadoop is also a good platform for search engines indexing all types of data. As for big velocity, you can now use Spark, Hadoop's most recent project incarnation, which provides streaming capabilities for fast incoming data[9].

- Use NoSQL systems when your data volumes are large, data velocity matters and may be large, data structure is not as relevant (it may be structured and even nested –e.g., JSON, XML, but you want to store it as is), and you need scalability and performance for online writes and simple reads.

  NoSQL systems are not meant for complex querying or OLAP style aggregation but rather for operational systems with simple analytics requirements, or for search-intensive applications. NoSQL databases trade off stringent consistency requirements (in the sense of the **consistency model** requirement factor) for availability[10], and are modeled for querying patterns for speed and agility (very simple queries or APIs and updates involving one or a few records accessed by their key).

To address consistency under write failures in a more comprehensive way, a data processing architecture, Lambda architecture [8], has been introduced and is now popular. It combines the use of batch (e.g., Spark) and stream data processing methods (e.g., Storm plus a NoSQL database). It applies to systems that try to provide to large, geographically disperse user scales the ability to query, with acceptable latency, continuously updated very large data volumes (hence, distribution and geo-redundant data is implied), where updates must be visible immediately. Lambda architecture chooses availability over consistency, arguing that sacrificing availability is bad for business and at least eventual consistency is needed[11].

[9] As Spark was a late comer, other popular solutions were crafted before such as Kafka, a distributed message broker, and Storm, a scalable, fault-tolerant, real-time analytic system.

[10] There's typically no transaction semantics beyond single record writes: this is also related to scalability, as discussed in Appendix 1.

[11] The CAP theorem [14] states a database cannot guarantee consistency, availability, and partition-tolerance at the same time. But in fact, the choice is really between consistency and availability when a partition happens; at all other times, no trade-off must be made. Database systems designed with traditional ACID guarantees choose consistency over availability (designers had LAN scenarios in mind), whereas NoSQL databases, more recent and scale-conscious, choose availability over consistency. The main contribution of the Lambda architecture is in isolating the complexity involved in maintaining eventual consistency in the stream processing part, containing only a few hours of activity, as the batch part constantly overrides the stream layer and manages a fault-tolerant, append-only, immutable state.

As you can see, no single technology can capture all requirements. Even two or three requirements, but taken to their extreme, are either difficult to satisfy with a single existing technology, or a combination of different database technologies is needed (Lambda architecture is in fact an example). Let us illustrate this with a couple of examples.

a. At Facebook, the analytics group had to provide OLAP style queries with very low latencies and very high velocities [12]. They experimented with several technologies, nothing worked and, at the end, had to build a data and query execution engine that worked for them[12].

b. Without extreme query performance requirements, the variety of analytics tasks may bend an architecture towards polyglot persistence, as in the case of Flipkart, an eCommerce company [13]: large incoming data volumes, data processing at different velocities (both real time and batch), and an analytics layer requiring ad-hoc analysis, search, machine learning and canned reporting. Their data layer includes Hadoop (Hive, Spark), Storm, Vertica (an MPP warehouse) and ElasticSearch (see 10.4.2.11). If the high velocity requirement is dropped there would still probably be Hadoop, Vertica and ElasticSearch in the picture, given the analytics requirements.

4.4 The BI / analytics tools

This is a very important aspect of decision making being the one that most impacts business end users. The modern cloud database needs to support the breadth of tools that organizations can use to get actionable results from the data. BI is a good fit for the cloud when the visualization tools are close to where the data is, which is now the case with cloud analytics. The choice of BI/Analytics tool depend on several dimensions.

a. **Query types** –Traditional BI tools were built for the reporting analytic workload; ad-hoc querying and OLAP came later and have more “free-form” user experiences and interfaces, sometimes imposing limitations on the types of queries that may be defined (see section 5.2.4).

b. **Performance and scalability** – This is an area normally associated with the database / data warehouse layer, but the analytics layer also contributes to the overall time spent (again, refer to section 5.2.4 below for an example).

c. **Analytic workload** – If the requirement is about reporting or dashboarding, then most cloud platforms also provide solutions e.g. from SAP, IBM and Oracle as SaaS services from their own clouds or from Azure and AWS. However, if you are looking for exploration and discovery use cases, then look for tools like Tableau, Qlikview, etc; these are mainly desktop solutions but can work with cloud sources and can publish reports and dashboards to the cloud. For machine learning use cases, cloud service providers do offer them as a service, e.g. Amazon ML, Azure ML, Watson Analytics. Google Analytics offers complete BI stack in the cloud: it not only offers visual data discovery, exploration, collaboration, and reporting, but also analytic applications for marketing, sales, service, and social platforms. Finally, if the requirement is to build a full-blown solution in a given vertical industry, then we are talking about embedding analytics capabilities in an application that is to be built and deployed using PaaS *development services* and tools.

d. **Data integration / data quality** – Also referred to as data preparation, it has been recently recognized that it is highly desirable, in a modern BI toolset, to include features to integrate data coming from different sources and address the heterogeneity of data representations, conventions and standards, missing values, as well as duplicated records, that impact the quality of data. The most common way this is being addressed is by loosely coupling self-service *data preparation* tools with BI tools, as will be explained in the next section.

[12] At the root of the problem, OLAP engines operate on mostly static datasets

e. **End user skills** – A growing demand for easy to use tools accessing trusted data in the cloud has created a shift in the BI market towards governed self-service. Organizations can now enable broader access of analytic insight to remain competitive without requesting their business users to improve their technical skills: business users can analyze data without necessarily having to write queries in SQL, as they did with Excel, but now through more powerful tools such as PowerBI, Tableau, and Qlik Sense. On the other end, traditional BI product suites require dedicated IT resources with developer skills, as they are more complex to implement; as mentioned above, most are available as SaaS services and can also be used in single tenant mode installed from the cloud provider marketplace directly on top of their infrastructure. The platform also needs to support a new breed of users, data scientists, who run experiments with the data, develop predictive analytic and ML models, and assist in real-time decision-making.

4.5 Data movement /ETL tools

Once you have decided a data model for your cloud database, you also need to decide how to transform and load data from one or multiple sources into it. Integration and data movement was identified in [1] as the second leading obstacle, after security, to cloud adoption, pointing to the critical importance of full-featured data integration tools for the cloud. For this reason, you might need to consider this before making the final choice of cloud platform provider.

a. **Data Integration** and **Data quality -** Data needs to be integrated and processed for quality either when it is written in the cloud data warehouse schema, a simpler NoSQL schema, or at a later point in time, in a data lake. Make sure your transformation needs are covered, whatever your data requirements might be. Possible pitfalls of PaaS data movement / ETL include (i) reusing legacy transforms: this is generally not supported, as the tool that was used on premises is not the same as the tool retained for the cloud[13]; (ii) the non-availability of quality specific transforms such as cleansing and de-duplicating, which are present generally in more mature on-premise tools; (iii) processing data at high velocities (see below): typical transformations on high velocity data may include joining data from multiple streams, and rolling window aggregation functionality; and (iv) make sure there is a comprehensive data lifecycle management and administration capabilities.

We believe that development productivity remains a serious obstacle in the cloud, as with on-premise ETL. Self-service data integration tools such as Trifacta and Alteryx (see last point below) are a possible path for mitigating this problem.

b. The choice of service level when managing your data movement tool –as with cloud databases, with data movement tools there is also a deployment choice between IaaS or as PaaS, so this can be seen as part of **Resource management.**

IaaS deployment of traditional ETL tools is a way to solve the DI/DQ pitfalls enumerated on the previous point, as they are still more mature than PaaS data movement tools. Internal technical skills, analyzed separately below, may also weigh in on the final choice. On the other hand, PaaS data movement requires less administration, management and setup than traditional ETL deployed on IaaS. As with their cloud database platform service counterparts, availability and scalability of ETL tools is also taken care by the PaaS provider: this matters with large data volumes (see more on this requirement below). PaaS data movement tools are much more likely to outperform 3rd party ETL tools, for instance, by taking advantage of parallelism in data transfers to internal nodes of a target MPP data warehouse cluster, something JDBC based connections of non-native ETL tools will have a hard time doing (especially if running outside of the cloud provider).

[13] Even within the same vendor, we have found that the on premises tool and the cloud tool are not always fully interoperable. In this case, one possible option is to deploy the on-premise tool on IaaS; another is to use the on-premise tool from your premises if it supports connectivity to the selected cloud database (requiring IT administrators to open an external communications port, something that administrators don't easily allow).

c. **Data volume** – One of the biggest challenges is when you have a huge volume of data available on premise and want to move it to the cloud. There are migration tools and fast networks connections available; however, if the volume is measured in hundreds of terabytes, then customers need to rely on a physical hardware based solution to transfer the data. Indeed, even at 100 Mbps[14], transferring 1 TB takes about a day. Most cloud providers including AWS and Azure provide such a mechanism. With these solutions, transferring 100 TBs can be done in a matter of a few days, including shipping time, as opposed to several months.

d. **Data velocity** – Today, business and IoT applications generate huge amounts of data per second and businesses may require that data to be consumed and analyzed in real time, as shown in the previous sections. Each cloud provider has its own data movement tool (Amazon Data Pipeline, Azure Data factory, Google Cloud Dataflow and IBM data connect), but if the velocity and throughput expected are high, open source solutions such as Kafka, Spark and Storm are available in these cloud platforms.

e. **Data variety** – In traditional data warehouses the typical number of enterprise data sources is low (in the tens of sources), so data processing to transform the incoming data to the target data structures is highly optimized, uses sophisticated techniques to deal with changed data capture at the sources, slowly changing dimensions to keep history, stability of source data structures, and the like. A similar pattern exists with cloud data warehouses.

   However, as explained in 4.3 above, medium and large data variety available for analysis has driven the use of Hadoop as data lakes for analysis, both as a service and managed by the customer on the provider's infrastructure, and has increased demand for a data movement style which takes data from these various data sources, loads it in Hadoop without a predefined schema necessarily, and defers transformation at a later point in time for further analysis –an ELT style, where the transformation part is done through Hadoop programming. The data movement/ETL tools you pick specializes in this case in extraction, so you should make sure it connects to the large variety of data sources you need, or has at least a development kits that enable customers to create custom adapters for non-supported systems. If change data capture or stability of source structures is important, as this topic is more mature with data integration tools than it is with Hadoop, we recommend picking a data integration tool (e.g., Informatica) that supports it directly on top of Hadoop.

f. **Internal technical skills** – Traditionally, developing ETL processes required heavy, intensive coding to operate. As time passes they become difficult to maintain and extend – often due to the original coders moving on and new developers not understanding the code. If, on top of all the changes involved in moving analytics to the cloud, a change of ETL tool is forced upon the team, this may prove to be too much; in this case, an option is to keep using the same tool on the cloud infrastructure environment to increase the IT team's comfort level.

   On the other hand, self-service, cloud-ready tools have recently entered the data integration and data governance field, allowing business analysts to develop their models with data-driven visual GUIs, without explicit coding. These tools complement and integrate with self-service BI tools and are becoming realistic alternatives for business units and departments to process data themselves, without or with limited IT assistance, while keeping some degree of IT governance.

4.6 Additional PaaS services

Apart from data management services, there are many other PaaS services available which can be used for your end to end processing.

Writing custom code responding to a variety of events is addressed through event-driven *functions-as-a-service* (e.g. Microsoft Azure Functions, AWS Lambda Functions or Google Cloud Functions). They differ in term of the language they support; see section 6 for a comparison. The main advantage is that this custom code once deployed as function as a service becomes automatic scalable and secured.

[14] Which is a fast connection, twice as fast than today's fiber optic cables.

Cloud platforms also provide *message queue services* (e.g. Azure Service Bus, Amazon SQS, Google Cloud Pub/Sub) which allows decoupling your components. The service provides asynchronous operations which enable flexible, brokered messaging between clients and servers, along with structured first-in-first-out (FIFO) messaging and publish/subscribe capabilities. While these are mainly used in transactional systems, this service could be leveraged for implementing analytics rules.

Providers also have started offering a variety of services to build applications (e.g., *API* and *workflow management*), as well as *mobile services, media services,* and *cognitive services*.

Cloud platforms also provide *backup* and *archival services* to import and export data to less costly storage. They can also be used to encrypt data and keep it secured until it is required to be accessed. They are generally available as part of the cloud database services (e.g., Azure), but some providers propose separate services for these tasks (e.g., Amazon has AWS Glacier as a service).

In a cloud platform, you also need **Resource management** for both platform and *application-level resources*, i.e., support services for creating, deploying and managing data management and application artifacts. Each cloud platform does provide those also as a service e.g. Azure Resource Manager, AWS CloudFormation, Google Cloud Deployment Manager. This helps create flexible templates that deploy a variety of Cloud Platform services, including your cloud database / data warehouse, data movement and your BI application. Finally, platforms allow to monitor resources running on the platform: to collect and track metrics, collect and monitor log files, set alarms, and automatically react to changes in your resources (AWS CloudWatch, Google Stackdriver, Azure ApplicationInsights).

5 A complete example

The customer is a large business process outsourcing and professional services company, serving both the public and private sectors. They work across eight markets, among which the education market. The customer has a technology solution to manage student, parental and staff information, and is used by a large proportion of schools in the country. The current production release, referred hereunder as “the OLTP product” is an on premise, client-server architecture solution based around Microsoft SQL Server with business logic handled by a custom .NET Framework module. Reporting and Analytics is an important module add-on of the overall solution.

PSL was tasked to develop a cloud-based reporting and analytics solution for the forthcoming version of this product. Both this analytics solution as well as the OLTP product will be moved to the cloud.

5.1 Requirements

We summarize in the table below the requirements for the customer’s cloud analytics implementation in terms of the dimensions defined in the previous section.

Factor	Requirement	Comments
Knowledge about queries	Queries are known: canned and ad-hoc queries.	
Types of queries	Canned queries (filtering, joining for operational reports; OLAP-style queries for analytical reports) Ad-hoc queries using potentially every field of every table and user-defined fields	Hundreds of reports, tens of dashboards

Analytic workload	• Operational reports to understand what is going on at this moment, based on granular level of detail. • Analytical queries (aggregations, slice/dice) for reports and dashboards identifying critical trends, problems, outliers, find out root causes. • [future release] Big Data – consolidate data from sources like curriculum, health, social • [future release] Predictive analytics – predict outcome and provide recommendations	• Operational reports ***Canned report:*** Get daily attendance of Pupils with SEN status from my class ***Ad-hoc report:*** Give me a list of all the pupils in my User Defined Field 'school choir' and their contact details • Analytical report: Get average attendance of pupils with SEN status for a given year at a monthly grain • Comparable number of reports of each category • Most users execute operational reports
Response times	Depend on type of queries. Simple canned operational reports < 1 Second; Medium < 2 Second; Complex < 3 Second; Dashboards are treated as multiple parallel simple reports; Analytic reports under 6 seconds	
User scales	10000 active users (10 per school) 1000 connected users (10% of active users) 100 concurrent users for simple reports	10 concurrent users for medium, 5 for complex 1000 schools per deployment unit (see below), each with 6 years of history
Data volumes	Large (not huge): 15 TB per 1000 tenants (schools)	
Data velocity	Small: ~ 830 records/sec (see comments column), which corresponds to 250,000 records every 5 min Leaves 2/3 of the capacity (10 min) for activity peaks	Synchro. with OLTP product every 15min, Avg #recs /tenant/15 min = 250
Data variety	Near term versions: data is well known, comes from OLTP product model. Beyond: small number of external sources (curriculum, health, social, etc.).	
Type of data	Structured for now; after next version of the product, there may be some unstructured data (social sources)	
Data integration / quality needs	Data transforms, slowly changing dimensions, change data capture, batch pull	Many entities with 1: m relationships that needed to be tracked for each change (SCD)

Consistency model	No writes –read-only reporting and analytics	
Security	Authentication based on school Id (tenant); Role based authorizations (row and column level); encryption of data at rest and on DB backups; Auditing; allow several BI tool clients	
Compliance	DPA (Data Protection Act) GDPR (General Data Protection Regulation)	
Performance	Sustained performance in the order of 384,000 completed queries in an hour based on customer analytic workload, which corresponds to Medium level performance as defined in Appendix 1	Customer workload has more operational query mix compared to TPC-H; However, performance characterization as Medium-level seems still adequate
Scalability	No real scalability, as number of users and data volumes largely known beforehand and (IaaS) infrastructure is statically provisioned	
Resource Management	Monitor and replace failed nodes, server backups and restore, OS updates/patches	SQL Server managed by the customer on IaaS initially (but see below)
Pricing Models	Pay as you go	
Cost	Low (associated with public cloud + IaaS) to medium (associated with public managed PaaS)	
User skills	Mostly end users; target skill level is 80% of users should create a simple list report with no training	
Internal technical skills	No development skills, large staffing needs	Reason why project subcontracted to PSL
Requirement on cloud computing ecosystem	SQL Server, Jasper Reports, Excel, Tableau	OLTP technology stack which based around Microsoft SQL Server

5.2 Decision points

We illustrate hereunder the thought process that the PSL analytics team in charge of the customer's product went through, following the decision steps introduced previously in section 4.

5.2.1 Type of cloud

Early on, the customer had decided they wanted to have a public cloud deployment. When the PSL team arrived, they had already decided to go with Azure VM infrastructure, given their technology stack which is based around Microsoft SQL Server.

With respect to the degree of influence of dimensions on the decision, the following can be said:

1. The dimensions that influenced most this decision was the pricing model ("pay as you go") and the low cost associated with Microsoft's public cloud.

2. Performance requirements seemed attainable (on the analytic workload defined by the customer, most reports should return in 1 second, some in more).

3. The same was thought of security and compliance: SQL Server provides workable options to encrypt personal identifiable information, and internal tables can be used to store metadata to control the data access per business rules and policies relevant to the customer's product tenants (schools).

5.2.2 Cloud database management level

The initial decision was to go with SQL Server managed directly on Azure VMs. The overall volume to manage is too large to a single SQL Server instance (over 200 TB). The solution adopted was to partition the data space in "deployment units" (DUs) managed through IaaS with an SMP database on each.

Each school tenant stores about 1 GB of OLTP data per year, and 6 years of history are required. This gives 6 GB per tenant. The working scale factor for data warehousing data expansion has been found out to be 2, which gives 12 GB per tenant. The customer expected number of tenants is about 22,000, so the total data warehouse size comes out to be 256 TB. SQL Server SMP architecture allows it to scale to the small tens of terabytes[15]. For this reason, the current design considers DUs of 1000 tenants each, storing 12 TB of data per DU. Each DU has been sized at 4 servers, each with 16 cores and 250 GB of RAM. There would be 22 DUs over time.

Of course, the customer will not be able to run a query needing data from more than one DU without extra-merging work; this need is nevertheless planned for future releases.

With SQL Server on Azure VMs, customers have the full administrative rights over a dedicated SQL Server instance and a cloud-based VM. This implies that the customer must have IT resources available

- To use the services managing the VMs and other infrastructure (storage and networks), and

- To manage SQL Server: the traditional management of SQL Server done on-premises needs to be done now on the cloud.

[15] A TPC-H world record was just broken by MSFT SQL Server on an SMP machine on the 10 TB category.

On the cost side, besides the IT resources costs for administering the database, SQL Server on Azure VMs is sold with an included license that customers pay per-minute and includes Windows Server and VM disk storage costs. Existing SQL Server licenses can also be brought in; in this case, customers will be charged for Windows Server and storage costs only. Pricing per month for the 22 DUs in this case comes out to be in the order of $250.000 to $350.000 per month (using VM configurations for the sizes presented above); this between 5 to 7 times more expensive than Azure SQL DW (see below), and does not include the cost of managing SQL Server.

Appendix 2 provides more detail on the management of the infrastructure layer and the tools available to IT administrators to manage SQL server on virtual machines in Azure.

Study on data warehouse as a service
The customer was open to go beyond the raw IaaS management level and wanted PSL to conduct a study to determine if Azure SQL DW, Microsoft's MPP cloud database service, made sense for their product.

The notion of deployment units, developed assuming a SQL Server implementation, would need to be revisited: indeed, a single Azure SQL DW could store the data of the entire set of school tenants, as it allows to store 240 TB compressed on disk; this compression allows the database to grow to approximately 1 PB when all tables are clustered with *columnstore* type, which is the default table type.

On the positive side, for the reasons we have explained in section 4, definite gains are expected on the performance and scalability, due to the MPP architecture and the elastic nature of the Azure SQL DW product. Cost should also be favorable, we talk about this aspect below.

On the negative side:

a. As of last year, there were unsupported features such as PK/FK constraints, more general constraints, unique indexes, UDFs, identity columns (required for DW surrogate key generation), triggers, indexed views, as well as certain T-SQL constructs.

b. Furthermore, certain limits such as 1024 active connections and maximum 32 concurrent users executing queries may be limiting, as it is expected that 100 concurrent users would be executing simple reports. However, Azure SQL DW can scale out and deliver results for small reports in fractions of a second, so this limit does not necessarily mean that the requested throughput can't be attained (this would need to be verified experimentally through a PoC, which was not part of the initial study).

Finally, the cost per 100 DW units (a measure of underlying resources like CPU, memory, and IOPS) at 1.21 USD/hour comes to be about $52,000 per month for 6000 DW units, the maximum in Azure SQL DW, which seems adapted to the size and load of this customer[16]. In comparison, this is about half of Amazon's Redshift on-demand pricing which, at a price of $5000 USD/TB/Year, comes to be $106,600 USD/month for the total 256 TB[17]. As it is a managed service, the price includes the devOps cost of managing the database. Thus, when compared to the cost levels we saw above for SQL Server on Azure VMs, multi-tenant is indeed less expensive than single-tenant IaaS deployments.

The first version in production will be running on SQL Server on Azure VM (an IaaS option), the main roadblock being the unsupported features needed to migrate the current application to Azure SQL DW. In later releases, as Azure SQL DW matures and features needed by the OLTP and the reporting/analytics application are supported, this choice might be reconsidered.

[16] Given the expected concurrent workload, 6000 DWUs would give 240 concurrency slots. Each query consumes one or more concurrency slots, dependent on the resource class of the query. Simple queries consume one concurrency slot. Queries running in a higher resource class consume more concurrency slots. This number of slots could probably accommodate the customer's 10 medium size queries and 5 complex queries, in addition to 100 simple queries.

[17] However, AWS allows you to pay upfront for 1 or even for 3 years, for steady-state production workloads, which offers anywhere from 25% up to 75% discounts over on-demand pricing depending the machine sizes and upfront payments.

5.2.3 Data model of the cloud database

As the queries are very well known and the performance requirements are rather stringent, the data model is a traditional dimensional data warehouse model.

PSL had initially decided to build an Operational Data Store (ODS) type of warehouse, using a very similar model to the OLTP normalized model, because (i) the large percentage of history tracking entities put a limitation on de-normalizing the data, and (ii) every field describing each entity was potentially used for ad hoc reports for slicing/dicing. The initial results for transactional reports were substantially above the requested performance requirements, due to the large number of joins between dimension tables in the normalized schema. Analytical queries were also off.

After relaxing the requirements in terms of history tracking entities and ad hoc reporting, PSL then proposed to follow a dimensional de-normalized model to serve both transactional reports as well as analytic reports on a particular effective date (for current date and time-travel queries). The model follows a shared schema approach where tables have a tenant-id column. In addition, PSL proposed to build aggregate tables along with Aggregate-aware feature in reporting tools like SAP BusinessObjects or IBM Cognos. Other database tuning and optimization around indexing, managing statistics and partitioning on tenant-id were suggested as well.

These decisions were accepted by the customer. The performance numbers at the database layer after this change were brought in line with the requirements.

Given that in the customer’s plans there is a will to handle unstructured and semi-structured data in the future, PSL also investigated big data stores such as Hadoop with MapReduce, Spark, as well as No-SQL databases. However, at this point, it was considered that moving to a big data stack would be an overfit, so building a dimensional data warehouse appears to be the most appropriate solution at this point.

5.2.4 BI / analytics tools

Jasper Reports is the tool selected by the customer and in production in the current version of the product; as such, it needs to be managed directly by the customer (no BI as a service option exists with this specific product).

Extensive tuning and experimentation was performed to make reports return data as per requirements, as at some point more time was spent on the reporting layer than on the database layer (and is still the case with ad hoc reports, as we explain below).

Jasper Reports has a limitation on ad-hoc reporting: it cannot make use of two date columns (StartDate and EndDate) from a table to filter data based on user provided Effective Date or date range, which is the traditional slowly changing dimension design –and there are dozens of SCD tables in the data warehouse. This limitation only applies to Ad-Hoc querying model but works for canned (operational and analytic) reports. PSL studied several alternatives and recommended a solution based on views on factless-fact tables which stores history per date by learner across dimension keys, which is not too demanding in terms of extra ETL work. Jasper has another product, Jaspersoft OLAP, that also overcomes this limitation and supports aggregate awareness to select aggregate tables or not depending on the query (experimentation is underway on this last point).

5.2.5 Data movement /ETL tools

On the data movement front, PSL also evaluated the Azure Data Factory service, a public cloud service allowing to populate SQL Server, Azure SQL DW and Azure storage. Transforms can be written in languages such as Hive/Pig/C#/T-SQL (Stored Procedure language for SQL Server); pipelines are composed of activities by scripting them in JSON code –there is no GUI. It is suitable for all kinds of data, not just structured data. Pricing is per data activity used.

On the plus side:

a. Elastic resources of Azure Data Factory will address scalability concerns with respect to the volume of data to move to the cloud and transform.

b. Flexible ETL and faster turn-around is also expected, even though debugging facilities appear to be limited.

On the negative side, some limits and constraints of the product would be hit in this project; for instance, the maximum number of pipelines is 100, maximum number of fields per database object is also 100.

Pricing follows the "pay as you go" model (see https://azure.microsoft.com/en-us/pricing/details/data-factory/) and is a function of activities being used. It depends on factors such as (i) the frequency of activities, (ii) place where the activities run (between cloud data stores or whether an on-premises store is involved, and the pricing is a fraction of a dollar per hour), (iii) whether a pipeline is active or not, and (iv) whether you are re-running an activity. As an example, if it takes 2 hours in a day to move data from on-premises SQL Server database to Azure blob storage, the total per month comes out to be $7.50 per month. Again, cost should generally compare favorably.

As transferring hundreds of terabytes over the internet is almost unfeasible (100 TB transfer over a dedicated 100 Mbps connection takes over 100 days), the solution would be to use Azure's Import/Export service. This service allows Azure customers to securely transfer large amounts of data to Azure blob storage by shipping hard disk drives to an Azure data center.

For now, the customer has decided to go with the standard alternative, namely, to use SSIS, deployed on Azure VMs (as opposed to on-premise). Some months into the project, as with many ETL tools, development productivity started becoming a serious obstacle, in line with common experience[18]. Productivity achievements from ETL tools is related to (i) the collection of features of the tool itself (which keeps improving with time), (ii) the skill of the person using the tool, (iii) the way in which the tool is used for the project at hand, and (iv) the amounts of tools and templates available to speed up the development process.

PSL tackled the last point and could demonstrate that over 75% of the tables in the warehouse could be produced with T-SQL stored procedure parameterized templates. Template parameters include schema names, table names, stored procedure names, field names, and values. Recent versions of SSIS come with predefined templates for many common tasks, as well as the ability to create a custom template for a complex script that you must create frequently. Once parameters in a T-SQL template have been replaced, the resulting stored procedure can be called from SSIS, which is therefore used as an orchestration engine and to implement the remaining transformations that cannot be templatized.

On the gains front:

- PSL found that 40% of the code could be reused, and that development times could be trimmed by about 33%.
- Also, there is an improvement in ETL performance, as T-SQL stored procedures execute natively on the database engine.

On the drawbacks side,

- Debugging and maintainability would be harder; in particular, re-start, check-pointing, and handling event errors are available at procedure-level, not line-level
- Portability suffers if the DBMS engine changes (e.g., if and when changing to Azure SQL DW)
- Logging and error handling would need to be implemented in procedures as well

The customer accepted the drawbacks in exchange for productivity and performance.

[18] ETL development costs typically amount to around 70% of the total cost of a warehousing project.

5.2.6 Additional PaaS services

The customer utilized the other PaaS services offered by the Azure cloud to meet a few other key requirements as described below.

- Azure Service Bus to decouple the source system database with the DW: The OLTP product, also to be deployed in the cloud, is the source application for the analytic component. The underlying OLTP database schema changes are more frequent than the Analytical DW product revisions. In addition, customer did not want tight coupling between DW and the source database, as there could be several consumers of the OLTP data in future –the DW is the single consumer for now. Changes happening in the source OLTP database are sent as an event message in the form of a JSON message to the Azure Service Bus. This message envelope has a product major version, minor version and patch version. This helped ETL to subscribe and consume only those messages which are relevant for ETL/ DW product version. The Service bus topics were partitioned per tenant to achieve parallel consumption of messages.

- DW backup and recovery: the customer plans to use Backup vault and the Recovery Services vault to back up the VMs periodically and restore in case of failures.

- Deployment tools: Customer has used Windows PowerShell scripts and workflows (runbooks), to build the code from the repository and create the deployment kit. This process is automated.

- Real-time resource monitoring: Customer plans to use Azure Diagnostics extensions to collect performance statistics on Service Bus worker roles, VMs and the OLTP application.

6 Product-to-Product comparison

As it can be seen from previous sections, most of the cloud service providers offer similar building blocks for data ingestion, processing, streaming, machine learning and visualizations. At the outset, all four have everything covered; however, there are minor feature differences in terms of implementation. This section describes these differences.

6.1.1 ETL

All cloud providers have ETL services to offer data flow from external sources into cloud storage. AWS has Data Pipeline and Kinesis, Azure has Data Factory and Stream Analytics, Google has Dataflow and IBM has Data Connect and Streaming Analytics. All of them provide basic ETL / data processing functionality; however, support to input/output sources differ. Since this support is also becoming available for additional sources every couple of months, we suggest the reader to go respective sites to find out the sources each product supports. One main difference is that Google is the only provider which doesn’t have two separate offering for traditional ETL and stream processing.

6.1.2 Machine Learning

While AWS has a solid set of products around machine learning, they lack in pre-trained learning models when compared to Azure. While AWS has good UI interface for ML, it lacks a managed lab notebook, which is a feature generally appreciated by data scientists. Azure also offers custom R models running over big data. Compared to AWS and Azure, Google’s Tensorflow has been getting a lot of attention recently and there will be many who will be keen to see Machine Learning come out of preview. While Google has a strong rich set of pre-trained APIs, it lacks BI dashboards and visualizations. On the other hand, IBM’s Watson Analytics works more like interrogating data by asking English questions, more useful for problems where pre-packaged solutions are not sufficiently available. In short, we can catalog it more as a data discovery tool while Azure ML would be more of development tool.

6.1.3 Cloud Function

Here only AWS Lambda service is ready for production, Azure functions are in preview and Google Cloud Functions are in closed alpha. Apart from that, there are some differences in them on supported languages, event sources, architecture etc. For detailed comparison, click here.

6.1.4 Cloud Data Warehouses

6.1.4.1 Scaling

In Amazon's Redshift, storage and compute units are grouped together as a node definition. So, while new clusters are being provisioned, the current cluster is available only in read mode. The time taken to complete the operation of cluster provisioning and data copying to new cluster could take a few hours to days in Redshift. In contrast to this, in Azure SQL Data Warehouse, the scaling of the clusters can happen in minutes as the scale out can be done for compute and storage units independently. Both Google's BigQuery and IBM's DashDB allows to scale and pay for compute and storage node independently.

6.1.4.2 Data sources

Both AWS Redshift and Azure SQL Data Warehouse have a mechanism to integrate with respective blob storage, Hadoop service and NoSQL data sources. To integrate with on premise database, it needs to be exported into a file and then imported to respective storage mechanism.

6.1.4.3 Client BI Tools

Redshift integrates with many popular BI tools, like Tableau. In addition, it also allows connecting using JDBC and ODBC drivers. Azure SQL Data Warehouse also supports integration with popular BI tools such as Tableau and Power BI.

Both Redshift and Azure SQL Data Warehouse look promising. Azure SQL Data Warehouse leads in some areas, such as the scalability and decoupling the store from compute. On the other hand, Redshift leads in security by enabling it to be hosted in a VPC.

Enterprises who have been using Microsoft SQL Server widely will naturally move their data warehouse to Azure SQL Data Warehouse as it is extension of SQL Server family of products and they will find developer knowledge and skills in house easily.

Goggle's big query is fully managed data warehouse where it requires minimum input and rest it manages in terms of number of nodes, indexes, periodic maintenance etc. Performance on BigQuery is generally better as it brings as many resources as needed to run the query versus other platforms where it is limited by number of CPUs customer is paying for.

In terms of overall adoption, Amazon Redshift is still leading the market as it can integrate well with other AWS services including DynamoDB, Amazon S3, Amazon Kinesis, AWS Data Pipeline and AWS Lambda. There are more number of vendors who have certified Amazon's Redshift data warehouse with their offerings to enable customers to continue to use the tools you do today.

6.1.5 Data Visualization

Azure PowerBI tool is a more mature visualization tool than AWS's recently released QuickSight. Google Data Studio 360 is also coming out of Beta. So, in this space, Azure PowerBI is well established. PowerBI integrates with many business systems and applications like Microsoft Dynamics, Salesforce, Google Analytics, and Microsoft Excel. Most of the service providers also provide integration with reporting tools like Tableau and QlikView, as these tools can connect to on premise data sources as well.

7 References

[1] https://globenewswire.com/news-release/2017/03/06/931924/0/en/New-Survey-Finds-Cloud-Analytics-Now-Mainstream-and-Riding-a-Wave-of-Governed-Self-Service.html

[2] https://blogs.endjin.com/2016/08/aws-vs-azure-vs-google-cloud-platform-analytics-big-data/

[3] http://www.kdnuggets.com/2015/04/cloud-machine-learning-amazon-ibm-watson-microsoft-azure.html

[4] http://stackoverflow.com/questions/40326085/compare-aws-lambda-azure-functions-and-google-cloud-function

[5] https://www.reddit.com/r/bigdata/comments/3jnam1/whats_your_preference_for_running_jobs_in_the_aws/cur518e/

[6] http://www.microsofttrends.com/2015/08/24/comparing-ibm-watson-analytics-with-azure-ml/

[7] https://ilyas-it83.github.io/CloudComparer/

[8] http://nathanmarz.com/blog/how-to-beat-the-cap-theorem.html

[9] https://www.linkedin.com/pulse/migration-data-warehouses-cloud-deepa-deshpande

[10] http://www.atscale.com/

[11] http://kylin.apache.org/

[12] Cubrick: Indexing Millions of Records per Second for Interactive Analytics, Pedreira et.al., Facebook, VLDB 2016, http://www.vldb.org/pvldb/vol9/p1305-pedreira.pdf

[13] Data Infrastructure at Flipkart, Sharad Agarwal, VLDB 2016, https://www.slideshare.net/sharad_ag/data-infrastructure-at-flipkart-vldb-2016

[14] Seth Gilbert and Nancy Lynch, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services", ACM SIGACT News, Volume 33 Issue 2 (2002), pg. 51–59.

[15] Best Practices in Data Management for Analytics Projects, https://www.persistent.com/data-management-best-practices/

[16] TPC-H: an ad-hoc, decision support benchmark, http://www.tpc.org/tpch/

8 Appendix 1 – Performance and scalability

Performance
Performance is a complex, derived requirement which is a function of several query and data requirements (query response times, query types, user scales and data volumes). It generally refers to sustained query response time during a time interval, on a well-specified workload under control where type of query mix, user scales and data volumes are precisely defined.

An example of a way to measure performance is the Composite Query per Hour, or QphH@size, as defined by the TPC-H performance benchmark [16], a well-defined analytic workload. The number of concurrent query streams, a complexity mix of queries including sequential scans, aggregation, multi-table joins and sorting, the database volume size and the query response times are all part of the computation of QphH@size, which boils down to the number of completed queries per hour at a database scale factor (volume). Even though our performance scale is applied informally throughout this document, we should define it precisely. On a 10 TB TPC-H database, using the TPC-H analytic workload, our performance level definitions are:

- *Very High* performance refers to a QphH@size > 10 million. Only an MPP database is capable of this result today on the TPC-H.

- *High* performance refers to a QphH@size between 1 and 10 million. This is one order of magnitude below. This is within the reach of SMP databases: Microsoft SQL Server is capable of high performance on this benchmark, with a QphH@size result above 1.1 million.

- *Medium* performance refers to a QphH@size between 100,000 and 1 million. This is yet an order of magnitude below.

- *Average* performance and below will refer to a QphH@size below 100,000. This corresponds to two orders of magnitude below what we call high performance, which corresponds roughly with the performance associated with Hadoop systems today.

Scalability
Scalability is a critical requirement to overcome performance limits (response time and/or throughput) by adding computing resources. Some benchmarks include the notion of resources needed to get performance: TPC-H, for instance, defines a price-performance metric ($/QphH/@Size) where $ is the total system price[19]. This made sense in on-premise environments hosting analytics solutions at the time TPC-H was defined. In the cloud computing realm, cost, although relevant, no longer is the most important issue: design of a scalable design has become most desirable and challenging system property, to ensure that the system capacity can be augmented by adding additional infrastructure resources, to either support large end user scales and/or to respond to sudden load fluctuations on a cloud-based service due to demand surges or troughs from the end users. Three notions of scalability are commonly being talked about in this context: *scale-up, scale out and elasticity*.

Scaling up, or vertical scaling, generally refers to adding resources to a single node in a system, typically involving the addition of processors and/or memory to a single computer. Such vertical scaling of existing systems also enables them to use virtualization technology more effectively, as it provides more resources for the hosted set of operating system and application modules to share.

Scaling out, or horizontal scaling, means adding and linking together other lower-performance machines to collectively do the work of a much more advanced one. With these types of distributed setups, it's easy to handle a larger workload by running data through different system trajectories.

[19] TPC-H specifies how to price resources and how to express the total system price.

Elasticity is the ability to fit the resources needed to cope with loads dynamically, usually in relation to scale out, so that when the load increases you scale by adding more resources and when demand wanes you shrink back and remove unneeded resources. Elasticity is mostly important in cloud environments where you pay-per-use and don't want to pay for resources you do not currently need on the one hand, and want to meet rising demand when needed on the other hand.

There are a variety of benefits and disadvantages to each approach. Scaling up can be expensive, and ultimately, some experts argue that it's not viable because of the limits to individual hardware pieces on the market. However, it does make it easier to control a system, and to provide for certain data quality issues. Scale out is very popular, as it is the technology behind tools like Apache Hadoop. Here, central data handling software systems administrate huge clusters of hardware pieces, for systems that are often very versatile and capable.

At the infrastructure layer, elasticity is easier to handle, and Hadoop-as-a-Service offerings providing elastic, auto-scaling clusters are becoming common (although not pervasive yet: this is something to watch-out for). At the database layer, NoSQL databases pioneered the introduction of the elasticity property in their systems: these DBMSs have been designed so that they can be elastic or can be dynamically provisioned in the presence of load fluctuations. As we argue in section 4.3, NoSQL databases are meant for operational systems with simple analytics requirements, and solve the elasticity problem by providing write atomicity only at a record level (i.e., for a single key, in key-value pair NoSQL systems)[20]. On the other hand, traditional DBMS systems handle more general transactions and more complex queries, and are in general intended for an enterprise infrastructure that is statically provisioned. Thus, elasticity for general DBMSs is a much more complex problem to solve. Even MPP databases, which are optimized for analytics workloads, assume static cluster hardware configurations. This is however starting to change. Some cloud databases are starting to provide for elasticity by separating storage and compute nodes, detect tenant contention and scale up and down tenants within resource limits. Both Azure SQL data warehouse and Snowflake are examples of this new trend.

9 Appendix 2 – IaaS management in Microsoft Azure

This appendix complements our running example in section 5 by providing more detail on the management of the infrastructure layer and the tools available to IT administrators to manage SQL server on virtual machines in Azure.

At the infrastructure level (VMs, storage, and networks), IT administrators use IaaS services as follows:

- Azure IaaS VM service allows to control the size of the VMs; parameters include the number of cores, size of RAM, storage capacity, disk throughput and network bandwidth. Each size determines an hourly price. As mentioned below, when SQL Server runs on these VMs, right-sizing and properly configuring them for performance must be well understood. The VM service provides automated features to dramatically simplify patching, backup, and high availability, as well as monitoring to diagnose problems in VMs and get alert notifications on metric values or events.

- The VM service provides a means to detect health of virtual machines running on the platform and to perform auto-recovery of those virtual machines in case they fail. Microsoft provides an availability SLA of 99.9% for single instance Virtual Machines[21]. This SLA does not cover processes (such as SQL Server) running on the VM and requires customers to host at least two VM instances in an *availability set*[22]. More on this below.

- At the Azure IaaS storage system, an unfamiliar aspect in cloud deployments is that there is no access to the underlying hardware. However, IO activity can be monitored and analyzed using analytics when enabled at the account level –in this case, blob operations are persisted and metrics can be defined and aggregated over time to understand and benchmark the storage system.

[20] Evidence is emerging that indicates that in many application scenarios this is not enough. This problem was recognized recently by senior architects from Amazon and Google (which has led to systems such as MegaStore, at the heart of Google's App Engine, that provide transactional guarantees on entity groups that represent fine-grained, application-defined partitions).
[21] This is a downtime of 43 minutes a month, approximately (MSFT even talks about guaranteed downtime of 15 minutes!)
[22] Defined as 2 or more VMs deployed across different Fault Domains to avoid a single point of failure –and avoiding guaranteed downtime.

- Finally, customers can isolate logically the Azure cloud resources of their subscription through Azure virtual networks (VNets) –these resources include IaaS VMs and PaaS role instances (windows servers in web or worker roles). On VNets, customers can fully control the IP address blocks, DNS settings, security policies, and route tables, and connect the virtual network to their on-premises network, allowing to build hybrid cloud applications.

On the SQL Server management side, IT administrators must directly manage the following areas:

- Connectivity: public over the internet, within virtual network, inside VM only.

- Authentication and security: managing accounts, using network security groups, consider using Azure virtual networks, enable encrypted connections, restricting access to VMs to certain IP addresses or subnets, etc.

- Backups: Reasons for enabling backup no longer include protection against media or hardware failures, as the IaaS VM service provides this already. Users can disable backups, or enable them to provide protection against user errors, or for archival purposes or regulatory reasons. In addition, they can configure retention period and backup encryption key if desired, as well as location: Azure blobs, a disk or tape device, or on an on-premises instance.

- High availability: beyond hosting two or more VMs in an availability set, administrators should understand the tradeoffs of the existing options (Availability Groups and Failover Cluster Instances) in SQL Server and configure one of these, which allows to achieve over 99.99% database availability.

- Disaster recovery: Understand the tradeoffs between existing options (cross-region availability groups, database mirroring, backup and restore to Azure Blob Service storage), and configure and manage them.

- Patching schedules and maximum time allotted for patching for SQL Server and Windows,

- Best practices for best performance from SQL Server running on an Azure VM: right-sizing VMs, configuring storage and following guidelines to improve IO (when to cache, to compress, etc).

- Encryption: decide which encryption features to use (file level, column level and backup encryption), all managed through cryptographic keys. A companion service, the Azure Key Vault service, is designed to improve the security and management of these keys in a secure and highly available location.

- Finally, as in an on-premises deployment, IT administrators must use monitoring tools to assess the availability, performance, security and functionality of the database. Traditional SQL Server tools, logs, SDKs and 3rd party tools allow to monitor performance metrics, query performance, and system activity through extended events.

10 Appendix 3 – Main cloud service provider competitors

10.1 Google Cloud Platform

10.1.1 Overview

Google Cloud Platform (GCP) is a cloud computing service by Google that offers hosting on the same supporting infrastructure that Google uses internally for end-user products like Google Search and YouTube. It is a part of a suite of enterprise services from Google Cloud and provides a set of modular cloud-based services with a host of development tools. It provides developer products to build a range of programs from simple websites to complex applications.

Components of Google Cloud Platform

- **Cloud Dataflow** - is a fully-managed data processing service, supporting both stream and batch execution of pipelines.
- **Cloud Functions** - A server less platform for building event-based micro-services. Currently in alpha testing FaaS service allowing functions to be triggered by events without developer resource management.
- **Google Cloud Pub/Sub** - brings the scalability, flexibility, and reliability of enterprise message-oriented middleware to the cloud by providing many-to-many, asynchronous messaging that decouples senders and receivers.
- **StackDriver Logging** - allows you to store, search, analyze, monitor, and alert on log data and events from Google Cloud Platform and Amazon Web Services (AWS).
- **BigQuery** – PaaS service providing columnar database.
- **Cloud Sql** - is a fully-managed database service that uses relational MySQL databases on Google Cloud Platform.
- **BigTable** - is a compressed, high performance, and proprietary data storage system built on Google File System. It is Google's NoSQL Big Data database service.
- **Google Cloud Storage** – is an IaaS and RESTful web service for online file storage and accessing data on Google's infrastructure.
- **Cloud Storage** - Google Cloud Storage is unified object storage for developers and enterprises, from live data serving to data analytics/Machine Learning to data archiving.
- **Cloud Machine Learning** - is a managed service allows to build machine learning models easily, that work on any type of data, of any size.
- **Cloud Dataproc** - is a cloud-based managed Spark and Hadoop service offered on Google Cloud Platform. It is used as a PaaS.
- **Google Compute Engine** – IaaS service providing virtual machines.
- **Google Cloud Datastore** – highly-scalable NoSQL database service for transactional applications. It has the capabilities such as ACID transactions, SQL-like queries, indexes.
- **Google App Engine** – PaaS service for directly hosting applications
- **Cloud IAM** - provides authorization and authentication of different services.
- **Cloud Datalab**- is a powerful interactive tool used to explore, analyze and visualize data.
- **Cloud key management service**- is used to protect secrets and other sensitive data which you need to store in Google Cloud Platform.
- **Data Studio 360**: Google Data Studio (beta) turns data into informative dashboards and reports that are easy to read, easy to share, and fully customizable
- **Cloud Deployment Manager:** A cloud infrastructure management service that automates the creation and management of Google Cloud Platform resources.

10.1.2 GCP Components in detail

10.1.2.1 Cloud Dataflow

Cloud dataflow is a fully-managed data processing service, supporting both stream and batch execution of pipelines. It is used to transfer data from multiple source like Avro files, BigQuery tables, Bigtable, Datastore, Pub/Sub, Text files etc. to multiple sinks. Developers can create a custom source and sink by extending the Dataflow SDK's abstract Source subclasses such as BoundedSource or UnboundedSource and by extending the abstract Sinkbase class. It provides features like auto scaling, easy to integrate with multiple tools Cloud Storage, Cloud Pub/Sub, Cloud Datastore, Cloud Bigtable and BigQuery.

References
https://cloud.google.com/dataflow/model/custom-io-java
https://cloud.google.com/dataflow/

10.1.2.2 Cloud Functions

Google Cloud Functions is a lightweight, event-based, asynchronous compute solution that allows users to create small, single-purpose functions that respond to cloud events without the need to manage a server or a runtime environment. Events from Google Cloud Storage and Google Cloud Pub/Sub can trigger Cloud Functions asynchronously, or you can use HTTP invocation for synchronous execution. It runs in a fully-managed, server-less environment where GCP handles the servers, operating systems and runtime environments, and developers focus on building solutions. It provides event-based services, Cloud Pub/Sub Triggers, Cloud Storage Triggers, HTTPS Invocation, Logging & Monitoring and GitHub/Bitbucket Support. It uses **Node.js** to write code and deploy using **gcloud**.

References
https://cloud.google.com/functions/

10.1.2.3 Cloud Pub/Sub

Cloud Pub/Sub is a fully-managed real-time messaging service. It allows developers to send and receive messages between independent applications. Cloud Pub/Sub's flexibility can be leveraged to decouple systems and components hosted on Google Cloud Platform or elsewhere on the Internet. By building on the same technology Google uses, Cloud Pub/Sub is designed to provide "at least once" delivery at low latency with on-demand scalability to 1 million messages per second (and beyond). It encrypts all messages on the wire and at rest provides data security and protection.

References
https://cloud.google.com/pubsub/

10.1.2.4 StackDriver Logging

StackDriver Logging is a GCP component. It allows to store, search, analyze, monitor, and alert on log data and events from GCP and Amazon Web Services (AWS). It provides support to integrate with Cloud Pub/Sub, Splunk & Log entries, Google Cloud Storage and google BigQuery; allows to browse log data and create metrics from log data. StackDriver Logs Viewer, APIs, and the gCloud CLI can be used to access Audit Logs that capture all the admin and data access events within GCP.

References
https://cloud.google.com/logging/

10.1.2.5 BigQuery

BigQuery is Google's fully managed, petabyte scale, low cost enterprise data warehouse for large-scale data analytics. BigQuery is the public implementation of Dremel, Google's internal columnar store, massively parallel, scalable query service for read-only datasets of nested data. BigQuery is server-less. There is no infrastructure to manage and no database administrator is needed, so developers can focus on analyzing data to find meaningful insights using familiar SQL. Its engine can scan 1 TB data in seconds and 1 PB in minutes by parallelizing queries and running them on tens of thousands of servers without using indexes. Developers can load data from Google Cloud Storage or Google Cloud Datastore, or stream it into BigQuery to enable real-time data analysis. BigQuery can easily scale databases from GBs to PBs. It automatically encrypts and replicates customer's data to ensure security, availability and durability. BigQuery can further protect data with strong role-based ACLs that GCP configures and controls using the customer's Google Cloud Identity & Access Management system.

References
https://cloud.google.com/bigquery/
https://cloud.google.com/bigquery/docs/reference/legacy-sql

10.1.2.6 Cloud SQL

Google Cloud SQL is a fully-managed database service that makes it easy to set up, maintain, manage, and administer relational MySQL databases in the cloud. Google Cloud SQL Second Generation offers high performance, scalability, and convenience with up to 10TB of storage capacity, 25,000 IOPS, and 208GB of RAM per instance. Hosted on GCP, Cloud SQL provides a database infrastructure for applications running anywhere. It automates all customer's backups, replication, patches, and updates - while ensuring greater than 99.95% availability, anywhere in the world. It encrypts the customer's data when store on Google's internal networks and in database tables, temporary files, and backups. Every Cloud SQL instance includes a network firewall, allowing customers to control network access to their database instance by granting access.

References
https://cloud.google.com/sql/

10.1.2.7 Cloud Bigtable

Cloud Bigtable is Google's NoSQL Big Data database service. It is the same database that powers many core Google services, including Search, Analytics, Maps, and Gmail. Bigtable is designed to handle massive workloads at consistent low latency and high throughput, so it's a great choice for both operational and analytical applications, including IoT, user analytics, and financial data analysis. Bigtable provisions and scales to hundreds of petabytes automatically, and can smoothly handle millions of operations per second.

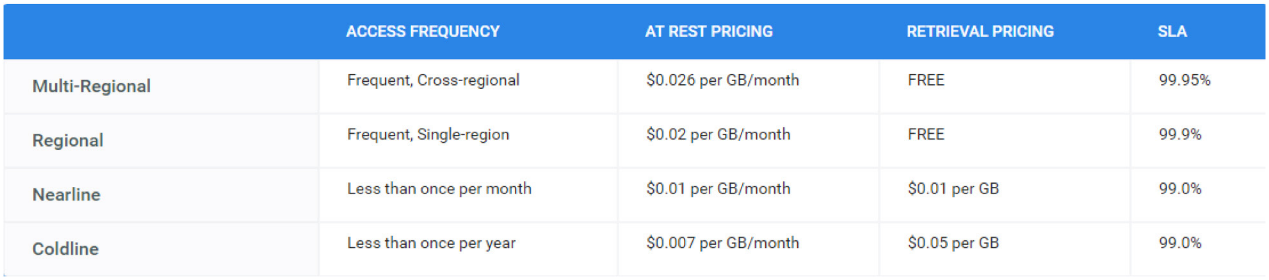
Bigtable can integrate with Big Data tools like Hadoop, Hbase as well as GCP products like Cloud Dataflow and Dataproc.

References
https://cloud.google.com/bigtable/

10.1.2.8 Cloud Storage

Google Cloud Storage is unified object storage for developers and enterprises, from live data serving to data analytics/Machine Learning to data archiving. It allows to store data world-wide and retrieve of any amount of data at any time. It can be used for multiple scenarios, including serving website content, storing data for archival and disaster recovery, or distributing large data objects to users via direct download. It provides four storage classes listed below; all storage classes offer the same throughput, low latency (time to first byte, typically tens of milliseconds), and high durability. The classes differ by their availability, minimum storage durations, and pricing for storage and access.

1. Multi-Regional Storage
2. Regional Storage
3. Nearline Storage
4. Coldline Storage



	ACCESS FREQUENCY	AT REST PRICING	RETRIEVAL PRICING	SLA
Multi-Regional	Frequent, Cross-regional	$0.026 per GB/month	FREE	99.95%
Regional	Frequent, Single-region	$0.02 per GB/month	FREE	99.9%
Nearline	Less than once per month	$0.01 per GB/month	$0.01 per GB	99.0%
Coldline	Less than once per year	$0.007 per GB/month	$0.05 per GB	99.0%

References
https://cloud.google.com/storage/

10.1.2.9 Cloud Machine Learning

Google Cloud Machine Learning (ML) Platform provides modern machine learning services, with pre-trained models and a service to generate the developer's own tailored models. Developers can create their models on TensorFlow, Google's second generation machine learning system, released as open source software in late 2015. Cloud ML has better training performance and increased accuracy compared to other large scale deep learning systems. It can support thousands of users and TBs of data, and can be deployed to run on multiple CPUs and GPUs. The services are fast, scalable and easy to use. Major Google applications use Cloud ML, including Photos (image search), the Google app (voice search), Translate, and Inbox (Smart Reply). The platform is now available as a cloud service to bring unmatched scale and speed to the customer's business applications. It is portable, fully managed, and integrated with other Google Cloud Data platform products such as Google Cloud Storage, Google Cloud Dataflow, and Google Cloud Datalab so you can easily train your models.

References
https://cloud.google.com/products/machine-learning/

10.1.2.10 Cloud Dataproc

Google Cloud Dataproc is a managed Spark and Hadoop services (Apache Hadoop, Apache Spark, Apache Pig, and Apache Hive). It is fast, easy to use, and easily process big datasets at low cost. Users can control their costs by quickly creating managed clusters of any size and turning them off when done. Cloud Dataproc integrates across GCP products, giving customers a powerful and complete data processing platform. It automates cluster management, resizes the clusters and integrates with Cloud Storage, BigQuery, Bigtable, Stackdriver Logging, and Stackdriver Monitoring, giving developers a complete and robust data platform.

References
https://cloud.google.com/dataproc/

10.1.2.11 Compute Engine

Google Compute Engine is a virtual machine running in Google's innovative data centers and worldwide fiber network. Compute Engine's tooling and workflow support enable scaling from single instances to global, load-balanced cloud computing. Its VMs boot quickly, come with persistent disk storage, and deliver consistent performance. Google's virtual servers are available in many configurations including predefined sizes or the option to create Custom Machine Types optimized for your specific needs. Flexible pricing and automatic sustained use discounts make Compute Engine the leader in price/performance. Google bills in **minute-level increments** (with a 10-minute minimum charge), so customers only pay for the compute time they use. You can run OS like Debian, CentOS, CoreOS, SUSE, Ubuntu, Red Hat, FreeBSD, or Windows 2008 R2 and 2012 R2.]

References
https://cloud.google.com/compute/

10.1.2.12 Cloud Datalab

Cloud Datalab is a powerful interactive tool created to explore, analyze and visualize data with a single click on GCP. It runs locally and optionally on Google Compute Engine and connects to multiple cloud services easily, so developers can focus on exploring their data. It is built on Jupyter (formerly IPython), which boasts a thriving ecosystem of modules and a robust knowledge base. It enables analysis of data on Google BigQuery, Google Compute Engine, and Google Cloud Storage using Python, SQL, and JavaScript (for BigQuery user-defined functions). It supports TensorFlow-based deep ML models in addition to scikit-learn.

References
https://cloud.google.com/datalab/

10.1.2.13 Cloud IAM

Google Cloud Identity & Access Management **(IAM)** lets users manage access control by defining who (members) has what access (role) for which resource. With IAM users can grant more granular access to specific GCP resources and prevent unwanted access to other resources. IAM lets customers adopt the security principle of least privilege, so they grant only the necessary access to their resources.

References

https://cloud.google.com/iam/

10.1.2.14 Data Studio 360:

Google Data Studio (beta) turns data into informative dashboards and reports that are easy to read, easy to share, and fully customizable. Dashboarding allows to tell great data stories to support better business decisions.

Using Google Data Studio allows to create unlimited Data Studio custom live, interactive reports and dashboards with full editing and sharing

References
https://www.google.com/analytics/data-studio/

10.1.2.15 Google Cloud Deployment Manager:

Google Cloud Deployment Manager is an infrastructure management service that automates the creation and management of Google Cloud Platform resources. Using Deployment Manager, developers can create flexible templates that deploy a variety of Cloud Platform services, such as Google Cloud Storage, Google Compute Engine, and Google Cloud SQL. Google Cloud Deployment Manager allows to specify all the resources needed for an application in a declarative format using YAML.

References
https://cloud.google.com/deployment-manager/

10.2 Amazon Web Services

10.2.1 Overview

Amazon Web Services (AWS) offers a suite of cloud-computing services that make up an on-demand computing platform. AWS has more than 70 services, spanning a wide range, including compute, storage, networking, database, analytics, application services, deployment, management, mobile, developer tools and tools for the Internet of things. The AWS Cloud operates 42 Availability Zones within 16 geographic Regions around the world.

Components of Amazon Web Services

- **Amazon Elastic Compute Cloud (EC2):** An IaaS service providing virtual servers controllable by an API, based on the Xen hypervisor.

- **Amazon Elastic MapReduce (EMR):** Provides a PaaS service delivering Hadoop for running MapReduce queries framework running on the web-scale infrastructure of EC2 and Amazon S3.

- **Amazon Machine Learning:** A service that assists developers of all skill levels to use machine learning technology.

- **Amazon Kinesis:** A cloud-based service for real-time data processing over large, distributed data streams. It streams data in real time with the ability to process thousands of data streams on a per-second basis.

- **Amazon QuickSight:** A business analytics service that provides visualizations and ad-hoc analysis by connecting to AWS or non-AWS data sources.

- **Amazon Lambda (LAMBDA):** Runs code in response to AWS internal or external events such as http requests transparently providing the resource required

- **Amazon Simple Storage Service (S3):** Provides scalable object storage accessible from a Web Service interface.

- **Amazon Glacier:** Provides long-term storage options (compared to S3).Intended for archiving data.

- **Amazon Redshift:** Provides petabyte-scale data warehousing with column-based storage and multi-node compute.

- **Amazon Relational Database Service (RDS):** Provides scalable database servers with MySQL, Oracle, SQL Server, and PostgreSQL support.

- **Amazon DynamoDB:** Provides a scalable, low-latency NoSQL online Database Service backed by SSDs.

- **AWS Data Pipeline:** Provides reliable service for data transfer between different AWS compute and storage services (e.g., Amazon S3, Amazon RDS, Amazon DynamoDB and Amazon EMR).

- **Amazon Identity and Access Management (IAM):** An implicit service, the authentication infrastructure used to authenticate access to the various services.

- **AWS Key Management Service (KMS):** A managed service to create and control encryption keys.

- **Amazon Aurora:** Provides a MySQL-compatible relational database engine that has been created specifically for the AWS infrastructure that claims faster speeds and lower costs that are realized in larger databases.

- **Amazon QuickSight:** An analytics service that makes it easy to build visualizations, perform ad-hoc analysis, and quickly get business insights from data.

- **AWS CloudFormation:** A service to create and manage a collection of related AWS resources, provisioning and updating them in an orderly and predictable fashion.

10.2.2 AWS Components in detail

10.2.2.1 AWS Data Pipeline

AWS Data Pipeline is a web service that helps to reliably process and move data between different AWS compute and storage services, as well as on premise data sources, at specified intervals. With AWS Data Pipeline, we can regularly access data where it's stored, transform and process it at scale, and efficiently transfer the results to AWS services such as Amazon S3, Amazon RDS, Amazon Dynamo DB, and Amazon EMR. AWS Data pipeline is a reliable, easy to use, flexible, scalable and transparent web service for data process and data transfer.

References
https://aws.amazon.com/datapipeline/
https://aws.amazon.com/blogs/aws/category/aws-data-pipeline/

10.2.2.2 AWS Lambda

AWS Lambda is a compute service that lets users run code without provisioning or managing servers. AWS Lambda executes code only when needed and scales automatically, from a few requests per day to thousands per second. AWS Lambda can be used to run a code in response to events, such as changes to data in an Amazon S3 bucket or an Amazon Dynamo DB table; response to HTTP requests using Amazon API Gateway; or invoke a code using API calls made using AWS SDKs.

Lambda performs operational and administrative activities for users, including capacity provisioning, scaling, high availability, monitoring fleet health, applying security patches, deploying the code, running a web service front end, and monitoring and logging the user's functions. Supported runtimes include Node.js, Python, Java and C# through .NET Core.

References
http://docs.aws.amazon.com/lambda/latest/dg/welcome.html

10.2.2.3 Amazon Redshift

Amazon Redshift is a fast and powerful, fully managed, petabyte-scale data warehouse service in the cloud. Amazon Redshift significantly lowers the cost of a data warehouse, also makes it easy to analyze large amounts of data very quickly. AWS Redshift provides different features such as specially optimized for data warehouse, Petabyte scale, automated backups, encryption, network isolation and fault tolerant.

References
https://aws.amazon.com/redshift/
http://docs.aws.amazon.com/redshift/latest/mgmt/overview.html
https://en.wikipedia.org/wiki/Amazon_Redshift

10.2.2.4 Amazon RDS

Amazon Relational Database Service (or Amazon RDS) is a distributed relational database service by Amazon Web Services. It is a web service running in the cloud designed to simplify the setup, operation, and scaling of a relational database for use in applications. Complex administration processes like patching the database software, backing up databases and enabling point-in-time recovery are managed automatically. Scaling storage and compute resources can be performed by a single API call. Amazon RDS provides six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, and Microsoft SQL Server.

References
https://aws.amazon.com/rds/
https://en.wikipedia.org/wiki/Amazon_Relational_Database_Service

10.2.2.5 Amazon DynamoDB

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed cloud database and supports both document and key-value store models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad tech, IoT (Internet of things) and many other applications.

References
https://aws.amazon.com/dynamodb/
https://en.wikipedia.org/wiki/Amazon_DynamoDB

10.2.2.6 Amazon S3

Amazon Simple Storage Service (Amazon S3) is object storage with a simple web service interface to store and retrieve any amount of data from anywhere on the web. It is designed to deliver 99.999999999% durability, and scale past trillions of objects worldwide. S3 used as primary storage for cloud-native applications; as a bulk repository, or data lake, for analytics; as a target for backup & recovery and disaster recovery; and for supporting server less computing. It provides features such as Backup and Archiving, Content Storage and Distribution, Big Data Analytics, Static Website Hosting, Cloud-native Application Data and Disaster Recovery.

References
https://aws.amazon.com/s3/?hp=tile&so-exp=below
https://en.wikipedia.org/wiki/Amazon_S3
https://aws.amazon.com/s3/details/

10.2.2.7 Amazon Glacier

Amazon Glacier is a secure, durable, and extremely low-cost cloud storage service for data archiving and long-term backup. It is low cost, secure, durable, flexible and integrated cloud storage service. It can reliably store large or small amounts of data for as little as $0.004 per gigabyte per month, a significant savings compared to on-premises solutions.

References
https://aws.amazon.com/glacier/

10.2.2.8 Amazon ML

Amazon Machine Learning provides visualization tools and wizards that guides through the process of creating machine learning (ML) models without having to learn complex ML algorithms and technology. Once models are ready, Amazon Machine Learning makes it easy to obtain predictions for an application using simple APIs, without having to implement custom prediction generation code, or manage any infrastructure. Amazon Machine Learning is highly scalable and can generate billions of predictions daily, and serve those predictions in real-time and at high throughput. With Amazon Machine Learning there is no upfront hardware or software investment: the pricing model is “pay as you go”, so customers can start small and scale as the application grows.

References

https://aws.amazon.com/machine-learning/

https://aws.amazon.com/machine-learning/details/

10.2.2.9 Amazon EMR

Amazon EMR provides a managed Hadoop framework that makes it easy, fast, and cost-effective to process vast amounts of data across dynamically scalable Amazon EC2 instances. AWS can also run other popular distributed frameworks such as Apache Spark, HBase, Presto, and Flink in Amazon EMR, and interact with data in other AWS data stores such as Amazon S3 and Amazon DynamoDB.

Amazon EMR securely and reliably handles a broad set of big data use cases, including log analysis, web indexing, data transformations (ETL), machine learning, financial analysis, scientific simulation, and bioinformatics.

References
https://aws.amazon.com/emr/
https://aws.amazon.com/emr/details/

10.2.2.10 Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows to obtain and configure capacity with minimal friction. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing quickly scale capacity, both up and down, as computing requirements change.

References
https://aws.amazon.com/ec2/
https://aws.amazon.com/ec2/details/

10.2.2.11 AWS Identity & Access Management

AWS Identity and Access Management (IAM) is a web service that helps to securely control access to AWS resources for users. IAM used to control who can use AWS resources (authentication) and what resources they can use and in what ways (authorization). AWS IAM provides features such as Shared access to AWS account, Granular permissions, Secure access to AWS resources for applications that run on Amazon EC2,Identity federation, Identity information for assurance, PCI DSS Compliance, Integrated with many AWS services, Eventually Consistent.

References
http://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html

10.2.2.12 Amazon QuickSight:

Amazon QuickSight is a fast, cloud-powered business analytics service that makes it easy to build visualizations, perform ad-hoc analysis, and quickly get business insights from data. Using QuickSight cloud-based service you can easily connect to data, perform advanced analysis, and create stunning visualizations and rich dashboards that can be accessed from any browser or mobile device.

References
https://quicksight.aws/

10.2.2.13 AWS CloudFormation:

AWS CloudFormation gives developers and systems administrators an easy way to create and manage a collection of related AWS resources, provisioning and updating them in an orderly and predictable fashion. Developers can use AWS CloudFormation's sample templates or create own templates to describe the AWS resources, and any associated dependencies or runtime parameters, required to run an application.

References
https://aws.amazon.com/cloudformation/

10.3 Microsoft Azure Cloud

10.3.1 Overview

Microsoft Azure is a cloud computing service created by Microsoft for building, deploying, and managing applications and services through a global network of Microsoft-managed data centers. It provides software as a service, platform as a service and infrastructure as a service and supports many different programming languages, tools and frameworks, including both Microsoft-specific and third-party software and systems.

Components of Azure Cloud

- **Compute Services:** Infrastructure as a service (IaaS) allowing users to launch general-purpose Microsoft Windows and Linux virtual machines.
- **Azure Machine Learning:** Microsoft Azure Machine Learning Studio is a collaborative, drag-and-drop tool you can use to build, test, and deploy predictive analytics solutions on your data.
- **HDInsight:** HDInsight is the only fully-managed cloud Hadoop offering that provides optimized open source analytic clusters.
- **Azure SQL database:** SQL Database is a relational database service in the Microsoft cloud based on the market-leading Microsoft SQL Server engine and capable of handling mission-critical workloads.
- **Azure DocumentDB:** Document DB is a fully managed NoSQL database service built for fast and predictable performance, high availability, elastic scaling, and Schema-free NoSQL database.
- **Azure Stream Analytics:** Azure Stream Analytics is a fully managed, cost effective real-time event processing engine which makes computation on data streaming from devices, sensors, social media and application etc.
- **Azure Data Factory:** Data Factory is a cloud-based data integration service that orchestrates and automates the movement and transformation of data.
- **Azure Active Directory:** Azure Active Directory (Azure AD) is Microsoft’s multi-tenant cloud based directory and identity management service.
- **Azure Storage Service:** Provide storages options such as Blob storage, Table storage, Queue storage and File storage.
- **Azure Functions:** Azure Functions is a solution for easily running small pieces of code, or “functions,” in the cloud.
- **Azure SQL Data warehouse:** Azure SQL Data Warehouse is a cloud-based, scale-out database capable of processing massive volumes of data, both relational and non-relational.
- **Log Analytics:** Log Analytics is a service in Operations Management Suite (OMS) that helps you collect and analyze data generated by resources in your cloud and on-premises environments.
- **Azure Data Lake Store:** Azure Data Lake Store is an enterprise-wide hyper-scale repository for big data analytic workloads.
- **PowerBI:** Business analytics service provided by Microsoft that provides interactive visualizations with self-service business intelligence capabilities to create reports and dashboards.

10.3.2 Azure Components in detail

10.3.2.1 Data Factory

The azure data factory service allows to create the data pipelines that moves and transform the data, and then run the pipelines on a specified schedule (hourly, daily, weekly, etc.). Basically, its purpose is to ingest the data from various on-premises and cloud data sources to Azure.

Features

- Easily move data movement from different azure cloud based storage and various Databases input sources.
- We can also move data from different file system's (e.g. HDFS, Amazon S3 and FTP).
- Ability to transform data using activities (e.g. Hive, pig and map-reduce etc.)
- Visualize, manage and monitor entire network pipeline to identify issue and tracks.

10.3.2.2 Log Analytics

Log analytics helps to collect and analyses the log data generated by various cloud resources or on-premises resources (i.e. this service collects log data). All collected logs will get stored in Operation Management Suite (OMS) repository.

Features

- Ability to gather the different types of log information such as text file log on windows, Windows Event logs Windows Performance, Linux Performance counters, IIS logs, Syslog, Azure Storage etc.
- Advanced searching on gathered log information with the help of supported keywords (e.g. error, computer name and timeout etc.) and search query language(e.g. system error | sort ManagementGroupName).
- Ability to schedule the alerts based on search criteria.
- The OMS UI allows to create dashboards/insights of log data for better visualization.
- Can export log data to POWER BI tool.

10.3.2.3 Document DB (NoSQL database)

Document DB is a fully managed NoSQL database as a service (DBaas).The Document DB is NoSQL document based database which provides fast and predictable performance, high availability, elastic scaling, and global distribution.

Features

- Support of SQL syntax to make query over the multiple documents.
- Document DB creates the index automatically on all the documents.
- This service has support of JavaScript language, which allows user to write transactional logic, triggers, user-defined functions and stored procedure etc.
- Can easily integrate with HDInsight service (Hadoop service).
- It provides data access control over the multiple databases as well as resources with the help of master key, read-only key and resource key etc.
- Automatically replicate all your data across region world-wide.

10.3.2.4 Azure Machine Learning

Azure ML's core feature is to make predictions on your data. This service provides studio which comes with lots of drag-and-drop tool such as projects, web services, datasets, and notebooks you can use these tools build, test, and deploy predictive analytics solutions.

Features

- Ability to load the dataset from sources such as delimited text files, SQL server, Azure storage services and Hive tables etc.
- Azure ML has supports classification, regression, text analytics, clustering etc. (e.g. K-means, Linear regression, SVM etc.) models.
- Ability to visualize the data using in the form of bar, histogram and bar plots etc.
- Ability to prepare/transform the data to make predications (such as feature selection, normalization, filter and clean missing data)
- Allow to create web service of well-designed model and use it in some another application.

10.3.2.5 Azure Resource Manager

The Azure RM is specially designed to monitor, deploy and maintain the cloud resources. The resources might be windows machine, Linux machine, storage services etc. Azure RM helps to deploy the solution as per your required machine specification and resources for your use case (i.e. helps to deploy your solution).

Features

- Repeatedly deploy your solution.
- Can group the most relevant resources which helps organization to view the billing.
- Can set RBAC to services in your resource group.

10.3.2.6 Azure Function

The Azure function is small piece of code which runs in the cloud and can be used to carry out time based processing of data (e.g. cleans up database table after every 15 minutes), develop REST API and trigger based event processing, image processing, etc.

Features

- Developer can code in variety of languages.
- Developer can add his/her own dependencies (e.g. jar files).
- Easy integration with azure other services: these services can trigger your functions and can be used as input or output for your functions.
- Function can be triggered on the occurrences of different event such as blob trigger, event hub trigger, timer trigger, HTTP trigger etc.

10.3.2.7 Azure SQL Data Warehouse:

The SQL data warehouse is a scale-out database capable of processing massive volumes of data, both relational and non-relational. It's built on a massively parallel processing **(MPP)** Architecture, combining SQL Server relational database technology with Azure cloud scale-out capabilities. The Azure SQL warehouse consist of separate storage and compute nodes. It's easy to deploy, seamlessly maintained, and fully fault tolerant because of automatic back-ups. This service also provides security features such as **authentication, authorization and data encryption.**

References
https://docs.microsoft.com/en-us/azure/sql-data-warehouse/sql-data-warehouse-overview-what-is

Features

- The storage and compute nodes are functions independently so that we can scale out the storage capacity or compute speed without depending on other nodes.
- Easy integration with PowerBI, ADF, Azure Table storage, SQL database azure services as well as third party tools (see https://docs.microsoft.com/en-us/azure/sql-data-warehouse/sql-data-warehouse-partner-business-intelligence).
- Increase, decrease, pause, or resume compute nodes.
- Automatically backs-up (un-paused) data at regular intervals.

10.3.2.8 Azure SQL database

The Azure SQL database is a cloud based transactional database service. Azure SQL enables to store relational data in the cloud. It is based on SQL server database engine.

Features

- Encryption allow to protect sensitive data such as card number etc.
- Auto Scale is used to change the Azure SQL database tier. Azure SQL database offers three types of tiers (basic, standard and premium) with multiple performance levels to handle different workloads.
- Auditing tracks database events and writes them to an audit log in your Azure Storage account.
- Database authentication using Azure Active Directory (AAD).
- Active geo-replication replication.
- Application Role enables an application to run with its own, user-like permissions and also enable access to specific data to only those are running application.
- Azure Active directory authentication for user authentication of databases.

References
https://docs.microsoft.com/en-us/azure/sql-database/sql-database-features

10.3.2.9 Backup service

This service can be used to take backup, restore your data in the cloud. It provides set of components/agent which we need to install on on-premises or cloud based machine to take data backup. The component, or agent, that you deploy depends on what you want to protect.

References
https://docs.microsoft.com/en-us/azure/backup/backup-introduction-to-azure-backup#which-azure-backup-components-should-i-use

10.3.2.10 Virtual Machines

Azure virtual machines are useful when you want to create/deploy windows or Linux machine on cloud. The virtual machine will be useful for development and test, application in cloud and extended datacenter. It provides various machine images. We can scale the machine instances as per our business requirement.

Supported images
https://docs.microsoft.com/en-us/azure/virtual-machines/virtual-machines-windows-overview

10.3.2.11 HDInsight service

The HDInsight is the service which provides complete Hadoop infrastructure on cloud. This service uses Hortonworks as distribution platform. It supports different versions of Hadoop components. HDInsight enable developer to do code with various platform such as eclipse, Scala, python and R etc. The service protect your assets by providing sign-on (SSO), multi-factor authentication and AAD options and also support of Jupyter and Zeppelin for Data scientists.

Supported components
https://docs.microsoft.com/en-us/azure/hdinsight/hdinsight-component-versioning

10.3.2.12 Azure Active Directory

Azure AD is identity management service. It has capabilities of multi-factor authentication, device registration, self-service password management, role based access control, application usage monitoring, rich auditing, group based access management and security monitoring and alerting. Azure AD can be used to secure on-premises devices also.

Reference
https://docs.microsoft.com/en-us/azure/active-directory/active-directory-whatis

10.3.2.13 Azure Data lake store

Azure data lake store is designed for big data analytics workload, it can be accessible from HDInsight cluster on cloud using Web-HDFS REST API. This store is compatible with different open source projects. Azure Data Lake enables you to capture data of any size, type, and ingestion speed in one single place for operational and exploratory analytics.

Compatible open source projects
https://docs.microsoft.com/en-us/azure/data-lake-store/data-lake-store-compatible-oss-other-applications

Features

- Provides authentication (to all stored data in data lake store), access control (to directory and files in store) and encryption (on data stored in store)
- Support Geo-redundancy on cloud.

10.3.2.14 Microsoft - Power BI:

Power BI is a cloud-based business analytics service that gives a single view of most critical business data. Developers can access diverse data source such as on-premises analysis technologies like SSAS, database services and analysis services that run in the cloud, and cloud applications from Microsoft and other vendors. They can create monitor the health of business using a live dashboard, create rich interactive reports with Power BI Desktop and access data on the go with native Power BI Mobile apps.

References
https://powerbi.microsoft.com/en-us/features/

10.4 IBM Bluemix

10.4.1 Overview

IBM Bluemix is a cloud platform as a service (PaaS) developed by IBM. It supports several programming languages and services as well as integrated DevOps to build, deploy, run, and manage applications on the cloud. Bluemix is based on Cloud Foundry open technology and runs on Soft Layer infrastructure.

Components of IBM Bluemix

- **Data Connect:** A fully managed data preparation and data movement service for users to access and prepare their data for analytics.
- **BigInsights for Apache Hadoop:** A fully managed service which provides Apache Hadoop infrastructure and allows to develop analytics applications on a Hadoop environment.
- **Compose for MongoDB:** A scalable JSON document database service to power web and mobile applications without database management headaches.
- **Compose for MySQL:** A relational database service which provides MySQL server 5.5 as backend storage.
- **IBM Watson Analytics:** A full-service offering to provide advanced analytics without the complexity. Provides a data discovery service to guide data exploration, automates predictive analytics to help create effortless dashboards and infographics.
- **IBM Watson Machine Learning:** A full-service offering that allows to integrate predictive capabilities within applications (does not provide a development environment comparable to Azure).
- **Insight for Twitter:** A service providing enrichments (e.g. author with Gender and Permanent Location) and Sentiment (e.g., positive, negative, ambivalent, or neutral for Tweets in English, German, French, and Spanish).
- **ElephantSQL:** This service provides PostgreSQL open source relational database. This database service is managed by a third party on IBM cloud.
- **Streaming Analytics:** A fully managed service allowing to ingest, analyze, monitor, and correlate data as it arrives from various real-time data sources.
- **Cloudant NoSQL DB:** A fully managed NoSQL document oriented database service, it uses CouchDB as the underlying database.
- **Compose for Elasticsearch:** A full-text search engine with the indexing strength of a JSON document database.
- **Compose for etcd:** Etcd is a key-value store developers can use to hold the always-correct data needed to manage a distributed cluster and enable automatic configuration backup feature.
- **Compose for RabbitMQ:** This service handles asynchronous messages between applications and databases, allowing to ensure separation of the data and application layers.
- **dashDB for Analytics:** A SQL cloud based relational database service optimized for data warehouse workloads.
- **Weather Company Data:** use this service to incorporate weather data from the Weather Company for a specified geolocation into your IBM Bluemix application.
- **IBM Graph:** A graph database-as-a-Service. The database is used to store and querying data points, their connections and properties. The service can be used for building recommendation engines, analyzing social networks, and fraud detection.

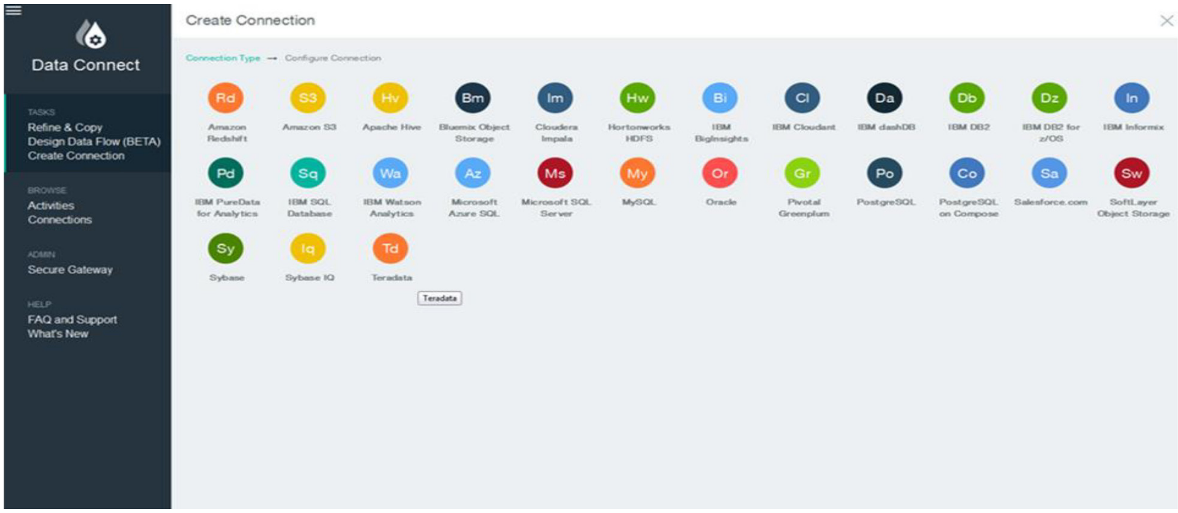
- **ClearDB MySQL database:** ClearDB is geo-distributed database-as-a-service for MySQL powered applications. It's built on native MySQL 5.5, and managed by a third party.
- **Compose for Redis:** Redis is an open source key/value in-memory data structure store, used as a database, a message broker and a cache. Redis supports master-slave asynchronous replication.
- **Compose for RethinkDB:** RethinkDB is distributed JSON document oriented database. It uses the ReQL query language built around function chaining and available in libraries for several languages.
- **Decision optimization:** This service provides a platform to help businesses solve difficult planning and scheduling challenges with the use of analytics.
- **IBM Master Data Management on Cloud:** IBM Master Data Management on Cloud provides IBM Master Data Management Advanced Edition on IBM Softlayer global cloud infrastructure.
- **IBM DB2 on Cloud:** DB2 on Cloud is a fully managed Relational Database service; extended with support of Object-Oriented features.
- **Information Server on Cloud:** This service provides different types of application development infrastructure (i.e. as per your need) for your application on cloud or on-premises with some predefined plans (e.g. 2GB storage, 2 CPU core etc.).
- **Informix on Cloud:** Is a fully managed relational database, similar to DB2, but it has a few NoSQL features which allow to combine structured as well as un-structured data, and allow to store JSON in store engine.
- **Lift:** Lift is a data and database migration service from on-premises to cloud.
- **Geospatial Analytics:** This service enable to connect various devices (cars, boats, people, even your dog) and to track or monitor them in different regions.
- **Namara.io Catalog:** This service aggregates open data released by all levels of government and present it to users in a single portal. It organizes and catalogues this public data, providing users with API access to high value information. This is a third-party service.
- **TinyQueries:** TinyQueries is a SQL generator service which can be connected to a SQL database running on Bluemix; queries are specified on a query editor in an OO style, similar to ORM tools. Users can create and publish queries as a REST API. The service is managed by a third-party.
- **IBM Backup Storage:** Bluemix provides turnkey Evault and R1Soft automatic backup solutions. This service is useful to take backup of **application data as well as servers**.
- **Compose for PostgreSQL:** A fully managed relational database service providing access to the open source PostgreSQL database.

10.4.2 Bluemix components in detail

10.4.2.1 Data Connect

IBM Bluemix Data Connect is a fully managed, data preparation and data integration service. Data Connect empowers Data Scientists, Data Analyst, and Data Engineers to **discover, cleanse, standardize, shape, transform and move data** to support their analytic or application development adventures. Use Data Connect to tap into multiple data sources, including multiple databases, applications and across different cloud platforms in order to generate actionable insights faster and build great applications.

Supported Connectors



10.4.2.2 BigInsights for Apache Hadoop

Develop analytics applications by using open source Apache Hadoop and Apache Spark APIs without having to manage the platform. Available under two different service plans: Enterprise and Basic.

- The Basic plan is a multi-tenant service based on containers on bare metal servers and enables users to instantiate and scale clusters within minutes; pricing is per hour.
- The Enterprise plan provides dedicated resources (single tenant), high availability and value-add services (for example BigSheets, Big SQL, and Text Analytics); pricing model is by subscription for a period agreed in advance.

References
https://console.ng.bluemix.net/docs/services/BigInsights/index.html

10.4.2.3 Compose for MongoDB

MongoDB, the document oriented database with powerful indexing and querying, aggregation and wide driver support, has become the go-to JSON data store for many startups and enterprises. IBM Compose for MongoDB is a fully managed MongoDB service, providing an easy, auto-scaling deployment system which delivers high availability and redundancy, automated and on-demand no-stop backups, monitoring tools, integration into alert systems, performance analysis views and more.

Features

- Support ad-hoc queries: developers can search by field, range query and it also supports regular expression searches.
- Indexing: any field in a document can be indexed.
- Aggregation: batch data processing and aggregate calculations using native MongoDB operations.
- Security: authentication, authorization, etc.
- Load Balancing: automatic data movement across different shards for load balancing. The balancer decides when to migrate the data and the destination Shard, so they are evenly distributed among all servers in the cluster.
- Can develop map-reduce job using JavaScript language.

References
https://console.ng.bluemix.net/catalog/services/compose-for-mongodb/
http://www.mongodbspain.com/en/2014/08/17/mongodb-characteristics-future/

10.4.2.4 Compose for MySQL

The IBM Bluemix Compose for MySQL provides a fully managed service to MySQL relational database, offering an easy, auto-scaling deployment system that delivers high availability and redundancy, and automated backups.

10.4.2.5 IBM Watson Analytics

IBM Watson Analytics offers the benefits of advanced analytics without the involved complexity. A smart data discovery service available on the cloud, it guides data exploration, automates predictive analytics and enables effortless dashboard and infographic creation.

Features

- Discovery platform powered by cognitive capabilities
- Enables its users to do complex analytics in minutes
- Helps business users drive their individual business in a better way
- Provides a natural language interface to asking questions for analytics

While true cognitive computing is yet to be realized anywhere, there is no question that the machine learning (ML) and early artificial intelligence (AI) behind IBM Watson Analytics is an impressive accomplishment. This smart data science assistant acts both as servant and guide for users with a wide range of data science skill sets and in roles ranging from business analyst to data scientist. It is comparable in breadth to the Microsoft PowerBI and Tableau tools.

There are 32 connectors to ease use of data from those sources. A sample listing of business connectors includes spreadsheets (CSV, XLS, TXT), Eventbrite, Hubspot, OneDrive,Paypal, SugarCRM, SurveyMonkey, and Twitter.

IBM Watson Analytics also lets you directly query a variety of databases including Cloudera Impala, Microsoft Azure, MySQL, Oracle, PostgreSQL, PostgreSQL on Compose, Structured Query Language (SQL) Server, Sybase, Sybase IQ, and Teradata.

References
IBM Watson Analytics: https://www.ibm.com/analytics/watson-analytics/us-en/

10.4.2.6 IBM Watson Machine Learning

IBM Watson Machine Learning is a full-service Bluemix offering that makes it easy for developers and data scientists to work together to integrate predictive capabilities with their applications. Built on IBM’s proven SPSS analytics platform, Machine Learning allows you to develop applications that make smarter decisions, solve tough problems, and improve user outcomes. IBM SPSS Modeler or Data Science Experience is required for authoring and working with models and pipelines. The focus of the Machine Learning service is deployment: the service is a set of REST APIs that can be called from any programming language.

References
SPSS Modeler and algorithms: https://www.ibm.com/support/knowledgecenter/SS3RA7
Data Science Experience and modeling algorithms: http://datascience.ibm.com/

10.4.2.7 Insight for Twitter

This service provides sentiment and other enrichments for multiple languages, based on deep natural language processing algorithms from IBM Social Media Analytics. Real-time processing of Twitter data streams is fully supported; configurable through a rich set of query parameters and keywords. Insights for Twitter includes RESTful APIs that allow you to customize your searches and returns Tweets and enrichments in JSON format.

Features

- Twitter data: Search Twitter content from the Twitter Decahose (10% random sample of Tweets) and PowerTrack stream (100% access to Tweets). The content store is frequently refreshed and indexed, making searches dynamic and fast.
- Enrichments: Get advanced enrichments based on real-time analysis of the Twitter data, like author with Gender and Permanent Location (defined by country, state, and city) and Sentiment (e.g., positive, negative, ambivalent, or neutral for Tweets in English, German, French, and Spanish).
- PowerTrack: Create, edit, aggregate, or remove rules and tracks to customize your connection to the content store and optimize the performance of your apps.

10.4.2.8 ElephantSQL

ElephantSQL is a PostgreSQL database hosting service, offering databases ranging from shared servers for smaller projects and proof of concepts, up to enterprise grade multi server setups.

10.4.2.9 Streaming Analytics

Streaming Analytics is powered by IBM Streams, an advanced analytic platform that you can use to ingest, analyze, monitor, and correlate data as it arrives from real-time data sources. The Streaming Analytics service gives you the ability to deploy Streams applications to run in the Bluemix cloud. This service also provides monitoring for all the jobs which are running on your stream instance. There are two ways that you can use your Streaming Analytics instance:

- Interactively – through the Streaming Analytics console
- Programmatically – through the Streaming Analytics REST API

A common use case is when you have a set of devices producing information in real-time that you need to analyze. There is an IoT service to collect the data from the devices which can be used in combination with Streaming analytics.

References

https://www.ibm.com/blogs/bluemix/2015/10/getting-started-with-streaming-analytics-and-iot/ https://developer.ibm.com/streamsdev/docs/streaming-analytics-now-available-bluemix/

10.4.2.10 Cloudant NoSQL DB

Cloudant NoSQL DB is a fully managed document oriented database-as-a-service. This service stores data as documents in JSON format. It is designed for modern web and mobile applications that leverage a flexible JSON schema. Cloudant is built upon and compatible with Apache CouchDB.

Features

- Store, delete, and retrieve the documents via a REST API.
- Support of features like full-text search, geo-replication etc.
- Cloudant Sync provides data access for mobile devices.
- Easy integration with Apache Spark to perform advanced analytics on JSON documents.
- Support of declarative Cloudant Query.
- Cloudant's horizontal scaling architecture can handle millions of users and terabytes of data to grow seamlessly alongside your business.

References
https://console.ng.bluemix.net/docs/services/Cloudant/index.html#getting-started- with-cloudant
https://console.ng.bluemix.net/catalog/services/cloudant-nosql-db/

10.4.2.11 Compose for Elasticsearch

Elasticsearch combines the power of a full text search engine with the indexing strengths of a JSON document database to create a powerful tool for rich data analysis on large volumes of data. With Elasticsearch searching can be scored for exactness, letting users dig through their data set for those close matches and near misses which they could be missing.

References
https://console.ng.bluemix.net/catalog/services/compose-for-elasticsearch/

10.4.2.12 Compose for etcd

Etcd is a key-value store developers can use to hold the always-correct data needed to manage a server cluster for distributed server configuration management. etcd uses the RAFT consensus algorithm to assure data consistency in the cluster and also enforces the order in which operations take place in the data so that every node in the cluster arrives at the same result in the same way.

References
https://console.ng.bluemix.net/catalog/services/compose-for-etcd/

10.4.2.13 Compose for RabbitMQ

RabbitMQ asynchronously handles the messages between your applications and databases, allowing you to ensure separation of the data and application layers. RabbitMQ enables developers to route, track, and queue messages with customizable persistence levels, delivery settings, and confirmed publication.

Features

- High availability queue: Queues can be mirrored across several machines in a cluster, ensuring that even in the event of hardware failure your messages are safe.
- Management UI: RabbitMQ ships with an easy-to use management UI that allows you to monitor and control every aspect of your message broker.
- Many Clients: There are RabbitMQ clients for almost any language you can think of.

References
http://www.rabbitmq.com/features.html
http://www.rabbitmq.com/documentation.html
https://console.ng.bluemix.net/catalog/services/compose-for-rabbitmq/

10.4.2.14 dashDB for Analytics

IBM dashDB for Analytics is a fully managed SQL cloud database service, featuring an MPP architecture optimized for data warehouse and analytics workloads. dashDB for Analytics allows to scale and pay for compute and storage node independently. A related but different service, IBM dashDB for Transactions SQL Database, is a cloud database that is optimized for OLTP workloads.

Features

- Combining the best of DB2 and Netezza technology with in-memory data processing, columnar tables, and in-database analytics.
- Easily integrate with other IBM cloud services such as Watson.
- Includes daily backups, at-rest database encryption, and SSL connections.
- Provides R language for data analysis.

References
https://www.ibm.com/analytics/us/en/technology/cloud-data-services/dashdb-managed-service/

10.4.2.15 Weather Company Data

This service lets you integrate weather data from The Weather Company into your IBM Bluemix application. You can retrieve weather data for an area specified by a geolocation. The data allows you to create applications that solve real business problems where weather has a significant impact on the outcome. Weather data for some countries and regions are not available.

Features

- Hourly forecast: An hourly weather forecast for the next 48 hours starting from the current time, for a specified geolocation.
- Historical data: Observed weather data from site-based observation stations for a specified geolocation that includes current observations and up to 24 hours of past observations.
- Location services: The ability to look up a location name or geocode (latitude and longitude) to retrieve a set of locations that match the request.
- Weather alerts: Government-issued weather alerts, including weather watches, warnings, statements, and advisories issued by the National Weather Service (US), Environment Canada, and MeteoAlarm (Europe).

References
https://console.ng.bluemix.net/catalog/services/weather-company-data/

10.4.2.16 IBM Graph

IBM Graph is an easy-to-use, fully-managed graph database service for storing and querying data points, their connections, and properties. IBM Graph offers an Apache TinkerPop3 compatible API and plugs into a Bluemix application seamlessly. This service can be used for building recommendation engines, analyzing social networks, and fraud detection.

Features

- Powered by Apache TinkerPop3 –IBM Graph is based on the TinkerPop stack for building high-performance graph applications.
- Scale Seamlessly: customers can start small and scale on-demand as their data grows.
- Highly Available: architected to ensure the service is always up and data is always accessible.
- Managed 24x7

References
https://console.ng.bluemix.net/catalog/services/ibm-graph/

10.4.2.17 ClearDB MySQL database

ClearDB is a reliable, fault tolerant, geo-distributed database-as-a-service for MySQL powered applications. It's built on native MySQL 5.5, and requires no use of special engines or alterations to your code. The service is managed by a third party.

10.4.2.18 Compose for Redis

Redis is an open-source, very fast, key/value low maintenance store. Compose's platform gives users a configuration pre-tuned for high availability based on master-slave asynchronous replication and locked down with additional security features. Developers can run **atomic operations** such as appending to a string; incrementing the value in a hash; pushing an element to a list; computing set intersection, union and difference; or getting the member with highest ranking in a sorted set.

Features

- Support of data structures such as strings, hashes, lists, sets, sorted sets etc.
- Transaction support: MULTI, EXEC, DISCARD and WATCH are the foundation of Redis. The EXEC command triggers the execution of all the commands in the transaction. The Calling DISCARD instead will flush the transaction queue and will exit the transaction.
- Support of multiple programming languages.
- Automatic failover: Sentinel constantly checks if your master and slave instances are working as expected.

References
https://redis.io/topics/introduction
https://redis.io/clients

10.4.2.19 Compose for RethinkDB

RethinkDB is an open-source distributed JSON document oriented database. Instead of polling for changes, the developer can tell RethinkDB to continuously push updated query results to applications in real-time. It has an integrated administration and exploration console. RethinkDB uses the ReQL query language which is built around function chaining and is available in client libraries for JavaScript, Python and Ruby. With ReQL it is possible to utilize RethinkDB server side features such as distributed joins and subqueries across the cluster's nodes. RethinkDB also supports secondary indexes for better read query performance.

Features

- RethinkDB supports ReQL as query language which allows users to perform CRUD operations, aggregations including map-reduce & group-map-reduce, Joins, Full sub-queries.
- Supports of primary key, compound, secondary, geospatial, and arbitrarily computed indexes.
- RethinkDB provides official libraries for JavaScript/Node.js, Python, Java, and Ruby.
- RethinkDB can be manually deployed on cloud platforms such as AWS.
- RethinkDB supports automatic primary re-election using the Raft algorithm which detects the server from which a heartbeat is not being received.

References
https://www.rethinkdb.com/faq/
https://www.rethinkdb.com/docs/comparison-tables/

10.4.2.20 Decision optimization

IBM's Decision Optimization Center provides a platform to help businesses solve difficult planning and scheduling challenges with the use of analytics. Users describe a situation as an optimization model, using their data and criteria, and Decision Optimization identifies the best solution. The service is based on a sophisticated analytics technology, called Prescriptive Analytics (touted as the third and latest generation of analytics systems). This service is useful in below use cases:

- What are the best actions to take to reduce predicted customer churn?
- What maintenance should we do to avoid the predicted failure in the machine?
- Given this customer's past spending level and products purchased, which product should I offer to cross-sell?

Features

- Decision Optimization uses the market-leading CPLEX Optimizers to solve real-world optimization problems quickly and efficiently.
- Simply connect with APIs: Use a Decision Optimization API to submit an optimization problem, including mathematical model and data, to the optimization engines. The API returns results to your application. APIs are available in REST, Java, Node.js, and Python.

References
http://quebit.com/what-we-offer/services/consulting/decision-optimization/

10.4.2.21 IBM Master Data Management on Cloud

This service offers the rich features of an on-premises IBM Master Data Management Advanced Edition deployment without the cost, complexity, and risk of managing your own infrastructure. As the IBM MDM AE, is multi-domain, and the MDM hub can be deployed in physical or virtual style.

Features

- Improved time to value: using this offering reduces the time required for provisioning and deploying MDM by providing preinstalled MDM configurations for production and development MDM environments in an IBM SoftLayer cloud-hosted environment.
- Flexible deployment: comes in Small, Medium, and Large configurations.

References
https://console.ng.bluemix.net/catalog/services/ibm-master-data-management-on-cloud/

10.4.2.22 IBM DB2 on Cloud

DB2 is a Relational Database Management System (RDBMS) originally introduced by IBM. DB2 is designed to store, analyze and retrieve the data efficiently. DB2 product is extended with the support of Object-Oriented features.

Features

- Securely Move Data on-premises to the DB2 instance on Cloud.
- Connectivity: Supports all native DB2 drivers, SQL, .NET, ODBC and JDBC. Compatible with Netezza and Oracle. Supports backups to S3 and Swift object storage.

10.4.2.23 Information Server on Cloud

As the previous service, IBM Information Server on Cloud provides IBM InfoSphere Information Server functionality on IBM SoftLayer global cloud infrastructure. It offers the rich features of the on-premises Information Server deployment without the cost, complexity, and risk of managing your own infrastructure.

Features

- Flexible packaging
  - Separate products:
  - Information Governance Catalog on Cloud: encourages a standardized approach to discovering IT assets and defining a common business language, and manages and explore data lineage to create trusted information that supports data governance and compliance efforts.
  - DataStage on Cloud: provides powerful and scalable ETL platform, supports near real time data integration, uses a high performance parallel framework and supports extended metadata management and enterprise connectivity.
  - Information Server on Cloud Data Quality: cleanses data and monitors data quality in batch and in near real time.
- Information Server on Cloud Enterprise Edition packages the three above products

References
https://console.ng.bluemix.net/catalog/services/information-server-on-cloud/

10.4.2.24 Informix on Cloud

Is a fully managed relational database, similar to DB2, but it has a few NoSQL features which allow to combine structured as well as un-structured data, and allow to store JSON in store engine.

References
http://db-engines.com/en/system/DB2%3BInformix
https://www.ibm.com/analytics/us/en/technology/informix/

10.4.2.25 Lift

Lift is a fully managed data and database migration service. Use Lift to quickly, easily and securely migrate data from your on-premises data sources to the cloud. When migrating between heterogeneous database engines, an integrated schema compatibility assessment walks you through the process of converting your source schema to your target engine.

Features

- Can migrate your data from IBM Pure Data System for Analytics and IBM DB2 to the IBM dashDB cloud data warehouse and IBM DB2 on Cloud.
- Secure migration: Any data movement across the internet requires strong encryption so that your data is never compromised.

References
http://www-03.ibm.com/software/products/en/ibm-bluemix--lift

10.4.2.26 Geospatial Analytics

This service enables users to connect various devices (cars, boats, people, even your dog) and to track or monitor them in different regions. With this service userds can connect to the devices which supports MQTT (MQTT is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol) protocol, can control monitoring regions using geospatial API also.

References
https://console.ng.bluemix.net/catalog/services/geospatial-analytics/
https://www.ibm.com/blogs/bluemix/2014/12/find-bluemix-geospatial-analytics/

10.4.2.27 Namara.io Catalog

This service aggregates open data released by all levels of government and present it to users in a single portal. It organizes and catalogues this public data, providing users with API access to high value information. This is a third-party service.

References
https://console.ng.bluemix.net/catalog/services/namaraio-catalog/

10.4.2.28 TinyQueries

TinyQueries is a SQL generator service which can be connected to a SQL database running on Bluemix; queries are specified on a query editor in an OO style, similar to ORM tools. Users can create and publish queries as a REST API. The aim of TinyQueries is to do spent less time developing SQL query and get accurate and fast results.

References
https://www.ibm.com/us-en/marketplace/4791
http://docs.tinyqueries.com/

10.4.2.29 IBM Backup Storage

Bluemix provides turnkey EVault and R1Soft automatic backup solutions, as well as the ability to create your own using our virtual or bare metal server running your own backup application. This service is useful to take backup of application data as well as servers.

Evault Backup

Evault Backup is an enterprise-level backup storage and disaster recovery solution hosted on an internal iSCSI mount, available for taking backup between servers in one or more data center. Backups can be set to follow hourly, daily, weekly, or custom schedules.

Features

EVault Agent's easily take the backup of below mentioned server's/databases.

- MS Exchange - Provides backup and restore for entire MS Exchange databases. Alternatively you may want to individually configure your backup (e.g. Define folders or mailboxes)
- MS SQL Server - Backup and save your SQL Server and transaction log.
- MS SharePoint - Backup your entire portal, a single or multiple location collections or single documents.
- Oracle servers - Provides online backup capabilities for Oracle databases.

R1Soft Server Backup

R1Soft Server Backup provides high-performance **disk-to-disk server backup**, featuring a central management and data repository. It protects data at block level, and unique disk blocks on the server are stored only once across all recovery points, increasing storage efficiency.

Features

R1Soft service does come with backup agents:

- bare-metal restores
- Databases (MS SQL, MySQL, MS Exchange)
- R1Soft does not have an Oracle or DB2 agent.
- R1Soft can also backup data from VMs deployed on self-managed hypervisors, KVM, VMWare and Hyper-V.
- R1Soft can take backup from various machines such as CentOS, Debian, Red hat, Ubuntu, Windows server 2008 and Windows server 2012.

References

https://www.ibm.com/cloud-computing/bluemix/server-backup
https://softlayerslayers.wordpress.com/2015/09/17/backup/

About Persistent Systems

Persistent Systems (BSE & NSE: PERSISTENT) builds software that drives our customers' business; enterprises and software product companies with software at the core of their digital transformation. For more information, please visit: www.persistent.com

India
Persistent Systems Limited
Bhageerath, 402,
Senapati Bapat Road
Pune 411016.
Tel: +91 (20) 6703 0000
Fax: +91 (20) 6703 0009

USA
Persistent Systems, Inc.
2055 Laurelwood Road, Suite 210
Santa Clara, CA 95054
Tel: +1 (408) 216 7010
Fax: +1 (408) 451 9177
Email: info@persistent.com

DISCLAIMER: “The trademarks or trade names mentioned in this paper are property of their respective owners and are included for reference only and do not imply a connection or relationship between Persistent Systems and these companies.”