Nomfundo Mthabela

ST10083777

CLDV6212

Part 1

**A.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Traditional On-Premises** | | **Modern Cloud** | |
| **On-premises**  **definition** | **On-premises example** | **Cloud definition** | **Cloud example** |
| **Monolithic:**  Monolithic software is constructed to operate as a single, integrated unit, where the program's elements and functions are closely interconnected rather than being loosely connected, as seen in modular software. In a monolithic design, every component and its related parts need to be present for the code to compile and the software to function. | **Monolithic:**  Consider a monolithic e-commerce Software as a Service (SaaS) application. It could encompass a web server, load balancer, a service responsible for displaying product images from a catalog, an ordering system, a payment module, and a delivery component.  Due to their wide-ranging functionality, monolithic software tools often possess extensive and sizable codebases. | **Decomposed:**  In the context of cloud computing, a deconstructed modern cloud is a cloud infrastructure that was created utilizing a micro-service or even server-less architecture. A deconstructed cloud is built of separate, flexibly interconnected services or operations rather than a monolithic cloud arrangement where all amenities are closely interwoven. These services offer more versatility and agility when handling cloud resources since they can dimension, implement, and modify autonomously. | **Decomposed:**  Consider a massive powered by the cloud online store that is divided up into a number of services rather than depending on a single monstrous system to handle activities like authentication of users, inventory control, and the processing of payments.  The fulfillment of orders service: handles delivery and shipping.  Analytical service: Offers perceptions into patron behavior. |
| **Designed for predictable scalability:**  The term "traditional on-premises scalability" refers to the capacity to increase or modify computing capabilities within a physical data center to accommodate shifting workloads. In order to effectively use software or apps, adequate speed is also required. To address this expanding burden, cloud scalability is used. Where the constant distribution of resources is necessary for managing the workload statically, scalability is frequently used. | **Designed for predictable scalability:**  Elasticity is used to meet the organization's dynamic needs, whereas scalability is applied to meet its static needs. Similar to other cloud services, scalability requires clients to pay for each use. In summary, we can state that scalability is helpful in situations where workload is static and substantial (Geeksforgeeks, 2023). | **Designed for elastic scale:**  Elasticity is the capacity of clouds to dynamically increase or decrease infrastructure resources in response to rapid changes in demand, allowing for effective workload management. This pliability lowers the cost of the infrastructure. | **Designed for elastic scale:**  Organizations may put into practice techniques like resource pooling, virtualization, and capacity planning in order to accomplish greater predictability elasticity in on-premises environments. These methods might fall short of the agility and value for money provided by solutions that utilize the cloud whereby elasticity is a key component. |
| **Relational database:**  Ancient relational database systems are exceptionally organized databases made up of standardized tables of data linked together by primary/foreign variables and searched through SQL (Techtarger, 2023). | **Relational database:**  A normal commercial order entry database, for example, would have a table that represents a client, with fields for the client's address, phone number, name, and so on (Sinha, 2020). A separate table might summarize an order, containing details such as the goods, customer, and price. | **Polyglot persistence:**  In order to accommodate the specific requirements of various data kinds in corporate programs, the term polyglot persistence encompasses the use of a variety of data storage techniques and technological advances (Techtarger, 2023). | **Polyglot persistence:**  Traditionally, these were handled by a single database, necessitating intricate information transformation (Techtarger, 2023). To prevent stressing an individual repository, polyglot persistence recommends employing dedicated repositories for every kind of information (Techtarger, 2023). |
| **Synchronized processing:**  The client device transmits the query to the servers, which opens the connection in a synchronous sequence shortly after (Predica, nb). | **Synchronized processing:**  When you access a website, you often need to wait for it to load before it displays the information (Predica, nb). Similarly, calling a friend on a mobile phone is a common example (Predica, nb). | **Asynchronous processing:**  Asynchronous operation denotes the operation of a process autonomously of other processes. Asynchronous processing operates independently of the sessions from which requests are submitted and responses are obtained (Techtarger, 2023). There is no direct association between an inquiry and a response, and no assumptions may be made regarding the time frame of the response. | **Asynchronous processing:**  An application for checking one's credit rating. A terminal operator can utilize a regional transaction for entering a series of requests without anticipating a response to each one(IBM, n.b.)**.** |

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
| **MTBF:**  Using MTBF for analysis promotes preventive maintenance, reduces reactive maintenance events, reduces overall downtime, and enables teams to successfully organize their activities (IBM, n.b.). The length between system failures happening during ordinary operations during a specific timeframe is monitored, and the standard deviation of these periods is computed (IBM, n.b.). | **MTBF**:  As an example, consider an automobile engine. You'd use MTBF, which signifies the mean time between failures, to calculate the period between unplanned engine upkeep (IBM, n.b.). | **MTTR**:  Demonstrates the average time during business hours between the start of an incident and the point at which regular operations have been entirely recovered (IBM, n.b.). | **MTTR**:  During this time, there were a total of ten failures, and the facilities were continuously conducting repairs (Atlassian, n.b.). |
|  |  |  |  |

**B.**

[iKoBKeUALPEhIACvvzXDrdITymK4UovowX11yEAzFuHDtSRQ==](https://st10083153fa.azurewebsites.net/api/id/%7bid%7d?code=DO-hXkiKoBKeUALPEhIACvvzXDrdITymK4UovowX11yEAzFuHDtSRQ==)

Referencing list

* TechTarget. N.b. *On premises vs. cloud: The major similarities and differences*, n.b. [Online]. Available at: <https://www.techtarget.com/searchcloudcomputing/tip/Evaluate-on-premises-vs-cloud-computing-pros-and-cons> [Accessed 28 August 2023].
* Techtarget. N.b. *Decomposing a monolithic database for microservices*, n.b. [online]. Available at: <https://www.techtarget.com/searchapparchitecture/tip/Decomposing-a-monolithic-database-for-microservices> [Accessed 3 September 2023].
* TechTarget. 2022. *monolithic architecture*, May 2022. [online]. Available at: <https://www.techtarget.com/whatis/definition/monolithic-architecture> [Accessed 25 August 2023].
* Techtarget. 2023. *polyglot persistence*, February 2023. [Online]. Available at: <https://www.techtarget.com/searchapparchitecture/definition/polyglot-persistence> [Accessed 28 August 2023].
* Predica. N.b. *Synchronous vs. asynchronous communication (in the cloud and beyond)*, n.b. [Online]. Available at: <https://www.predicagroup.com/blog/synchronous-vs-asynchronous-communication/> [Accessed 28 August 2023].
* Hevo. 2022. *Snowflake On Premise Comparisons: 10 Key Critical Differences,* 29 December 2022]. Available at: <https://hevodata.com/pricing/pipeline/> [Accessed 2 August 2023].
* GeeksforGeeks. 2023. *Scalability and Elasticity in Cloud Computing*, 16 January 2023. [Online]. <https://www.geeksforgeeks.org/scalability-and-elasticity-in-cloud-computing/> [Accessed 25 August 2023].
* IBM. N.b. *What is a relational database?*, n.b. [Online]. Available at: <https://www.ibm.com/topics/relational-databases> [Accessed 26 August 2023].
* Sinha, R. (2020). *The journey of Database from On Premise to Cloud*. [online] Medium. Available at: [https://rumasinha.medium.com/the-journey-of-database-from-on-premise-to-cloud-2c99edce15ea#:~:text=Traditional%20relational%20databases%20are%20highly](https://rumasinha.medium.com/the-journey-of-database-from-on-premise-to-cloud-2c99edce15ea) [Accessed 11 Sep. 2023].