This Vs That

Prof. Jigar Pandya • Jul 5 (Edited Aug 14)

- Programming paradigm
- Computing Architectures
- Scaling
- Data Processing
- Data Address
- Types of cloud
- Consistency Requirements
- Data Stores
- Data Backup and Size
- Latency Vs Throughput
- Schema on Write Vs Schema on Read

Programming paradigm

Structured programming, object oriented programming, functional programming, aspect oriented programming, Parallel Programming (map-reduce programming), etc.

Computing:

Client Server Architecture Vs Master Slave Architecture

Shared Nothing Vs Shared Disk Architecture

Scaling:

Horizontal scaling (Scale Out) vs Vertical scaling (Scale Up)

Data Processing:

OLTP (Online Transaction Processing) Vs OLAP (Online Analytical Processing) VS RTAP (Real Time Analytical Processing)

Data Address:

Data at rest Vs Data in motion

Types of clouds:

cloud(web)/public vs on-premise/private vs Hybrid

Consistency Requirements:

ACID (Atomicity, Consistency (Strong), Isolation, Durability)

Vs

BASE (Basically Available Soft state Eventual consistent)

Data Stores:

SQL-NoSQL(Document, Columnar, Key-value, Graph)-NewSQL

Data Backup and Size:
Archival vs Compression vs Both

Latency Vs Throughput

Latency has to do with time while throughput (a formula output divide by input in general but with networks has to do with size or capacity) can be primarily with quantity.

High Latency Systems may be okey with time gap/network delays but like online gaming, stocks etc are low latency systems where immediate response is a must to achieve. Observe this w.r.to CAP theorem and BASE.

Schema on Write Vs Schema on Read

Traditionally we create schema and data gets generated by applications accordingly. i.e. Relational Databases and applications OLTP. Its considered to be schema on write, meaning we have schema created for storing data at the time or writing and of course the same data store is utilised primarily for reading too.

Schema on Read can be considered to be you do not need to know the purpose of usage in advance. The mixed flow of data (structured, semi and un) dumped together may be in a data lake or distributed file systems like HDFS or NoSQL databases like mongoDB. Importantly either Hive like Query or code logic in applications will impose required schema on the data while reading for a purpose. Know that thus multiple purpose can be fulfilled.

Also, refer to query first approach.

p.s. markup vs markdown. copyright vs copyleft.
p.s Full forms for fun GNU (GNU is Not UNIX), YAML(Yaml ain't Another Markup Language)

Class comments

Add class comment…