Recent trends in Data and Information Architecture

Informatics society of Iran (ISI) webinars
Enterprise architecture group

- Introduction to data and information architecture
- Modern architecture
- Architecture trends
- Revolutionize architecture
- Modern architect

Hamid Reza Aghiri
Informatics Services Company

- **Current State of "Data"**

- **What's Data Architecture (DA)**

- **"DA" vs Information Architecture (IA)**

INTRODUCTION

Current State of "data" and "information"

- **Market-driven innovations** such as personalized offers, anticipated business needs, real-time alerts, and predictive maintenance

- Deployment of **new data** technologies **alongside legacy** technologies.

- **Organizations that don't renovate** their data architectures **won't be able to meet their goals** in terms of regulations, customers, and market.

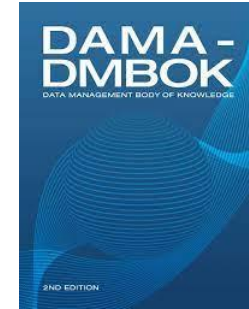# Current State of "data" and "information"

- New technologies have **increased the complexity of data architectures** enormously.

Ancient to **modern**

What's Data Architecture (DA)

- **Data strategy specifications** that outline the current state, describe data requirements, direct data integration and manage data assets.


- **Processes to capture, transform, and deliver usable data to business users**.

- Structure of an organization's **logical and physical data assets** and **data management resources**.

"DA" vs Information Architecture (IA)

- An **information architecture** defines the architecture which **enables the business operations** and realizes business user needs.

- The **building blocks** of an information architecture are the **organization's applications, business processes and data flows.**
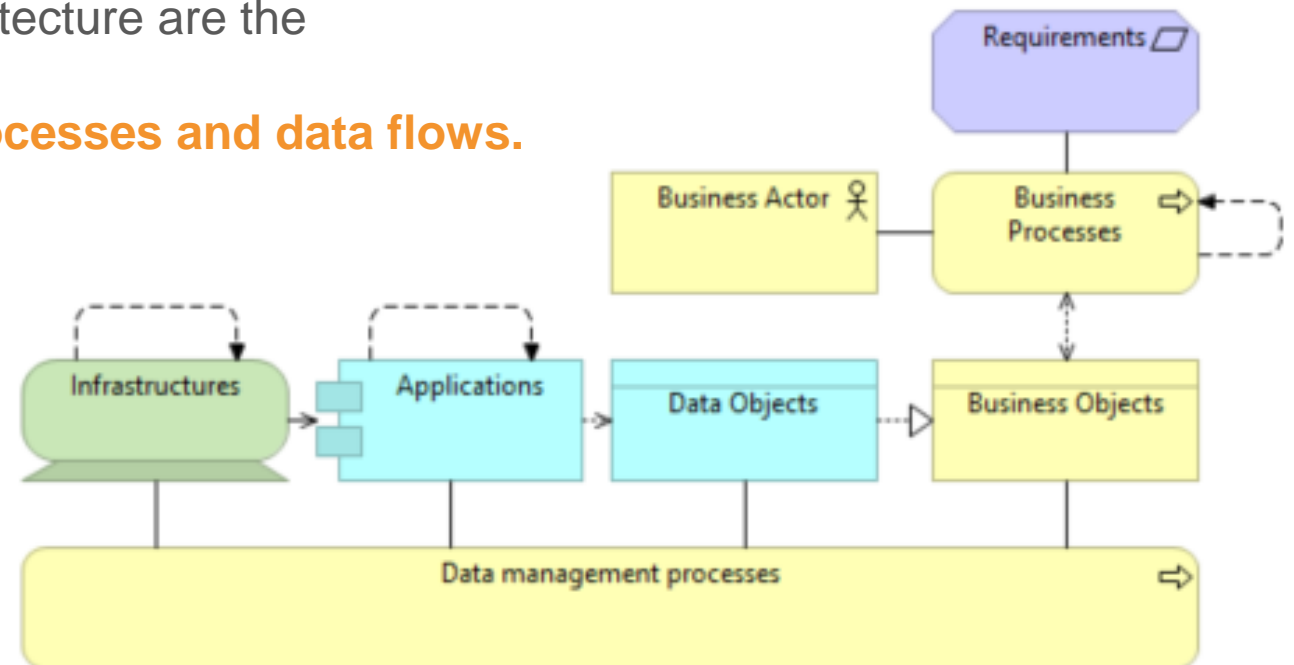
"DA" vs Information Architecture (IA)

- The **data architecture provides a foundation for the information architecture.**

- Data architecture includes **file systems and databases** that span cloud, on-premises and all other infrastructures and technologies.

- The basic **building blocks** of a **data architecture** are a **data warehouse**, **staging data** stores and **business intelligence** schemas.

- More sophisticated data architectures may add a **data lake**, **analytics sandboxes**, a **data science hub** and **operational data stores**.
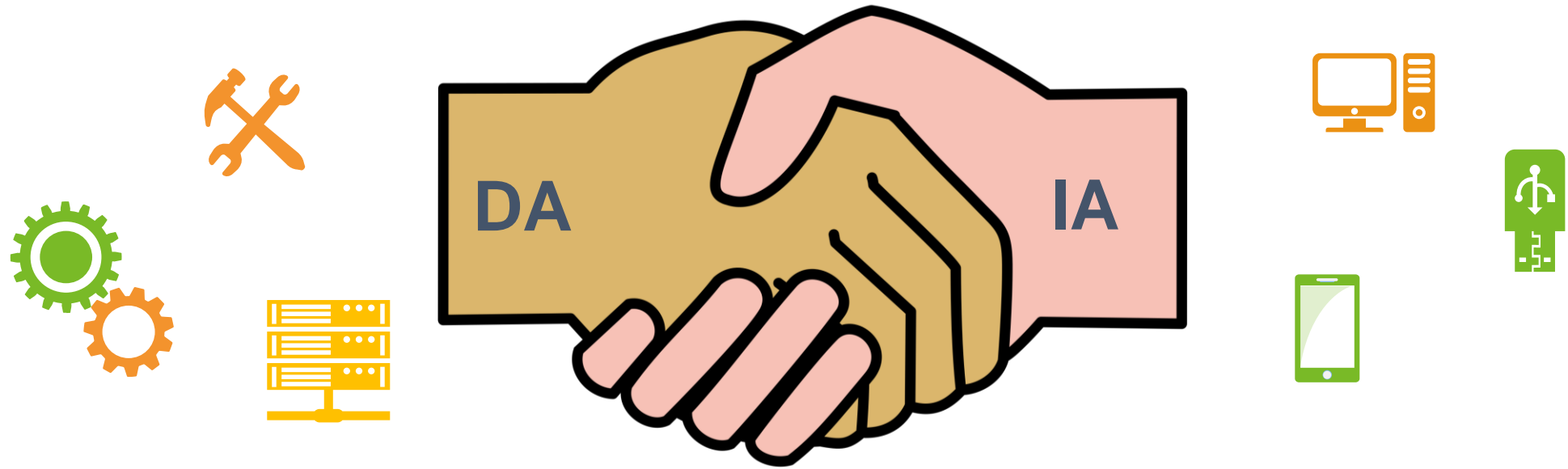
- **From Ancient to Modern DA**

- **Modern DA**

- **Snowflake Reference Architecture**

- **Eckerson Group Reference Architecture**

- **McKinsey Reference Architecture**

- **What Does a "Good" DA provide?**

MODERN DATA REFERENCE ARCHITECTURE

From Ancient to Modern DA

- Prominence of **warehouses** for many years

- Warehouses **hardly responded** to the **constant changes** in the business environment.

- Organizations also use **data lakes to store raw data**.

- Data lakes require large storage capacities, firms can **analyze** the data for **any purpose**.

- **Lack of efficient data governance** strategies has **plagued this resource**.
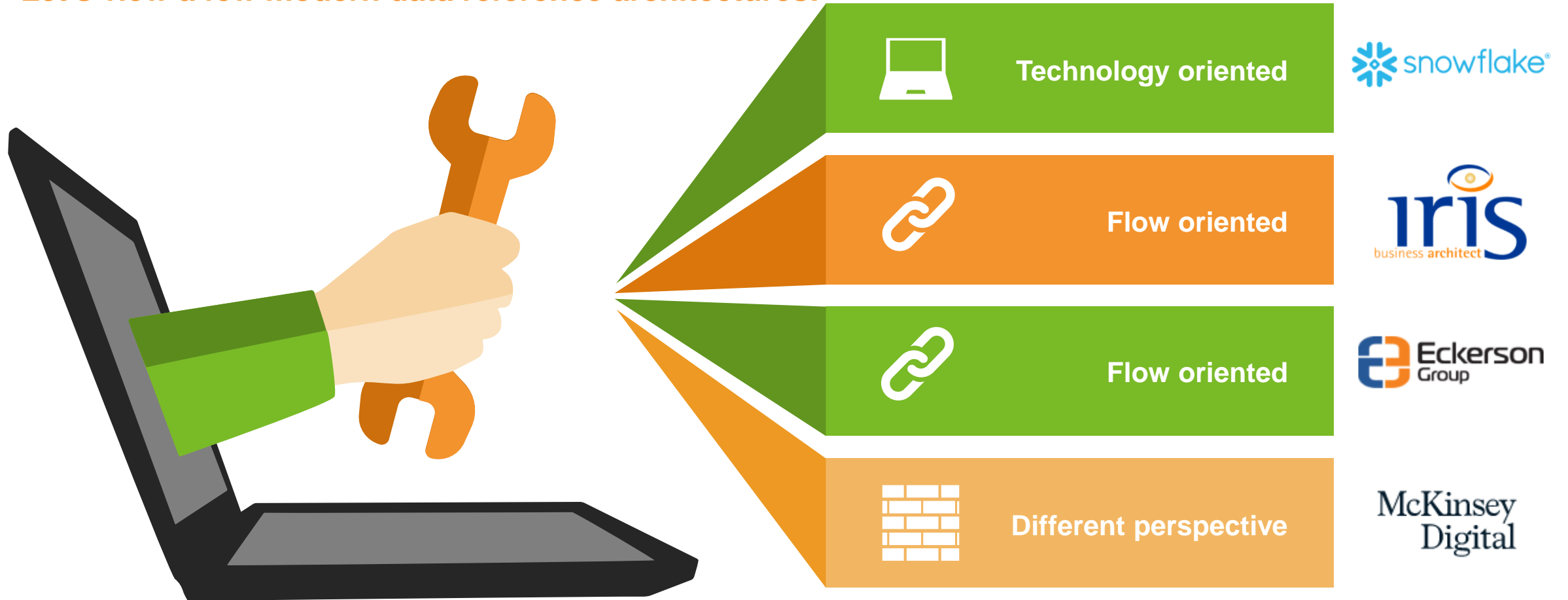
Modern DA

Still needs old days!

Old school or hits?

- **Organizations are not limiting themselves to warehouses as they take too many resources to implement and change.**

- **Agility is the heart of a modern data architecture.**

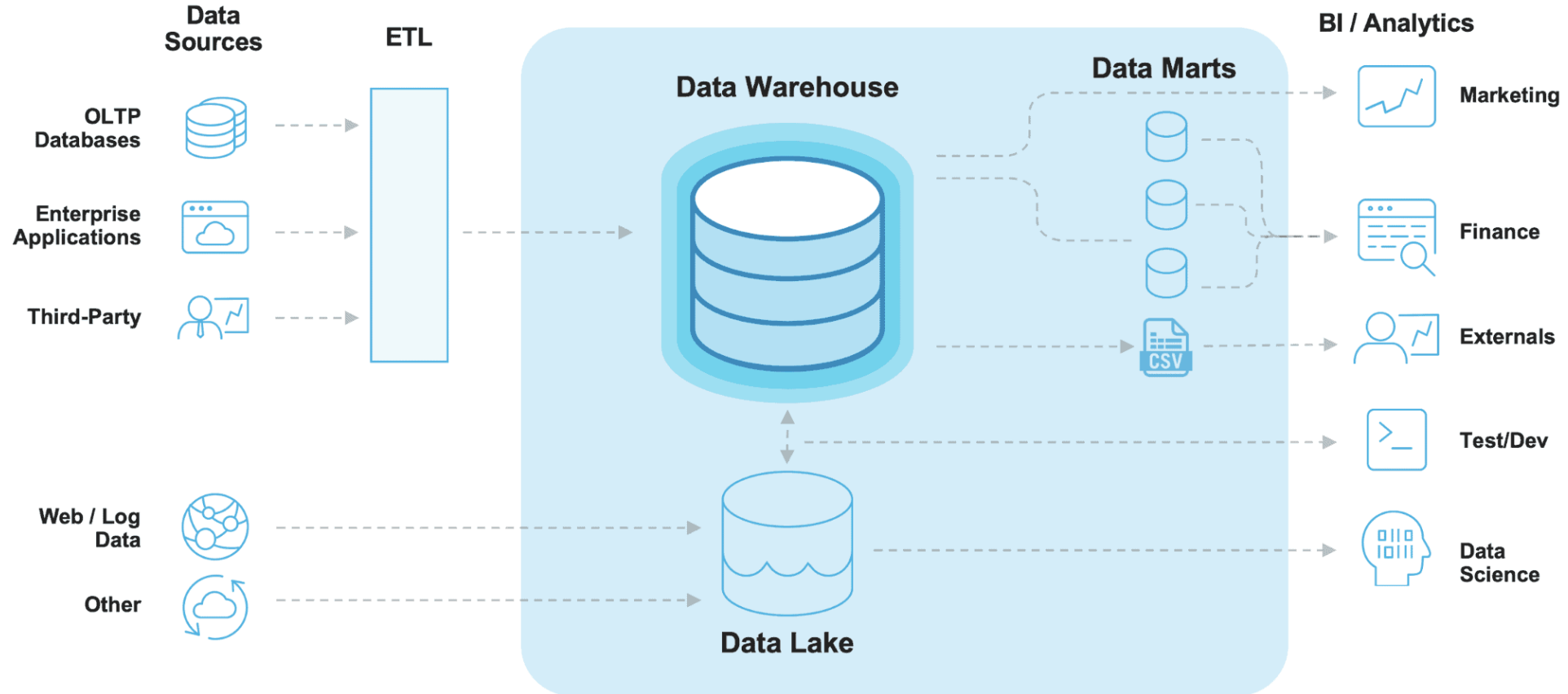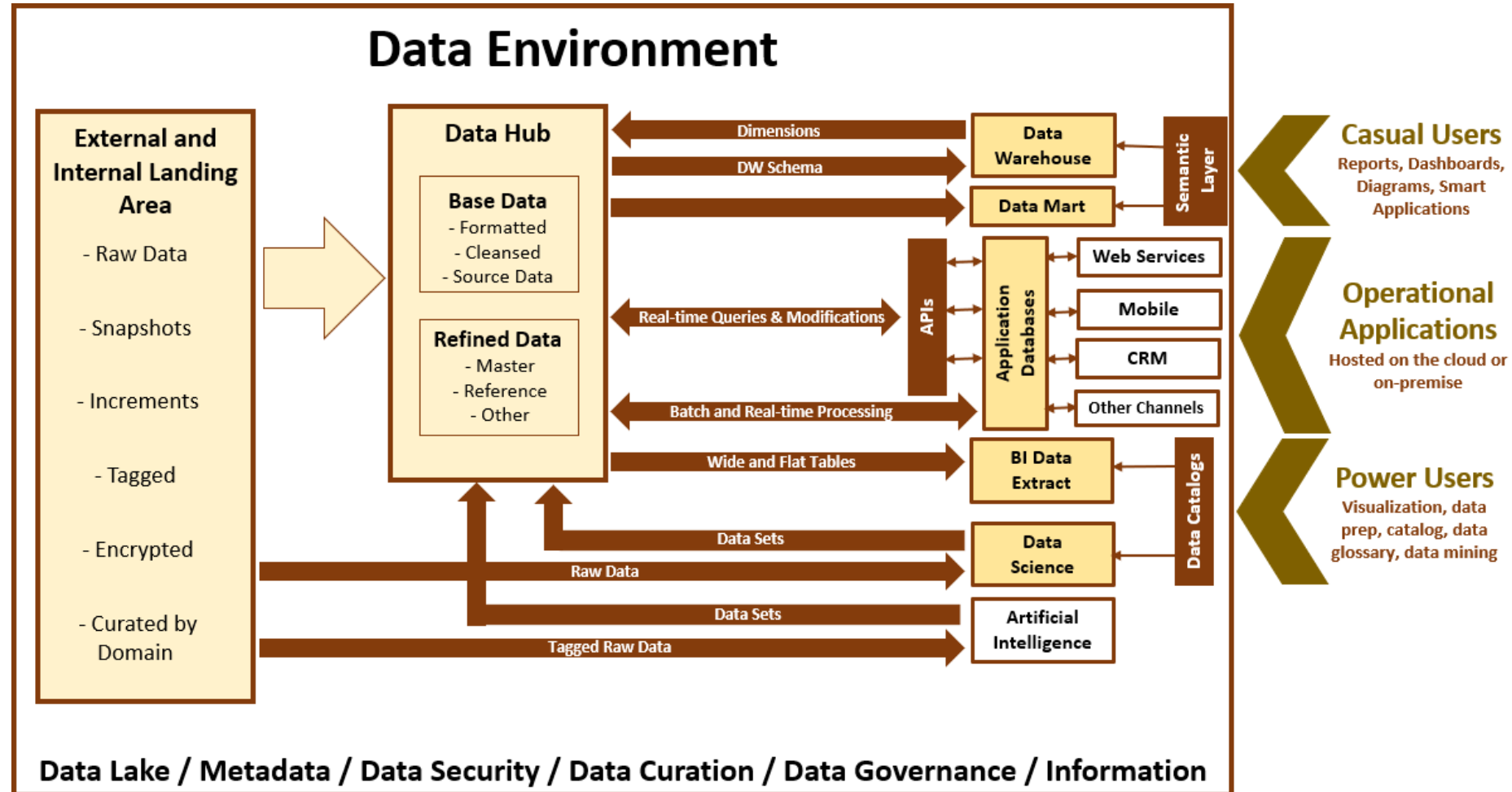- **Data warehouse and data marts, are not to throw away! Just need to be more agile.**

Modern DA

Modern DA

Modern DA

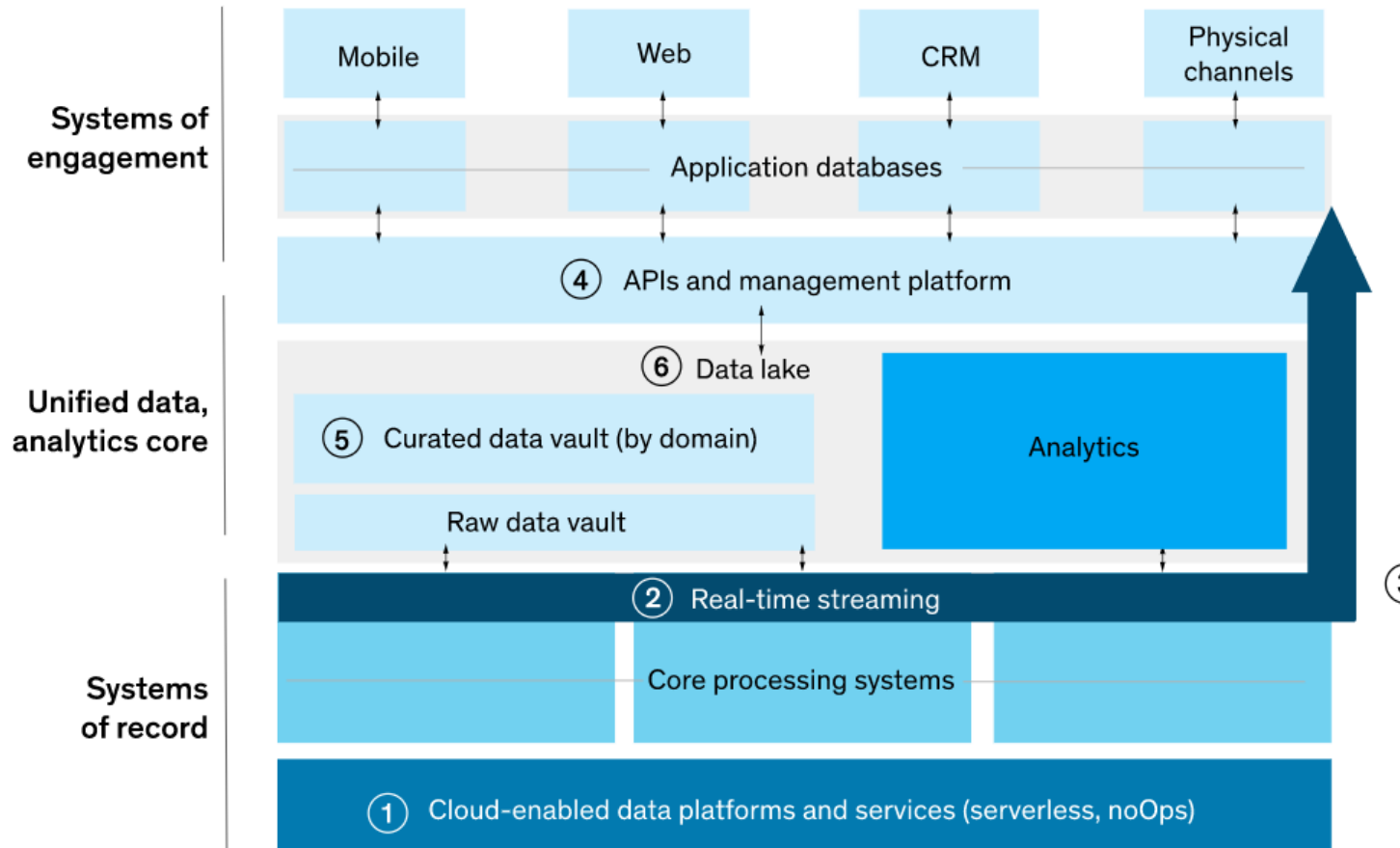# Modern DA

- **Cloud-based and Multiple Cloud Data Platforms**

- **Real-time Data Processing**

- **Domain-based Architecture**

- **Modular, Best-of-breed Platforms**

- **Decoupled Data Access**

- **Flexible, Extensible Data Schemas**

TOP TRENDS SHAPING DA

- **Self-service Analytics**

- **Industry 4.0**

- **Digital Transformation**

- **Data as an Asset**

TOP TRENDS SHAPING DA

Cloud-based and Multiple Cloud Data Platforms

- Probably the **most disruptive driver** of a data-architecture

- **Infinite scale** in server-less data platforms, such as Amazon S3 and Google BigQuery.

- **Decreased expertise** requirements

- **Faster deployment**, from several weeks to as little as a few minutes

- **Less** operational **overhead**

Cloud-based and Multiple Cloud Data Platforms

- **Well suited** for modern **containerized** data solutions and **VMs** as well

Ancient   **Near past**   **Now**
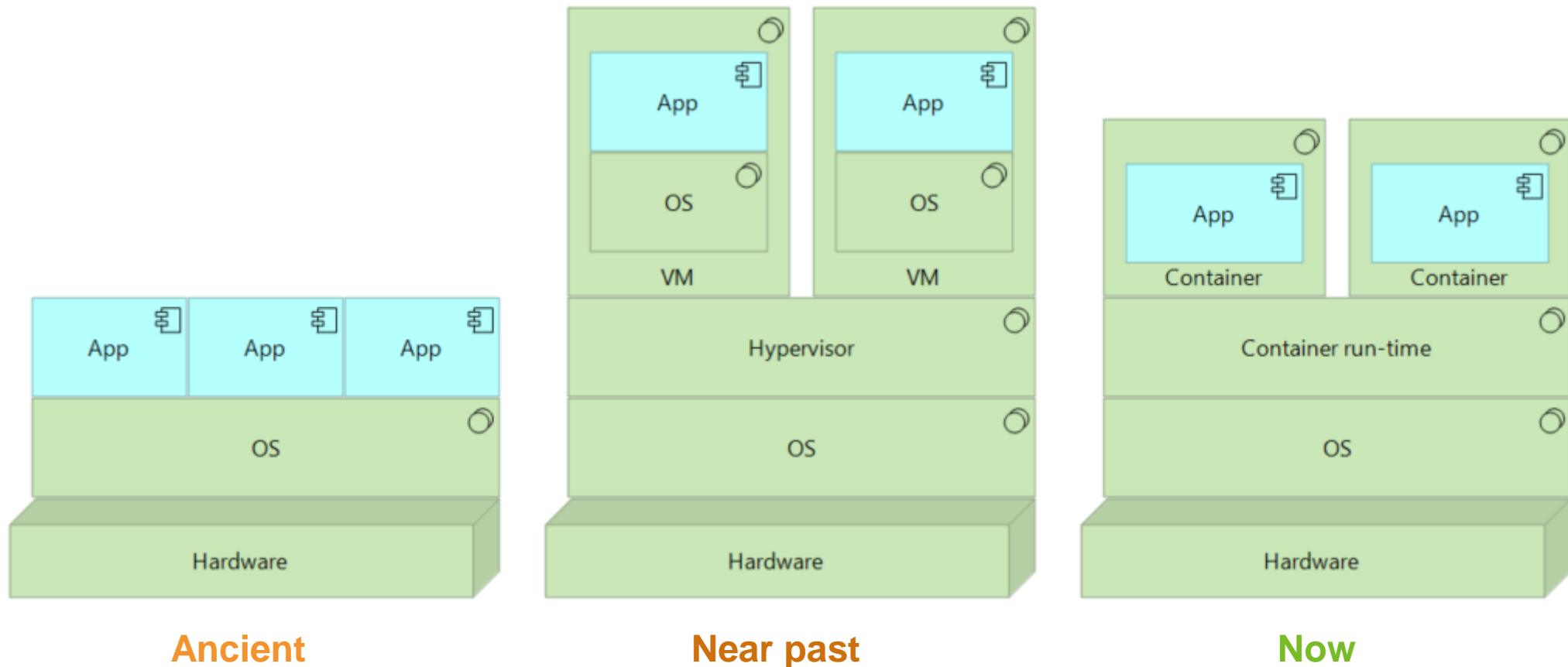
Cloud-based and Multiple Cloud Data Platforms

- **Modern DA building block: Cloud Manager**

  - **Data flow management, WS integration, Cloud accounting, Quality management**
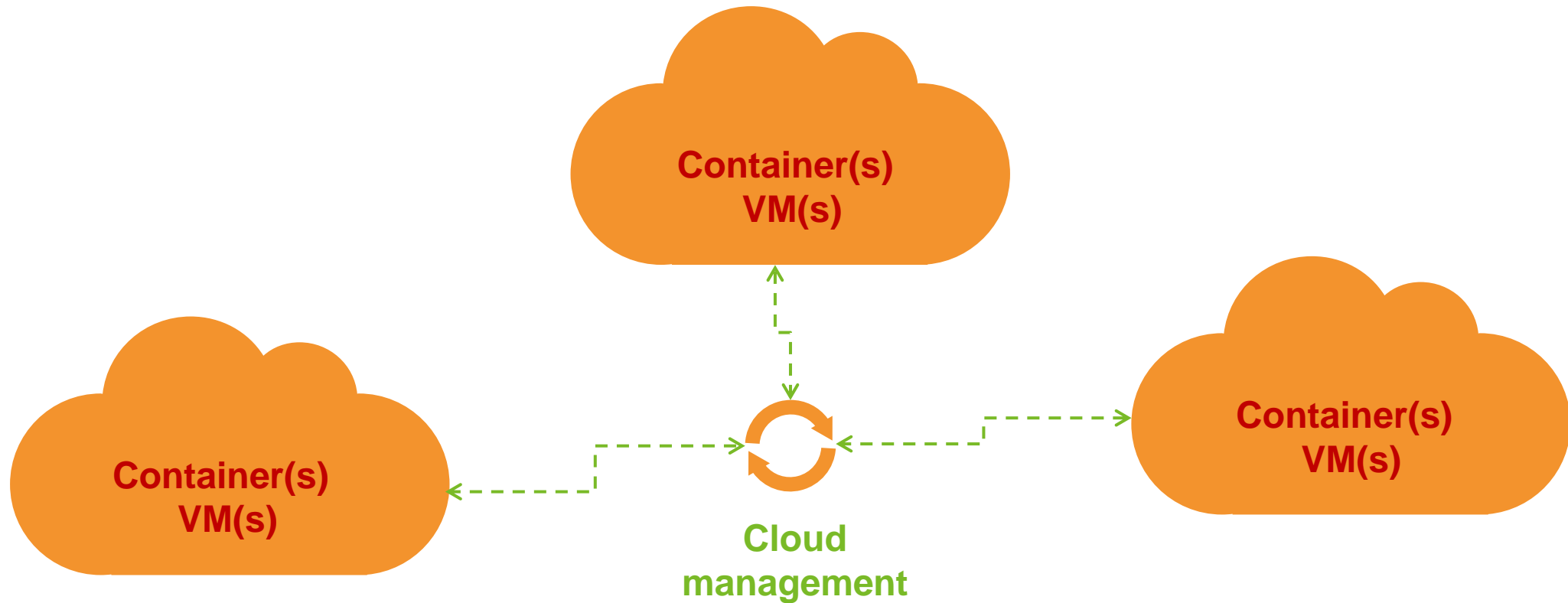
Real-time Data Processing

- **Decreased costs of real-time** data processing

- The whole idea is about having a **capability to react** in the moment and change the outcome **while there's still time**.

- A **data lake** typically serves as the "brain" for such services

- **Data at transit** (motion)

Data at Rest vs. Data In-Transit

USB & Local Drives

Servers & Databases

Web Downloads & Uploads

Wifi & Mobile Networks

Personal Devices

Data at Rest

Data In-Transit

LAN File Transfers

Cloud Data Storage

Archive & Physical Storage

Peer-to-Peer File Sharing

Cloud Data Sync

SPANNING
A Kaseya COMPANY
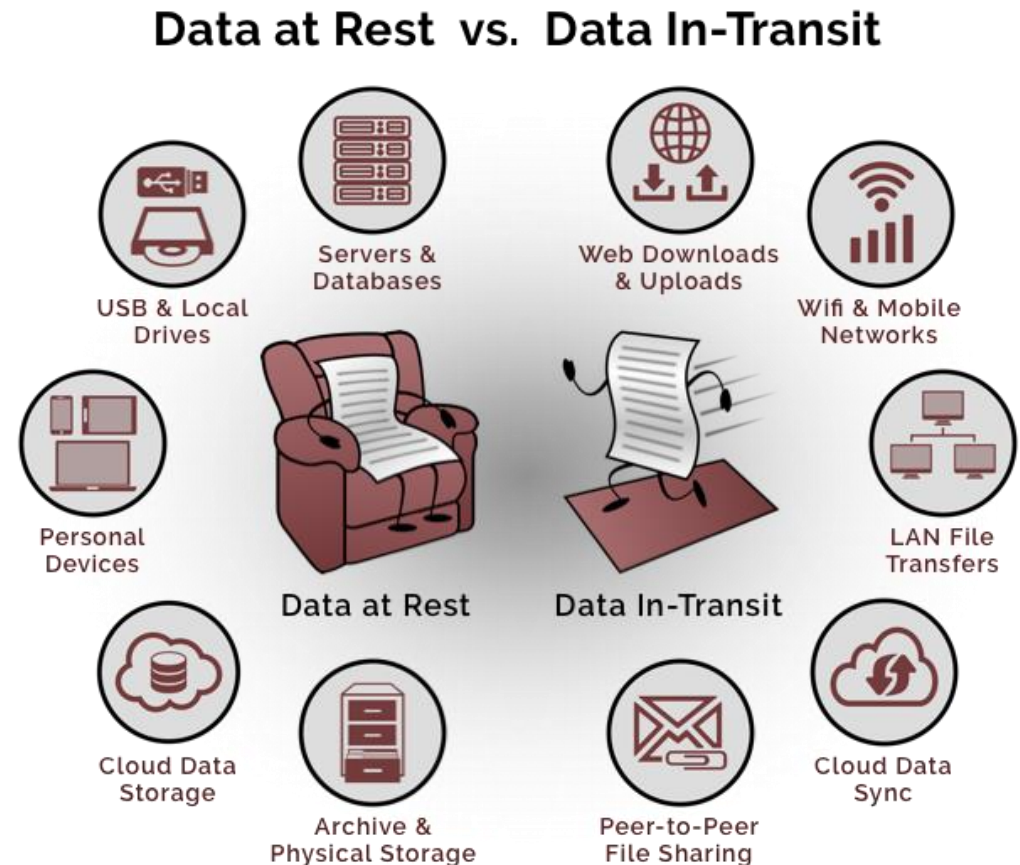
Real-time Data Processing

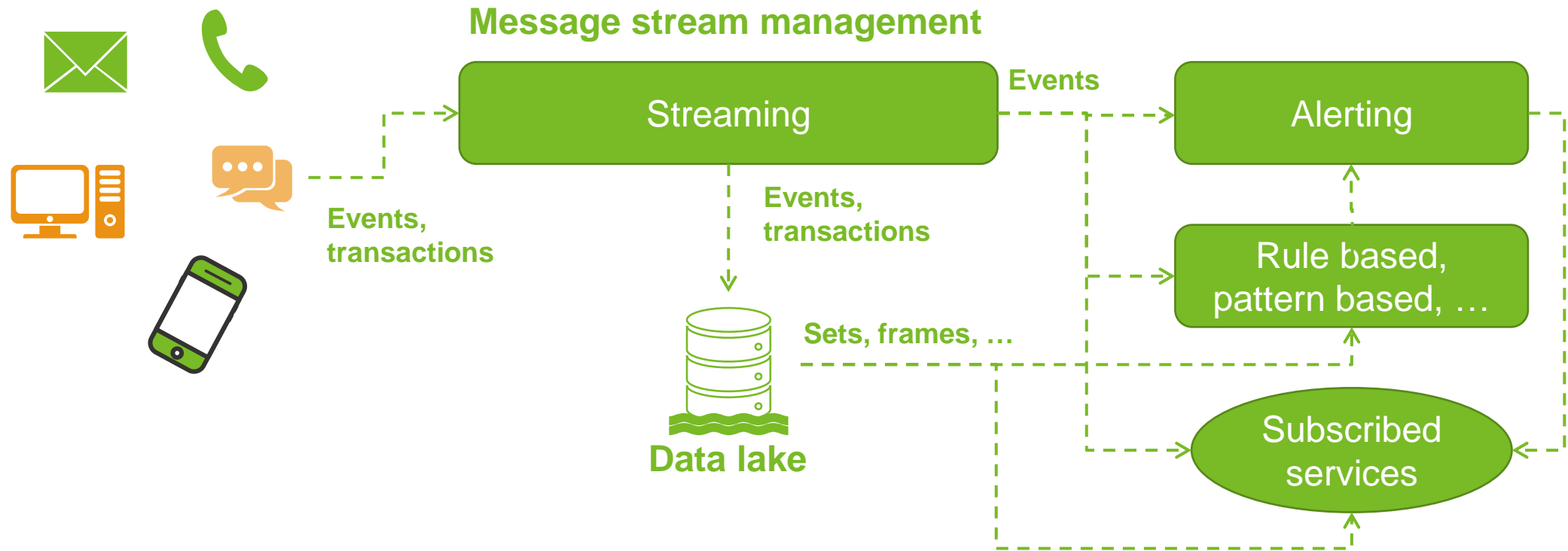- **Publish/subscribe** pattern

- **Messaging platforms** process and store **millions of messages every second**

- **Real-time analysis of messages** (streaming and analytics)

  - rule based

  - advanced analytics to extract events or signals from the data

- **Alerting platforms**

  - notifying

  - processes that may run in ERP or CRM systems.

Real-time Data Processing

- **Modern DA building block: Message stream management**

Message stream management

Streaming → Events → Alerting

Events, transactions

Events, transactions

Data lake

Sets, frames, …

Rule based, pattern based, …

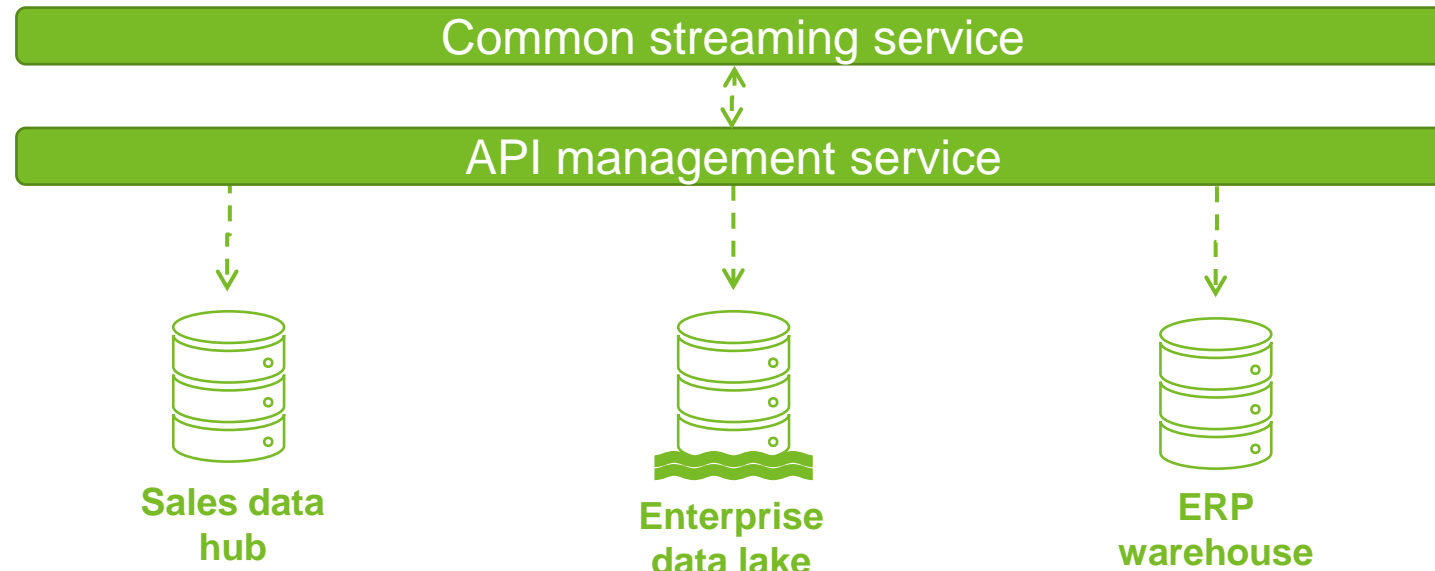Subscribed services

Domain-based Architecture

- From a central enterprise data lake **toward "domain-driven" designs**

- Can be customized and **"fit for purpose"**

- Requires a careful balance to **avoid becoming fragmented** and inefficient

- **Reduces** the **time spent on building new data models** into the lake

- **Fits federated** business **structures**

Domain-based Architecture

- Data sets may still reside on the **same physical platform**

- **Data infrastructure as a platform**

  - **common tools for storage** management

  - **speeds** implementation

  - data producers **won't need to build** their own **data-asset platform**

- **Data cataloging** tools provide:

  - **search** and **exploration of data** without requiring full access or preparation

  - **metadata** definitions

  - **interfaces** to simplify **access to data** assets.

Domain-based Architecture

Data architecture building blocks

Common streaming service

API management service

Sales data hub

Enterprise data lake

ERP warehouse

Data management platform

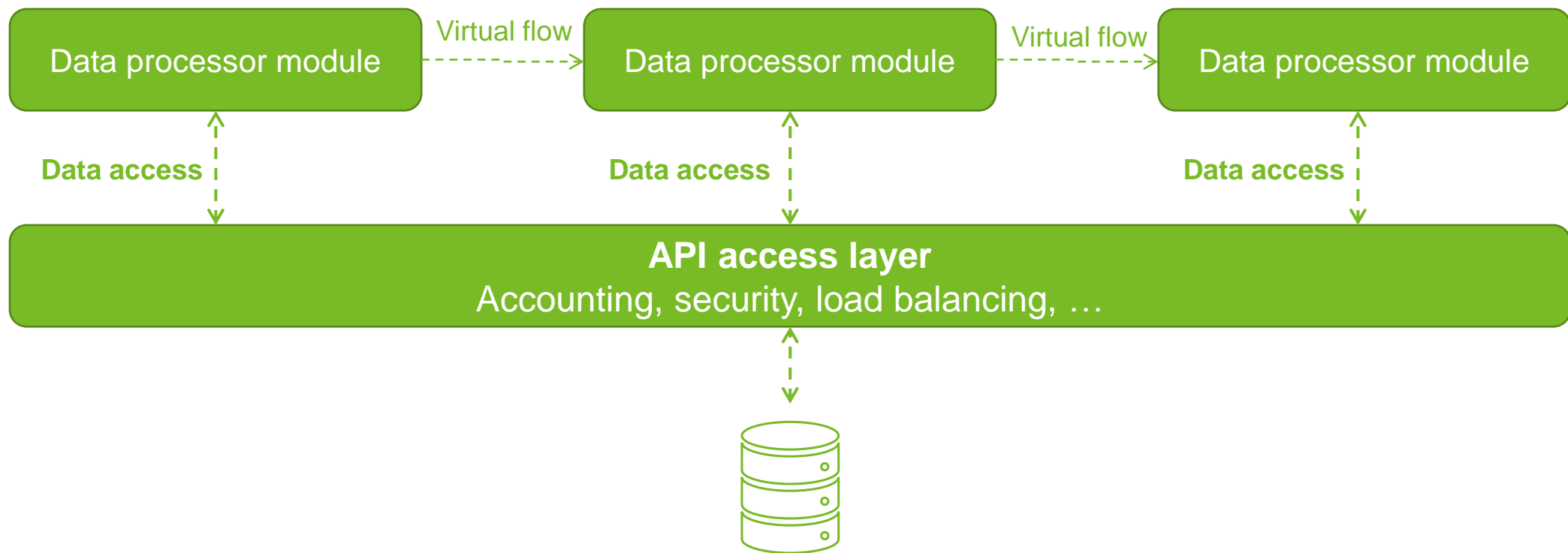|		
MDM	File system	Messaging tool
API management	Analytics platform	ETL/ELT tool

Modular, Best-of-breed Platforms

- **Modular data architecture** for scaling up

- Best-of-breed components that can be **replaced** with new technologies as needed **without affecting other parts** of the data architecture.

- **Data pipeline and API-based interfaces simplify integration** between different tools and platforms
  - **Speeds** time to market and **reduces** problems in existing applications

- **Enables Data-as-a-Service** concept and exposing data **via APIs**

  - ensures **secure** view and modify data

  - **faster** access to up-to-date data sets

  - easy **reuse** of data

Decoupled Data Access

- A **data platform to "buffer"** data outside of core systems is often required

    - **could be a central** data platform such as a data lake

    - **or a composition** of data lakes, data warehouses, and so on, based on business domains needs

Decoupled Data Access

- **Modern DA building block: API Access Layer**

- **Modern DA pattern: Data pipelines**

| Data processor module | → Virtual flow → | Data processor module | → Virtual flow → | Data processor module |

Data access ⇅   **Data access** ⇅   **Data access** ⇅

API access layer
Accounting, security, load balancing, …

Data hubs, Data warehouses, Data lakes, Domain specific data stores

Flexible, Extensible Data Schemas

- **Conventional** database or warehouse data models are often **highly normalized** with rigid database tables and data elements to minimize redundancy

- It **fits reporting** and **regulatory** needs but **weakens** organization ability to incorporate **new data elements** or **data sources**.

- **Shema-light approaches** enable **360 degree view** of any subject

- **Shema-light approaches** use **de-normalized** data models, with have **fewer data tables**

  - **agile** data exploration

  - **flexibility** in **storing** structured and unstructured data

  - **reduced complexity** (no need to introduce additional abstraction layers)

  - **unstructured data** are not only to store, but to be analyzed and processes

NoSQL

- **NoSQL** database is a non-relational Database Management System (DBMS)

  - does **not** require a **fixed/schema**

  - fits **distributed data stores** with **BASE** features; **ACID** concept can be **sacrificed**

  - offers **heterogeneous** structures of **data** in the same domain

  - **schema-free or relaxed schema**

  - very high **scalability**

  - fits **real-time** processing

  - processing **unstructured data**

SQL ACID

Atomicity

Consistency

Isolation

Durability

SQL

A   **No partial transactions**
All or no part of a transaction must be executed

C   **No integrity rules breaks**
Each transaction must make changes that preserve integrity of data.
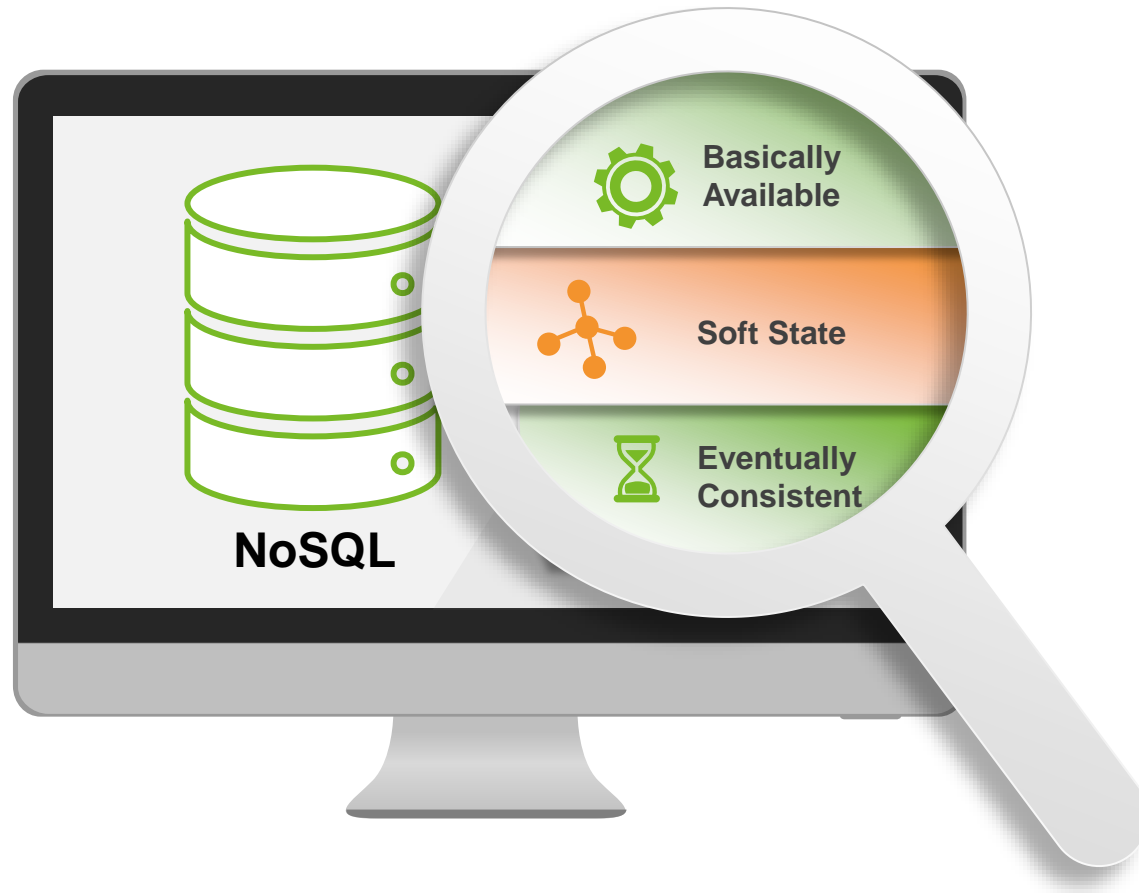
I   **No simultaneous operations**
Operation on the database can begin only when the previous operation is done.

D   **No temporary results**
When a transaction is committed, results must be permanent.

NoSQL BASE

NoSQL

- Basically Available
- Soft State
- Eventually Consistent

BA

No partial transactions

DB is available all the time as per CAP theorem.

S

No integrity rules breaks
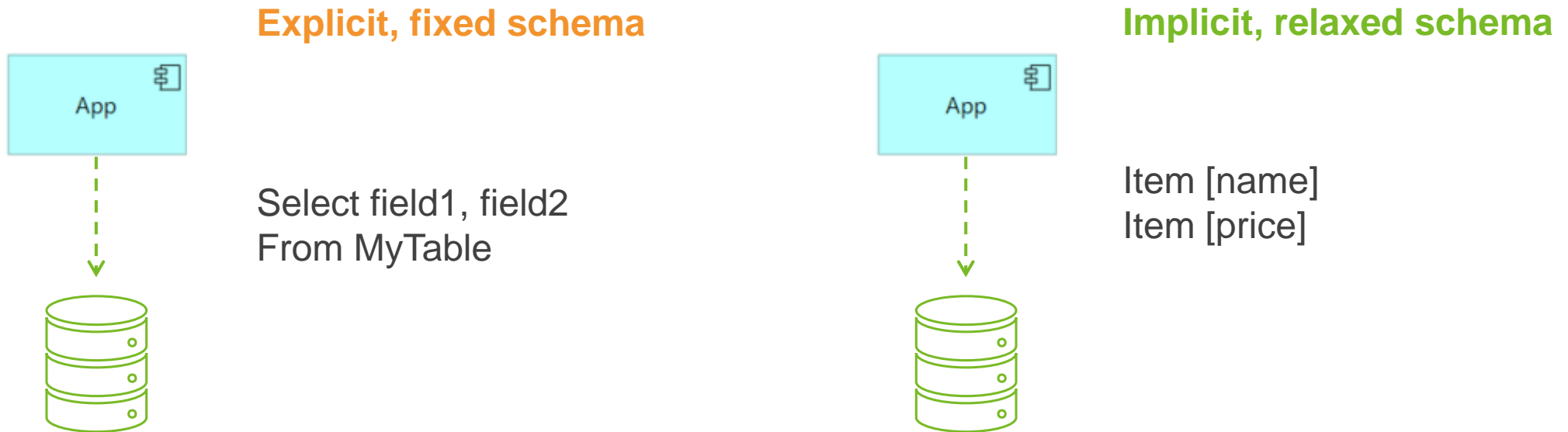
without an input; the system state may change.

E

No simultaneous operations

system will become consistent over time
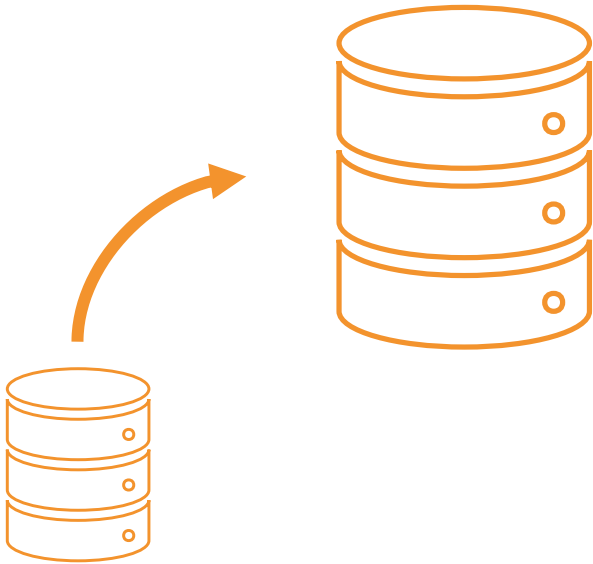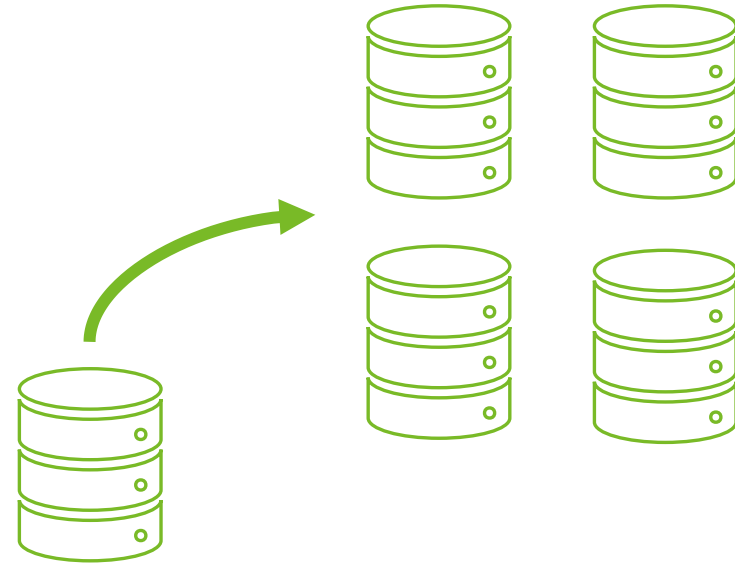
SQL vs NoSQL

- **Schema-free feature**

Explicit, fixed schema

App

Select field1, field2
From MyTable

Implicit, relaxed schema

App

Item [name]
Item [price]
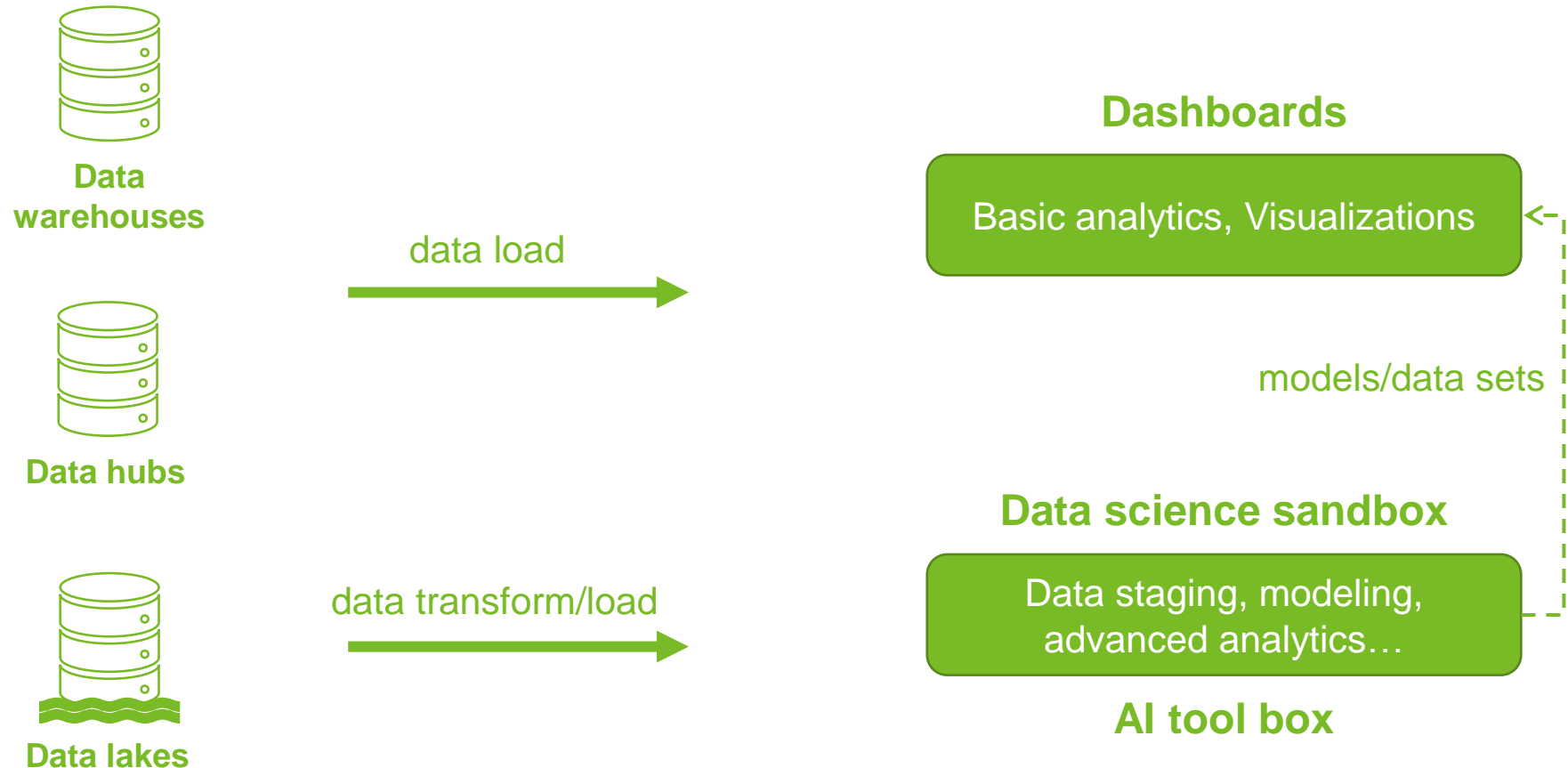
Self-service Analytics

- **Self-service analytics**: provide intelligence to the right individuals at the right time.

  - needs **free flow** of information

  - removing **bottlenecks**

  - eliminating **non-value-added tasks** in the information value chain

  - providing tools and training to all stakeholders to ultimately

Self-service Analytics

Data
warehouses

Data hubs

Data lakes

data load

data transform/load

Dashboards

Basic analytics, Visualizations

models/data sets

Data science sandbox

Data staging, modeling, advanced analytics…

AI tool box

Data warehouse

- **Data warehouses** is a data store that gathers data from a wide range of sources within a company and uses the data to support management decision-making

- still a good way of **enterprise data integration**

- Need to **interoperate with data lakes**, **master data repositories**, and **analytic sandboxes** in a fast, scalable and agile fashion

Data lake

- Data lake can store **all forms of data**.

- It can hold a **vast amount** of raw **data** in its original format.

- **Centralized repository** for data collected from any data producer

- **Extension of EDW**

Data hub

- A **hub-and-spoke** approach to store data

- **Data** is physically **moved** and **re-indexed**

- This provides **more structure to the data** and permits diverse business users to access information that they need **more rapidly than in a data lake**.

Data hub, warehouse, lake!

Data hub	Data warehouse	Data lake
Structured master/reference data	Structured data	All types of data
Multiple sources	Multiple sources	All sources
Data integration	Predefined and repeatable analytics patterns	Massive data storing and analytics
Trusted data	Quality-checked data	Unrefined data
Fixed schema	Fixed schema	Relaxed schema
Pillar for data governance enforcement rules	After the fact governance	Lightly governed
ETL (Extract Transform Load)	ETL (Extract Transform Load)	ELT (Extract Load Transform)
Urgent needs	Urgent needs	Less/no urgent needs

Industry 4.0

- Talking **industry 4.0**

    - **data streaming, data in motion (data in transit)**

    - **Internet of Things (IoT)**
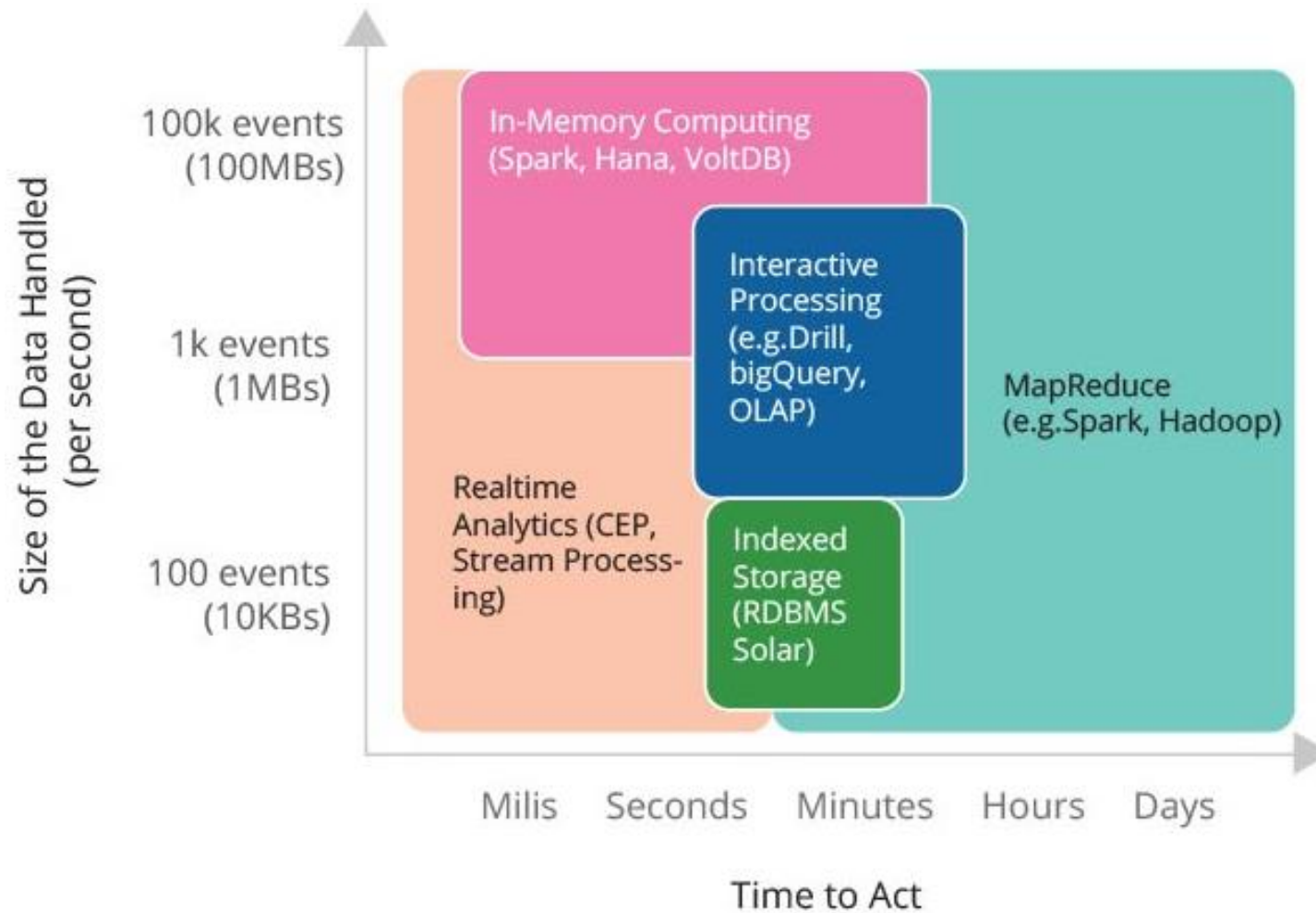
    - **big data**

IoT

- **Data flow** in **IoT** architecture:

  - **Sensors** collect and transfer data **to** a **gateway**.

  - **Gateway** send them **to** a **processing system**.

  - **Gateway** can choose either **to or not summarize** or preprocess the data.

IoT

- The **data architecture** and technology **depends** so much on the **size** of data and **time** to **respond**:

  - A few **hours**: send your data into a **data lake**

  - A few **seconds**: send data into a **stream processing system**

  - A few **milliseconds**: send data to a system like **complex event processing** where records are

    processed one by one and produce very fast outputs.

IoT

Big data

- The **4+3 Vs**:

  - **Volume**: Zettabytes, Exabytes, and Yottabytes

  - **Variety**: structured, semi-structured, unstructured

  - **Velocity**: how fast the data can be processed and accessed

  - **Variability**: data which keeps on changing constantly

  - +

  - **Veracity**: is the data "true"?

  - **Visualization**: data must visualized in to a readable, usable and useful visuals

  - **Value**: turn data into value

Digital Transformation

- **Customer Experience** will remain the **focal point of digital transformation**

  - needs **360 degree view of customer**

  - seamless experience

- Enabled by **Flexible, Extensible Data Schemas**

Data as an asset

- Assets **need to be managed**

  - Data governance frameworks
  - Data lineage tools
  - Data catalogues
  - MDM
  - ETL/ELT tools

  - Messaging tools (bus)
  - API management
  - Analytics platform
  - DBMS
  - File system

Other tools may be needed for monetization and so forth

Best tools worth nothing with a vigorous data management governance practice

Technical tools

- **Key Actions**

- **Modern Architects**

HOW TO REVOLUTIONIZE DA

Key actions to revolutionize DA

- Data **technologies** are **evolving quickly**.

- **Traditional approach** (define and build toward three-to-five-year target architecture) is **risky**.

- **Data and technology leaders** will be best served by **practices** that enable them to rapidly evaluate and deploy new technologies so they can **quickly adapt**.

Key actions to revolutionize DA

- Promote **Data-Ops**

  - agile practices, **build and refine** approach for instance, have been applied in application

    development for quite a while and have recently moved into the data space.

- Establish **data quick reaction** teams

  - squads of data stewards, data engineers, and data architects work together with end-to-end

    accountability for building the data architecture.

- Cultivate proper organizational culture

  - training and talking and making brochures **is not enough**!

  - tie data strategy to business goals/needs and **enforce it. Proper culture will soon appear!**

Key actions to revolutionize DA

- Get **buy-in** from **C-suit**

  - **value** of developing a data architecture must be **communicated** to C-suite executives

- **Establish** an effective data **governance** framework

- **Business** user information **needs** are **Polaris**

- Keep **road-map updated** based on constant SWOT analysis or other analysis methods

- **Don't miss** architectural **debt**

Modern architects

- Better and more **communication with SMEs**

- **Research into clues from legacy systems**

- **Research** into external **forces driving change** in business architecture

- A **data architect AND** a **security** architect!

- Explore **data engineers skills**

- Learn agile concepts and **become agile**

- **Keep in mind:** A good data architecture [process] flows right to left:

  from data consumers to data sources—not the other way (TOGAF) –

  From IA to DA!

Thank You
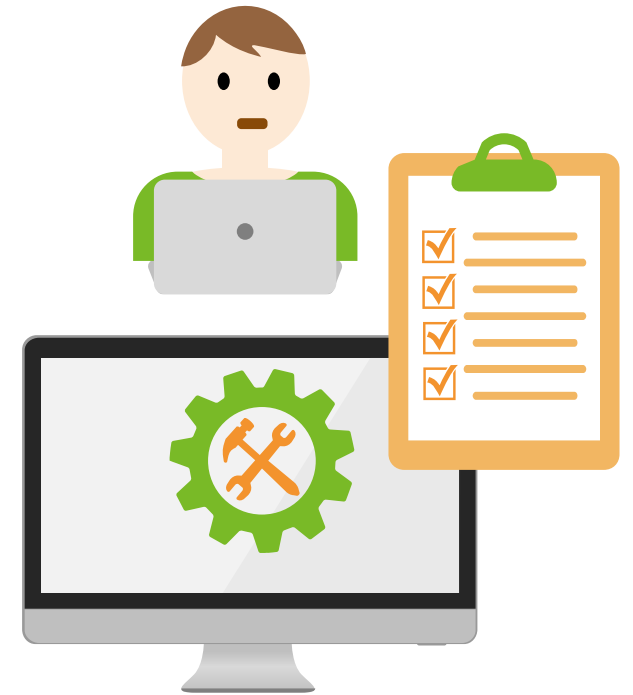
Hamid Reza Aghiri

@ HrAghiri@gmail.com
@ H_Aghiri@isc.co.ir

به گروه تخصصی معماری سازمانی انجمن انفورماتیک ایران بپیوندید...

www.isi-ea.ir