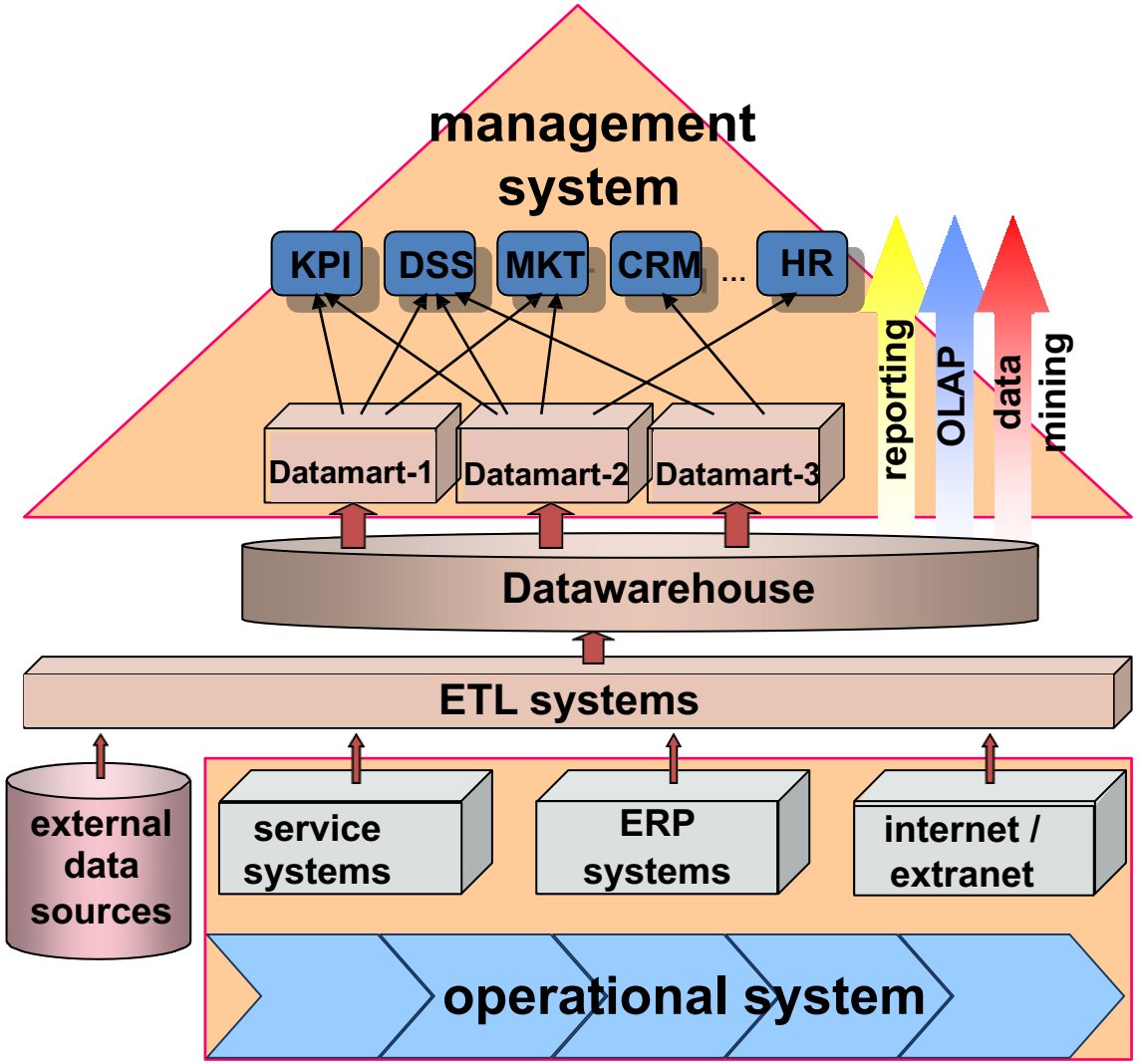
Bachelor Degree Programme in Applied Computer Science and Artificial Intelligence

SAPIENZA
UNIVERSITÀ DI ROMA

12. Introduction to Business Intelligence and Data Warehousing

Prof. Ing. Claudio CILLI
cilli@di.uniroma1.it
http://wwwusers.di.uniroma1.it/~cilli

Architecture for Business Intelligence



What is Data Warehousing

- Collection of methods, technologies and tools to assist the "knowledge worker" (manager, analyst) to conduct data analysis aimed at supporting decision-making and/or improving the management of information assets

What is a Data Warehouse

A data warehouse is a collection of data

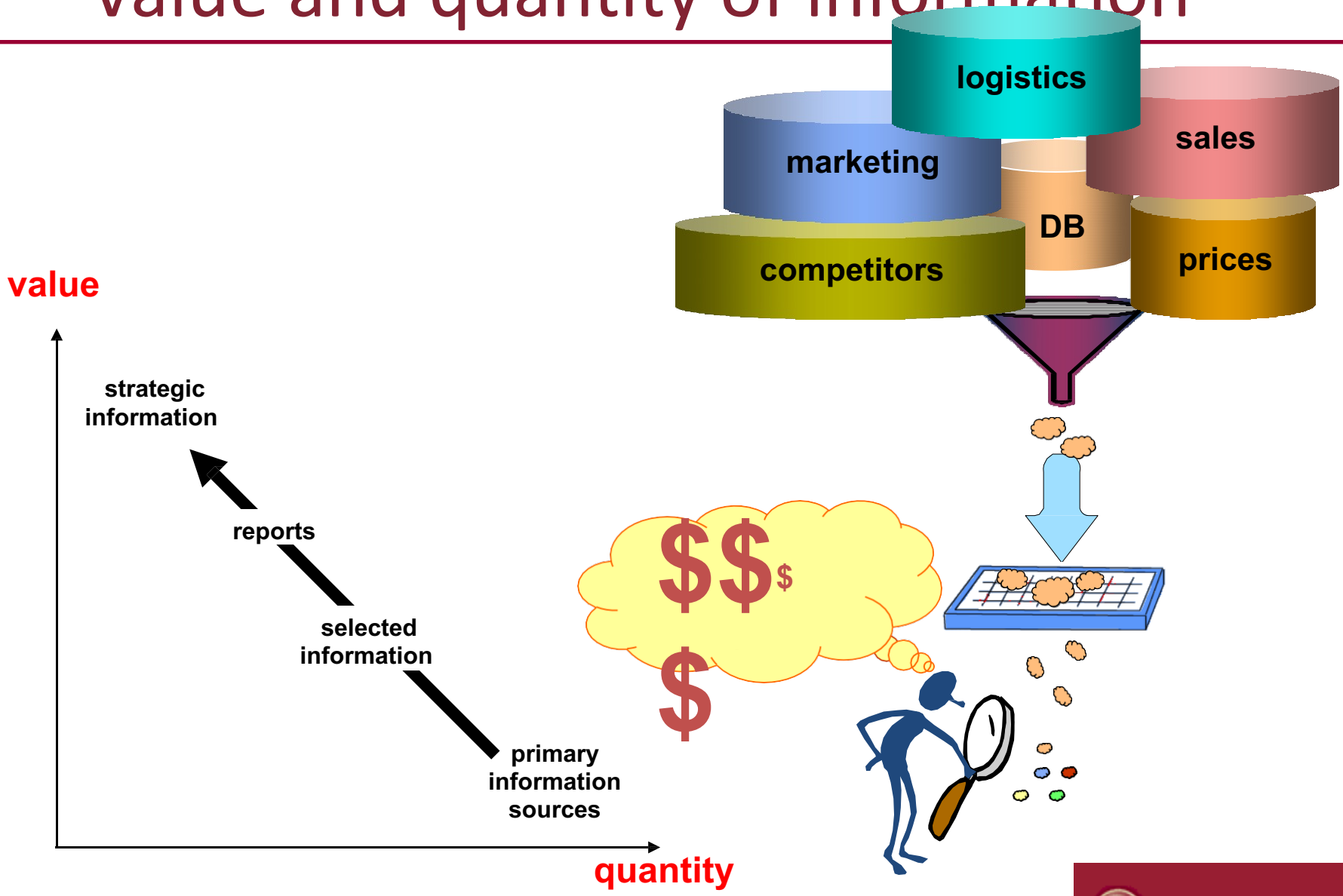
- integrated (far beyond the organization)
- consistent (despite the heterogeneous origin)
- focused (an interest area is defined)
- historical (over a consistent timeframe)
- permanent (never delete your data!)

Purpose of a Data Warehouse

A Data Warehouse helps (allows) you:

- to take decisions
- to identify and interpret phenomena
- to make predictions about the future
- to control a complex system

Value and quantity of information



OLTP & OLAP

OLTP - On-Line Transaction Processing

- realm of (write and / or read) transactions, recovery,
- consistency
- many, fast and frequent operations
- high level of concurrency
- access to a small amount of data
- on-the-fly data update

OLAP - On-Line Analytical Processing

- read only
- few operations
- low level of concurrency
- access to huge amounts of data
- historical but essentially static data

Separation between:

Operational Database & Data Warehouse

- different computational load
- different needs:
  - DB: dynamic data, asynchronous updates
  - DW: static data, periodic updates
- integration with business activity:
  - DB: supporting operations (focused, timely)
  - DW: supporting decisions (descriptive, historical)
- data collection:
  - DB: minimal
  - DW: maximal

Two issues with different perspectives

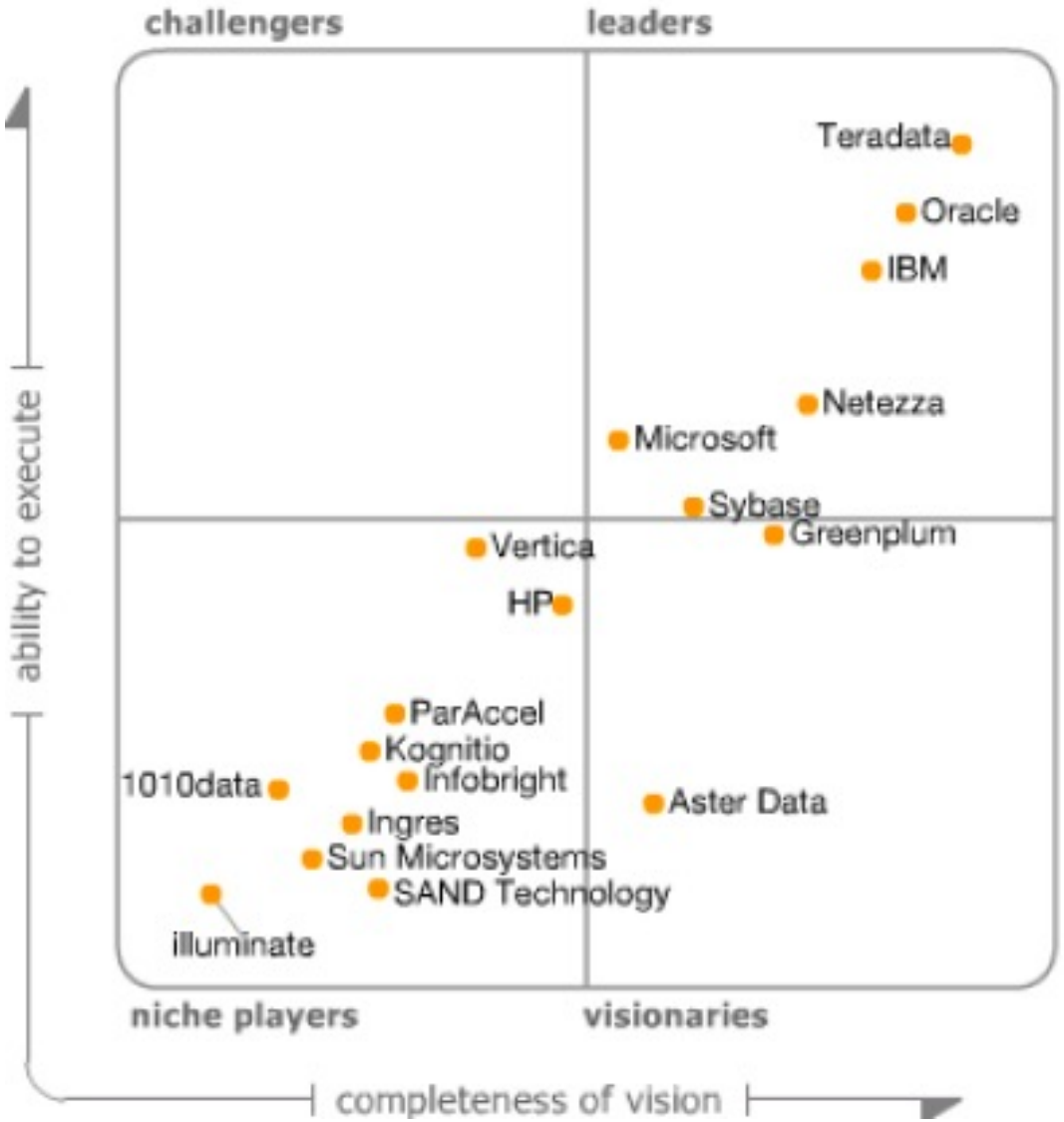
- Data redundancy
  - OLTP (DB): to avoid, bringing to inconsistency and/or inefficiency on updates
  - OLAP (DW): redundancy avoids recomputation and shorten response time
- Indexing
  - OLTP (DB): good when you search – bad when you update... you need some trade-off
  - OLAP (DW): the more, the best

Some Data Warehouse Systems

- Oracle
- IBM InfoSphere
- Microsoft SQL-Server 2014 – Analysis Services
- Sybase IQ
- Hyperion (bought by Oracle)
- Teradata (division of NCR)
- Netezza – Cognos (bought by IBM)
- Business Objects (bought by SAP)
- …

A comparison by Gartner (2013)

2013



Source: Gartner 2013

A comparison by Gartner (2019)

2019

Source: Gartner (February 2019)

Architectures for Datawarehousing: issues

- separating OLTP & OLAP
- scalability
- extensibility
- security
- administrability

Architecture for Datawarehousing

- determined by design choices
- determined by / determines the choice of a software system
- determines the cost and makes possible future integration (quantitative and / or qualitative)
- affects the cost of data processing

Data Mart

Collection of data focused on particular user profile or on particular target analysis

Alternatives:

1. dependent Data Mart: it is a subset and/or an aggregation of data in the primary DW

   → **DM extracted from a DW**

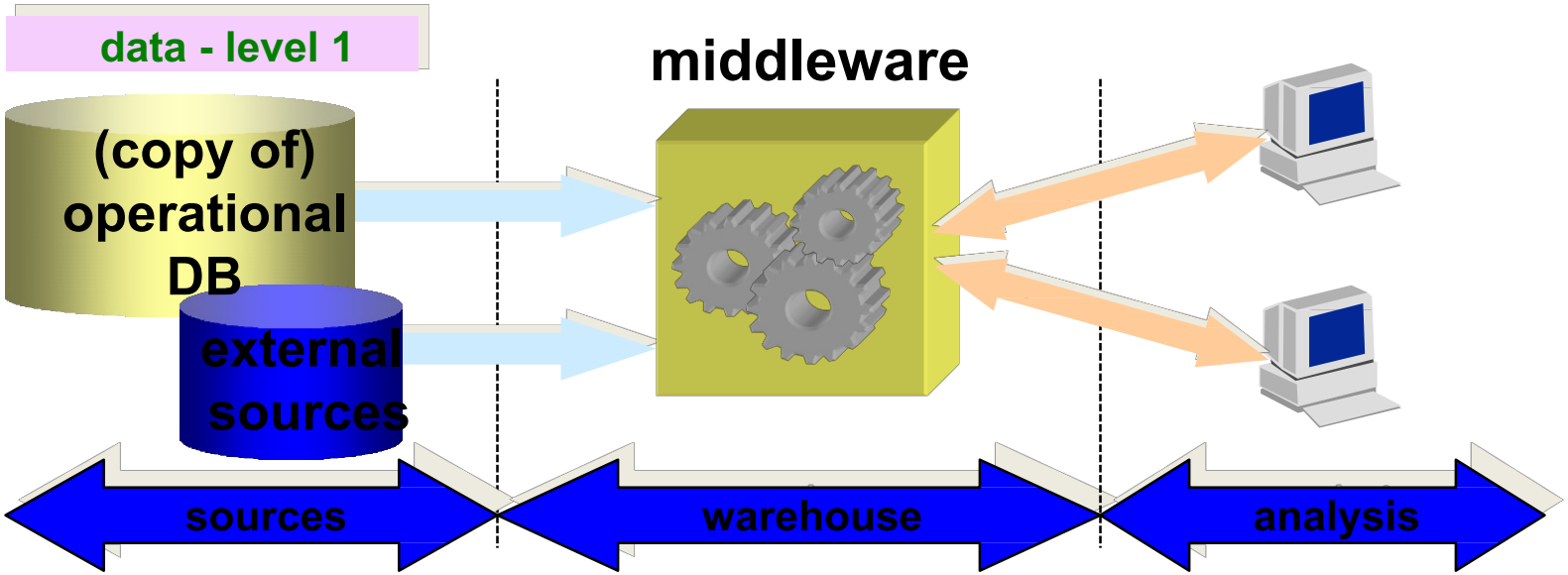
2. independent Data Mart: it is a subset and/or an aggregation of data in the operational DB

   → **DW=Ui(DMi), that is, DW is a set of DM**

3. hybrid solution, combining 1, 2

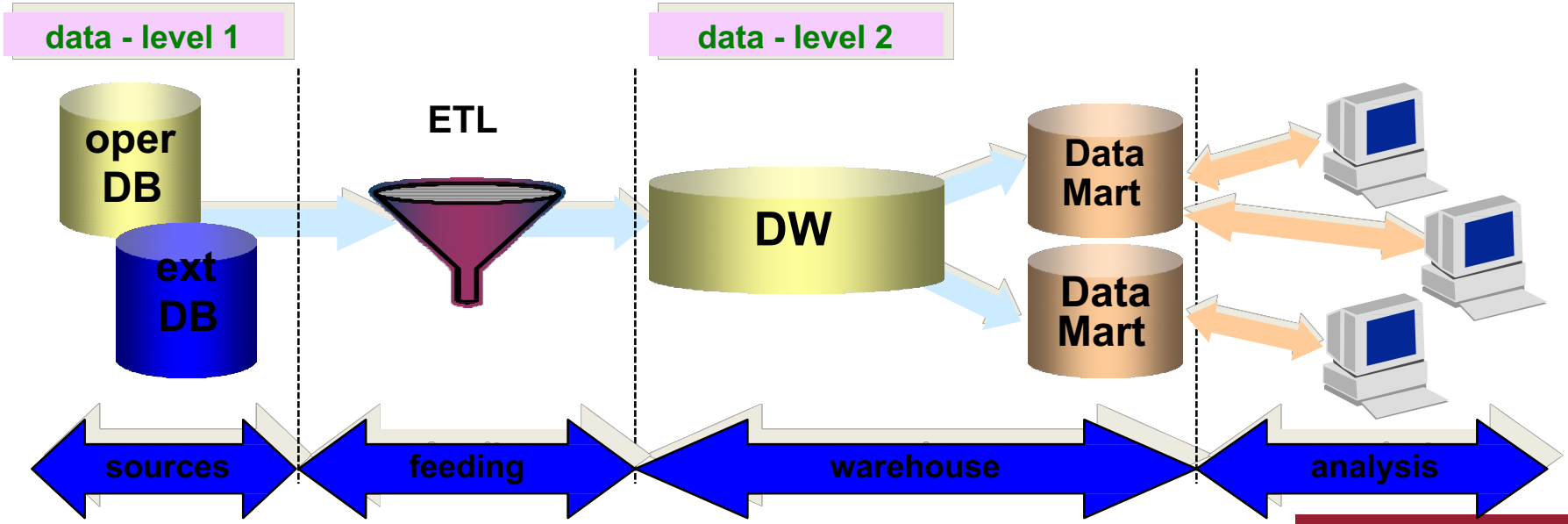
DW architecture: 1 Level

- there is only an operational DW
- virtual DB (no OLTP-OLAP separation)
- data coincident with DB operational
- difficult integration with other sources



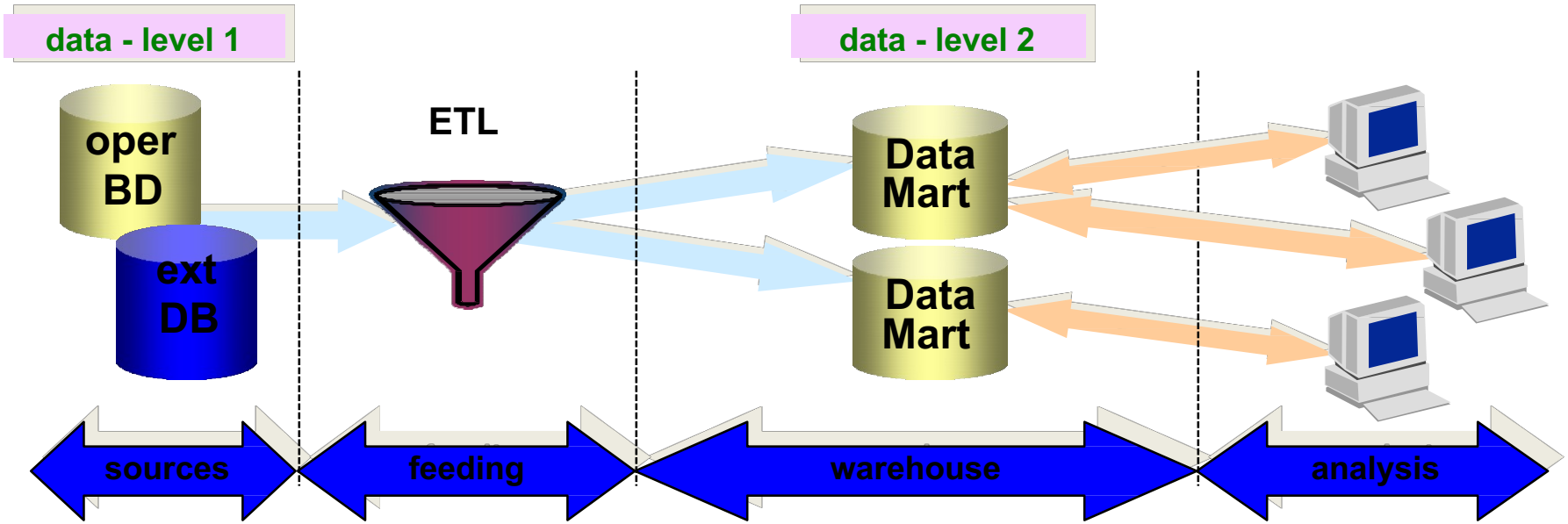
DW architecture: 2 Levels – dependent DMs

- data sources complemented with external sources
- running on dedicated software platform
- ETL: Extraction, Transformation, Loading
- materialization of the DW
- materialization of Data Marts



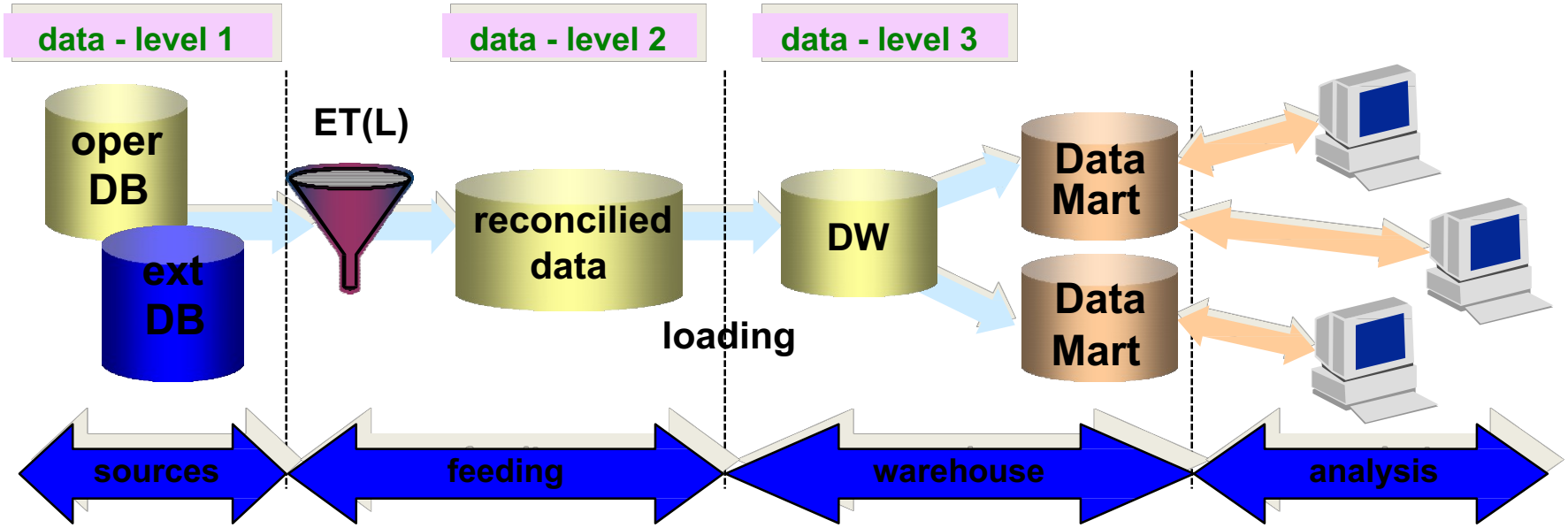
DW architecture: 2 Levels – independent DMs

- Data Mart are materialized by feeding
- DW = union of DMs

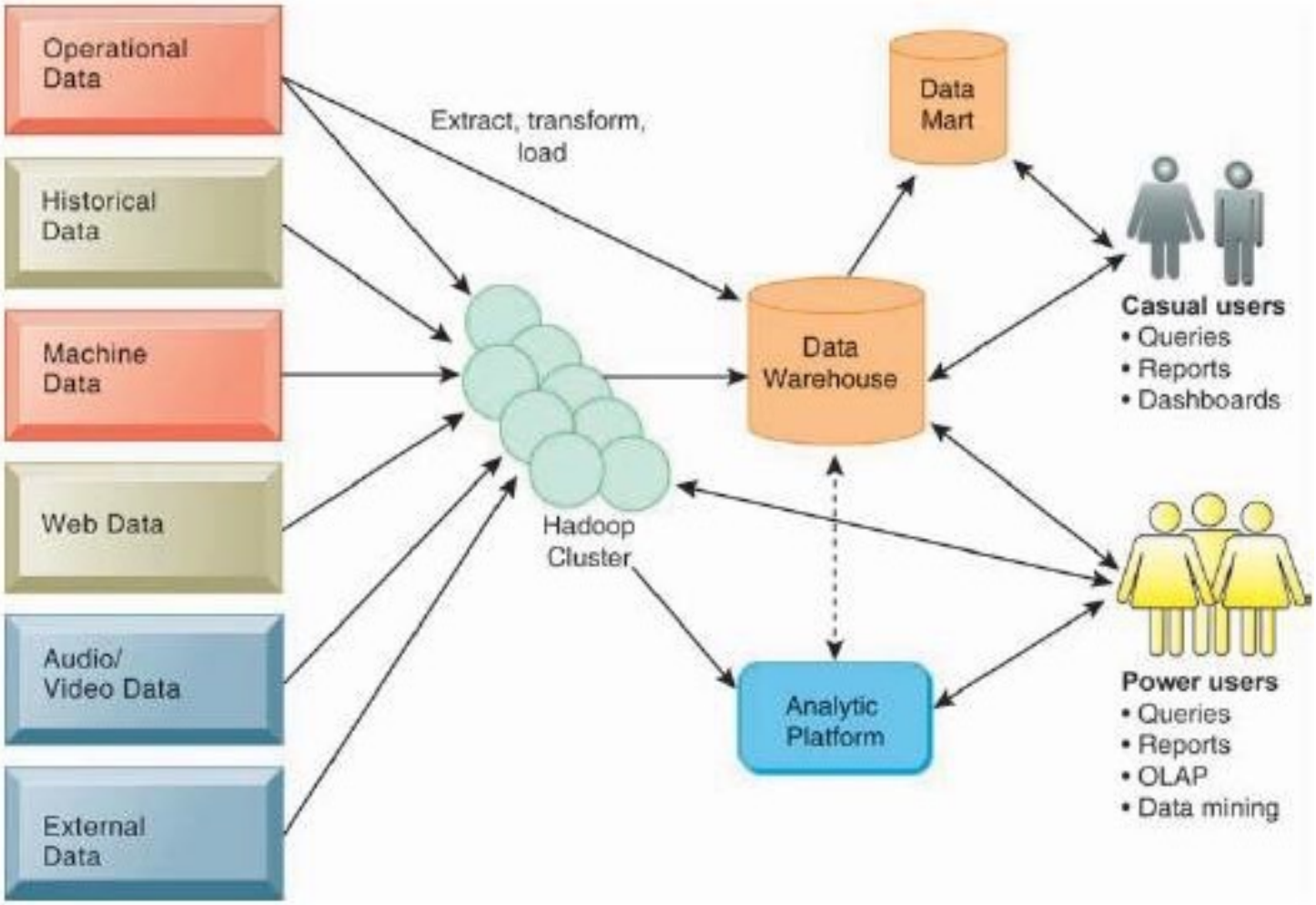


DW architecture: 3 Levels

- a level of "reconciled" data (operational data store) is introduced
- separation into two phases of ETL activities:
  1. extraction / transformation
  2. loading



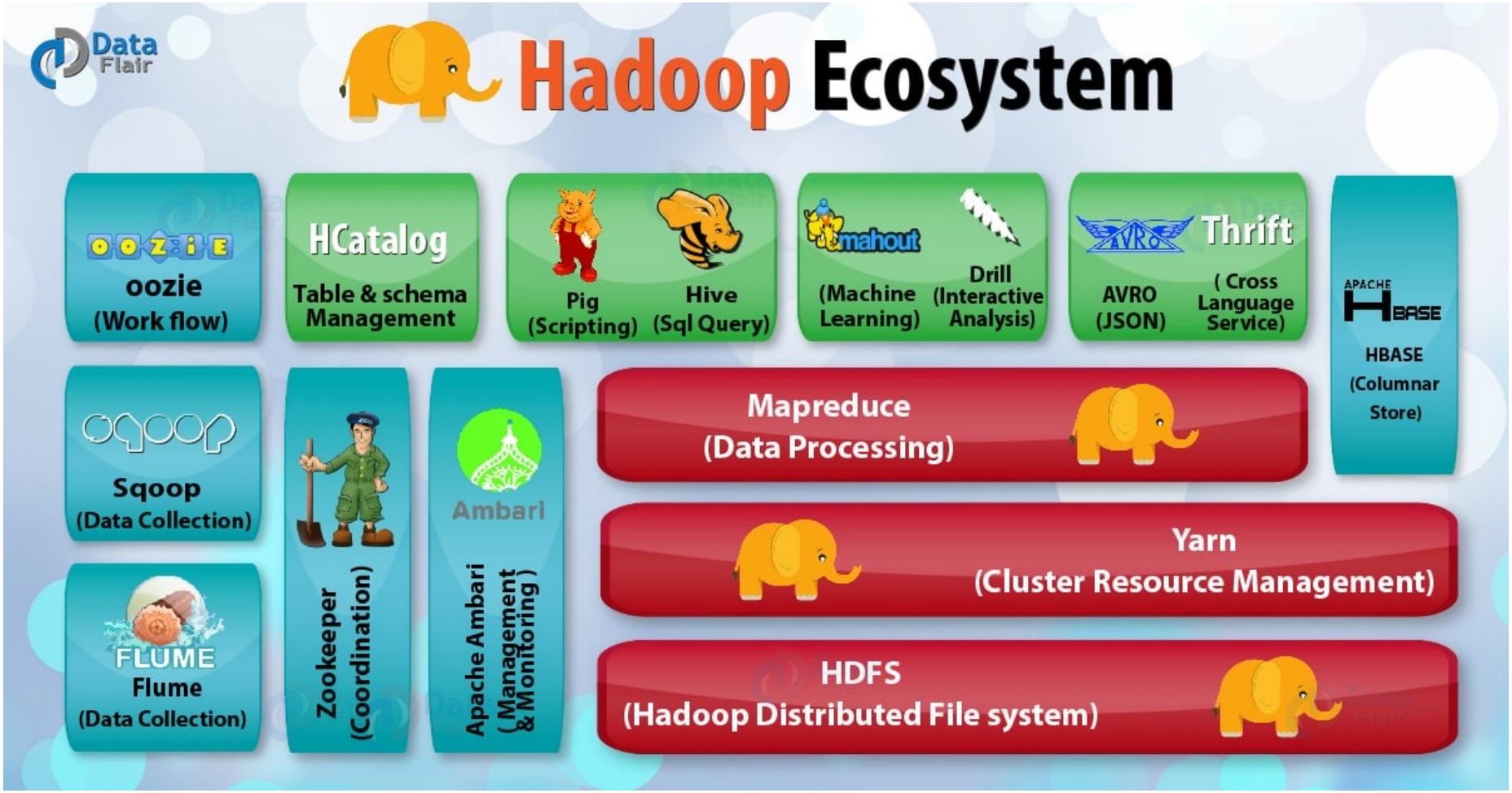
Data Source



Apache Hadoop

- The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models
- It is designed to scale up from single servers to thousands of machines, each offering local computation and storage
- Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures

Apache Hadoop Ecosystem



ETL: Extraction, Transformation, Loading

Operational Data, External Data

- extraction
- cleaning - validation - filtering
- transformation

Reconciled Data

- loading

Data Warehouse

Extraction

- initial extraction:
  - targeted at the creation of the DW
- furter extractions:
  - static (replaces the whole DW)
  - incremental
    - log (journal)
    - timestamp

Cleaning

- changing **VALUES**
- duplicates
- inconsistencies
  - domain violation
  - functional dependency violation
- null values
- misuse of fields
- spelling
- abbreviations (not homogeneous)

Transformation

- changing FORMATS:
  - misalignment of formats
  - field overloading
  - inhomogeneous coding

Loading

- Refresh:
  - ex-novo loading of the whole DW
- Update:
  - differential updates

Metadata

- internal metadata
  - concerning the administration of the DW (i.e., sources, transformations, schemas, users, etc..)
- external metadata
  - interesting for users (e.g., measurement units, possible combinations)
- STANDARDs
- CWM - Common Warehouse Model (OMG), defined by:
  - UML (Unified Modeling Language)
  - XML (eXtensible Markup Language)
  - XMI (XML Metadata Interchange)

OMG = Object Management Group: **CORBA** (Common Object Request Broker Architecture), **UML** (Unified Modeling Language), **MDA** (Model-Driven Architecture)