Data Lakes

ELENA BARALIS POLITECNICO DI TORINO



Data lake

Data repository for
 Original data in *raw* format
 Transformed data used for various types of reporting

Data formats

Structured data (e.g., relational data)

Semi-structured data (e.g., CSV, JSON, XML)

Unstructured data (e.g., text documents, emails)

Binary data (e.g., images, audio files)

Query more similar to a google search (+ data wrangling)





Why data lakes?

Often not all questions data can answer are known a-priori
 hard to store data in some «optimal» form

An attempt to break down information silosInformation not adequately shared among data systems

Based on exploiting massive, cheap data storage





Data lakes characteristics

Data lakes store all data

DW design requires deciding what data to include (and to not include) in the warehouse

Data lakes include also data that might be used "someday"

Data lakes manage all data types

Data lakes provide service to all users

Users process a variety of different types of data and answer new questions

Data lakes adapt easily to changes

All data is stored in its raw form and is always accessible

Users are empowered to explore data in novel ways

Data lakes provide faster insight

... but early access to the data comes at a price





Data warehouse

Relational data coming from transactional systems, operational databases, and line of business applications

Schema designed prior to DW implementation (schema-on-write)

□ High cost storage

Data quality: highly curated data that serves as the central version of the truth

Users are business analysts

Analytics: BI and visualization, batch reporting





Data lake

Data is both non-relational and relational, coming from IoT devices, web sites, mobile apps, social media, and corporate applications

Schema is written at the time of analysis (schema-on-read)

Low-cost storage

Data quality: Any data that may or may not be curated (ie. raw data)

Users are data scientists, data developers

business analysts, if using curated data

Analytics: full-text search, machine learning, predictive analytics, data discovery and profiling





Pros of data lakes

- Ability to harness more data, from more sources, in less time
- Data structures and business requirements are defined only when needed
- Empowering users to collaborate and analyze data in different ways
 self service analytics
- Integration happens outside the storage environment
- Minimal involvement of IT
 - Wrangling with data is a self-service function
- Sandboxes for self-service analytics
 - Need well defined problems





Cons of data lakes

Raw data is stored with no oversight of the contents
 Storing data does not, on its own, provide business value
 Need data governance, semantic consistency, mechanism to catalog data

Consistency and data quality are uncertain
 Data brought into a data lake is co-located not integrated

Business users don't have time/willingness to learnHow can they wrangle with raw data?

Rogue queries can bring down big clusters

The central question is whether collecting and storing data without a pre-defined business purpose is a good idea





From data lakes...





... to data swamps

massive repositories of data that are completely inaccessible to end users

data collected without any clear way to get value from it

risk to be abandoned (budget cut)

To avoid drowning in your data lake

Collect less data, at least in the beginning



