



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Introduzione ai DBMS NoSQL

## Basi di Dati

*Corso di Laurea in Informatica per il Management*

*Alma Mater Studiorum - Università di Bologna*

---

**Prof. Marco Di Felice**

Dipartimento di Informatica – Scienza e Ingegneria

marco.difelice3@unibo.it

# Introduzione ai DBMS NoSQL

---

**NoSQL:** Movimento che promuove l'adozione di **DMBS non basati sul modello relazionale**.

- Il termine NOSQL appare per la prima volta in una pubblicazione di Carlo Strozzi nel 1998.
- Oggi, il termine NOSQL viene usato per lo più nell'accezione **NoT Only SQL**.

*“Next generation databases mostly addressing some of the points: being non-relational, distributed, open source and horizontally scalable”* (definizione da <http://nosql-database.org/>)

# Introduzione ai DBMS NoSQL

---

## PROPRIETA dei SISTEMI NoSQL

- (Principalmente) database **distribuiti**
- (Generalmente) tool **open-source**
- NON dispongono di **schema**
- NON supportano operazioni di **join**
- NON implementano le proprietà **ACID** delle transazioni
- Sono **scalabili** orizzontalmente
- Sono in grado di gestire **grandi moli di dati**
- Supportano le **repliche** dei dati

# Introduzione ai DBMS NoSQL

---

Motivazioni legate alla diffusione dei database **NoSQL**:



- Gestione dei **Big-data**
- Limitazioni del **modello relazionale**
- Teorema **CAP**

# Introduzione ai DBMS NoSQL

---

**Big data:** moli di dati, eterogenei, destrutturati, difficili da gestire attraverso tecnologie tradizionali.

- Tecnologie tradizionali → RDBMS
- Il termine big-data è oggi usato sia per denotare tipologie di dati, sia le **tecnologie** e i **tool** di gestione degli stessi.
- Le quattro **V** dei big data: ***Volume, Velocità, Varietà, Valore***

# Introduzione ai DBMS NoSQL

---

✧ **Volume:** Big data = grosse moli di data

Esempio: Dati da esperimenti scientifici



<http://home.web.cern.ch/about/computing>

Acceleratore di particelle  
presso il CERN di Zurigo

- 600 milioni di collisioni al secondo
- 30 Petabyte di dati annui relativi ai *collision event*

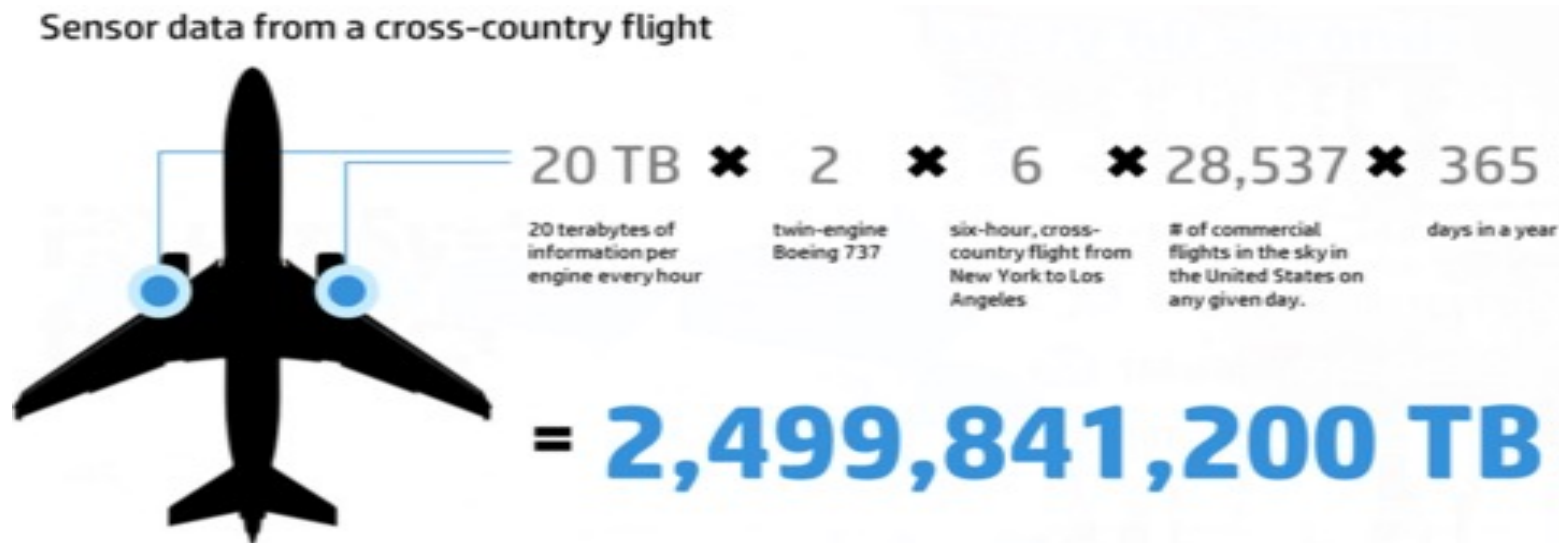
**BASI DI DATI**

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT

# Introduzione ai DBMS NoSQL

✧ **Volume:** Big data = grosse moli di data

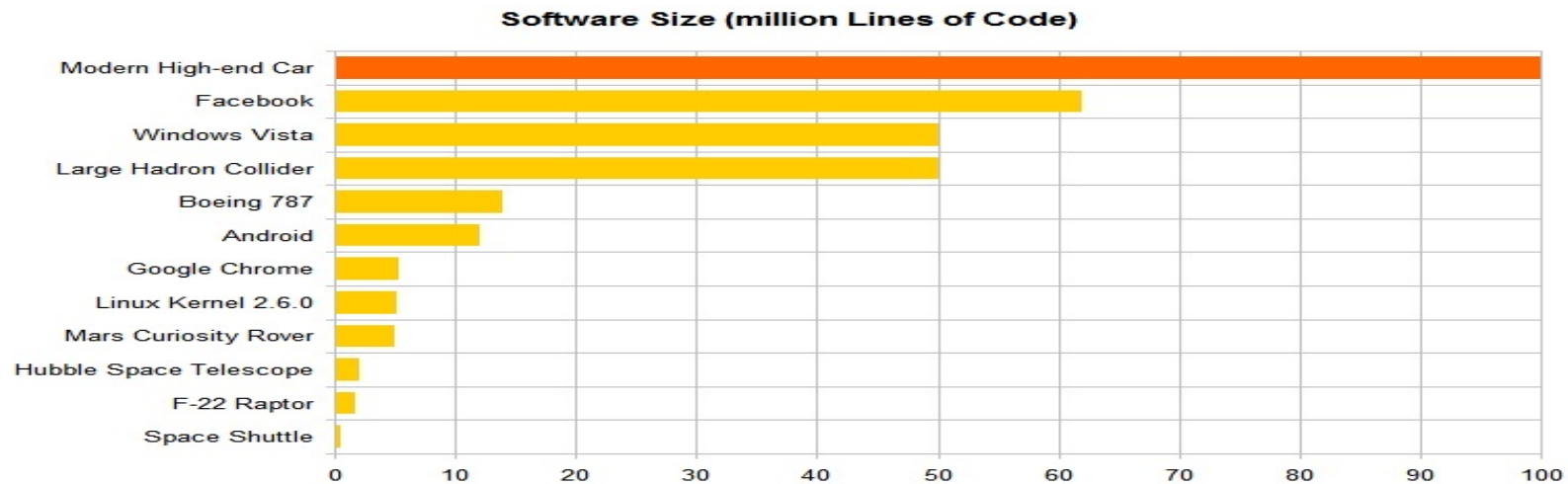
Esempio: Sensoristica



# Introduzione ai DBMS NoSQL

✧ **Volume:** Big data = grosse moli di data

Esempio: The Internet of Things (IoT)



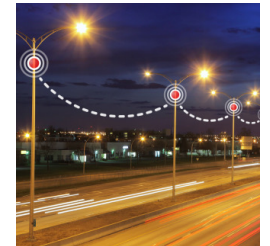
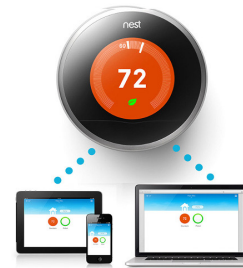
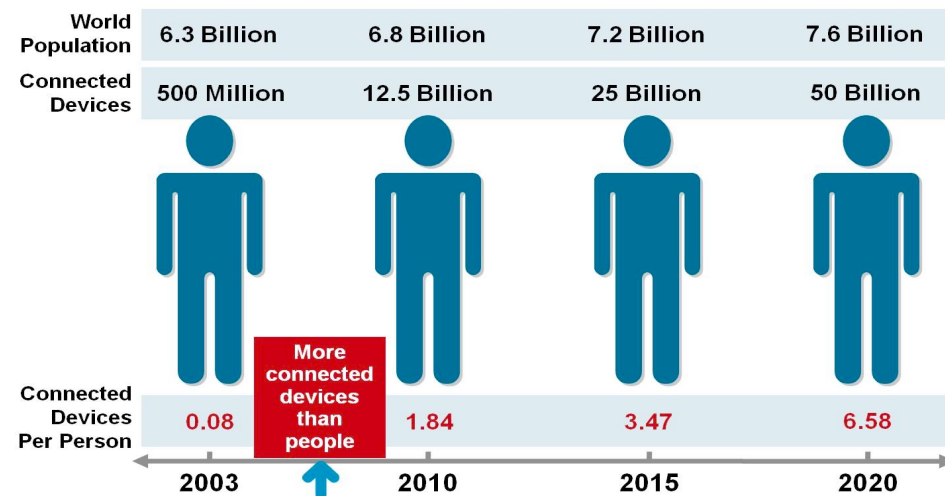
<http://www.informationisbeautiful.net/visualizations/million-lines-of-code/>



# Introduzione ai DBMS NoSQL

✧ **Volume:** Big data = grosse moli di data

Esempio: The Internet of Things (IoT)

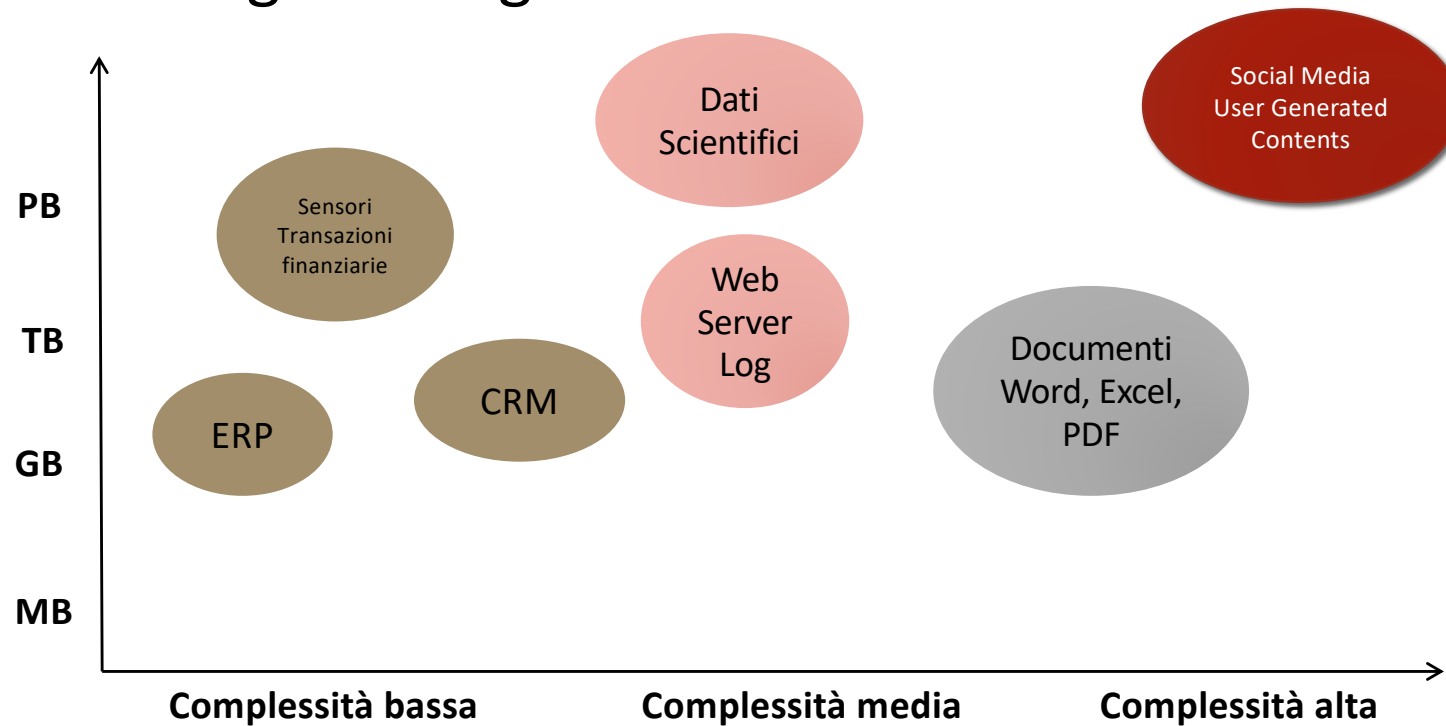


BASI DI DATI

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT

# Introduzione ai DBMS NoSQL

✧ **Volume:** Big data = grosse moli di data



# Introduzione ai DBMS NoSQL

✧ **Velocity**: Big data = *stream* di dati

Esempio: Sistemi health-care



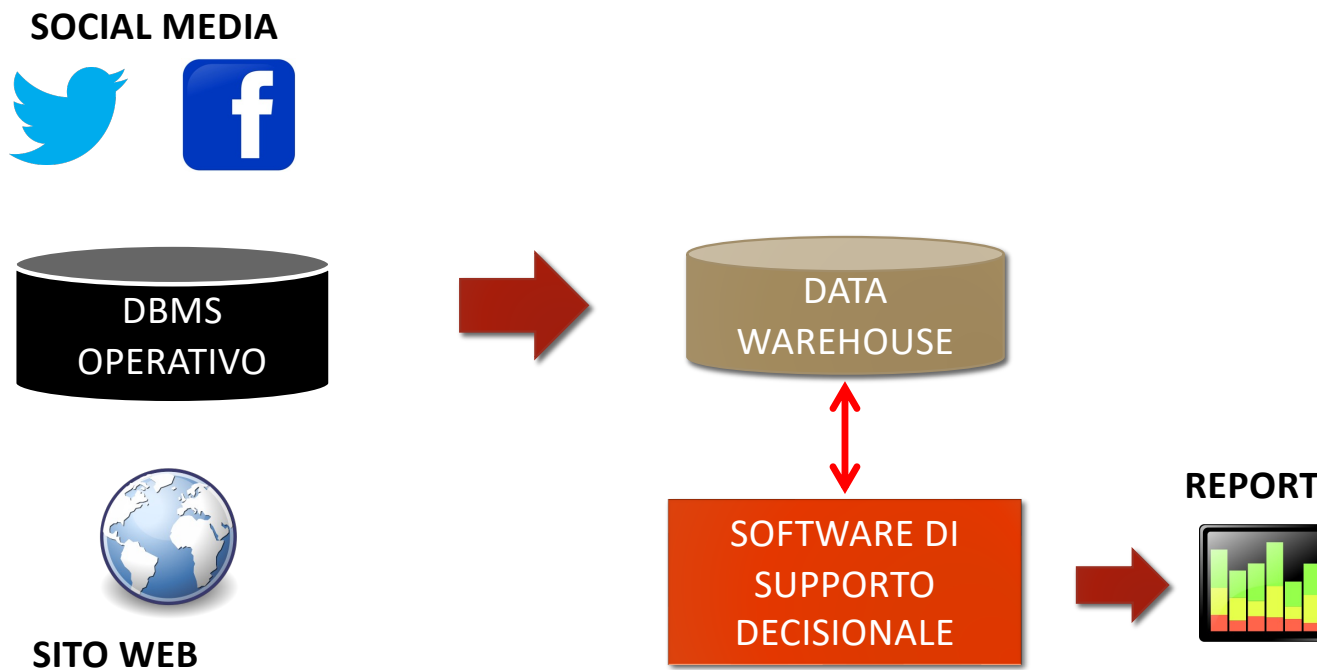
[http://www-03.ibm.com/press/us/en/pressrelease/40624.wss?cm\\_mc\\_uid=34296522129514448284754&cm\\_mc\\_sid\\_50200000=1444828475](http://www-03.ibm.com/press/us/en/pressrelease/40624.wss?cm_mc_uid=34296522129514448284754&cm_mc_sid_50200000=1444828475)

## Early-Warning Health-Care System

- Predizione della pressione arteriosa nel cervello in pazienti che hanno subito eventi traumatici rilevanti.
- Stream di 1000 dati al secondo.

# Introduzione ai DBMS NoSQL

✧ **Variety:** Big data = *dati eterogenei, multi-sorgente*



**BASI DI DATI**

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT

# Introduzione ai DBMS NoSQL

---

✧ **Valore:** possibilità di estrarre conoscenza dai big-data

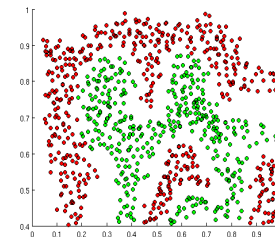
- **Data Mining:** tecniche di **apprendimento computerizzato** per analizzare ed estrarre **conoscenze** da collezioni di dati.



- **Pattern** e **relazioni** non note a priori e non immediatamente identificabili.

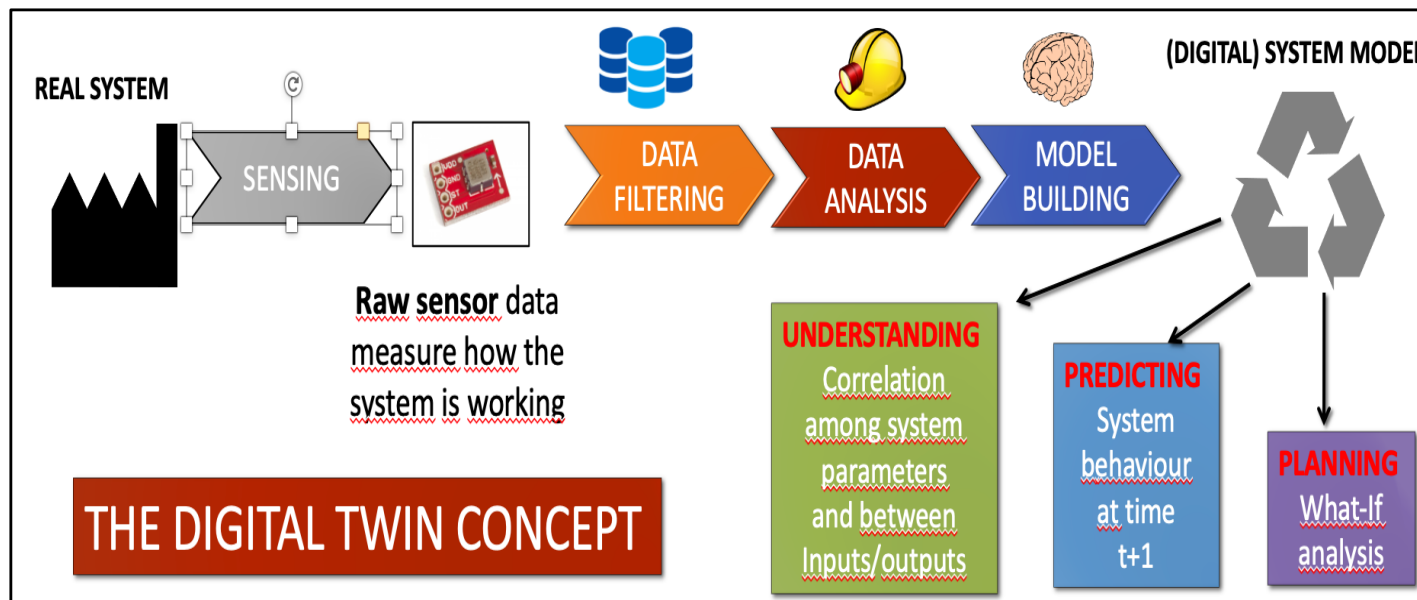


- Come? Mediante utilizzo di tecniche di **machine learning**, **intelligenza artificiale** e **statistiche** ...



# Introduzione ai DBMS NoSQL

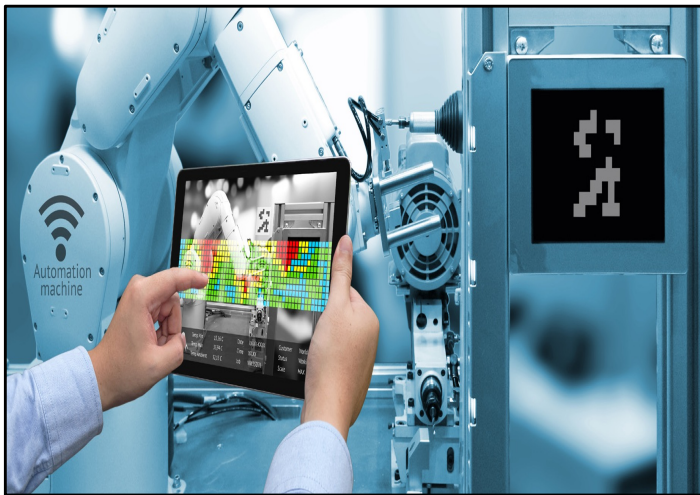
✧ **Valore:** possibilità di estrarre conoscenza dai big-data



# Introduzione ai DBMS NoSQL

---

✧ **Valore:** possibilità di estrarre conoscenza dai big-data



PREDICTIVE  
MAINTANANCE



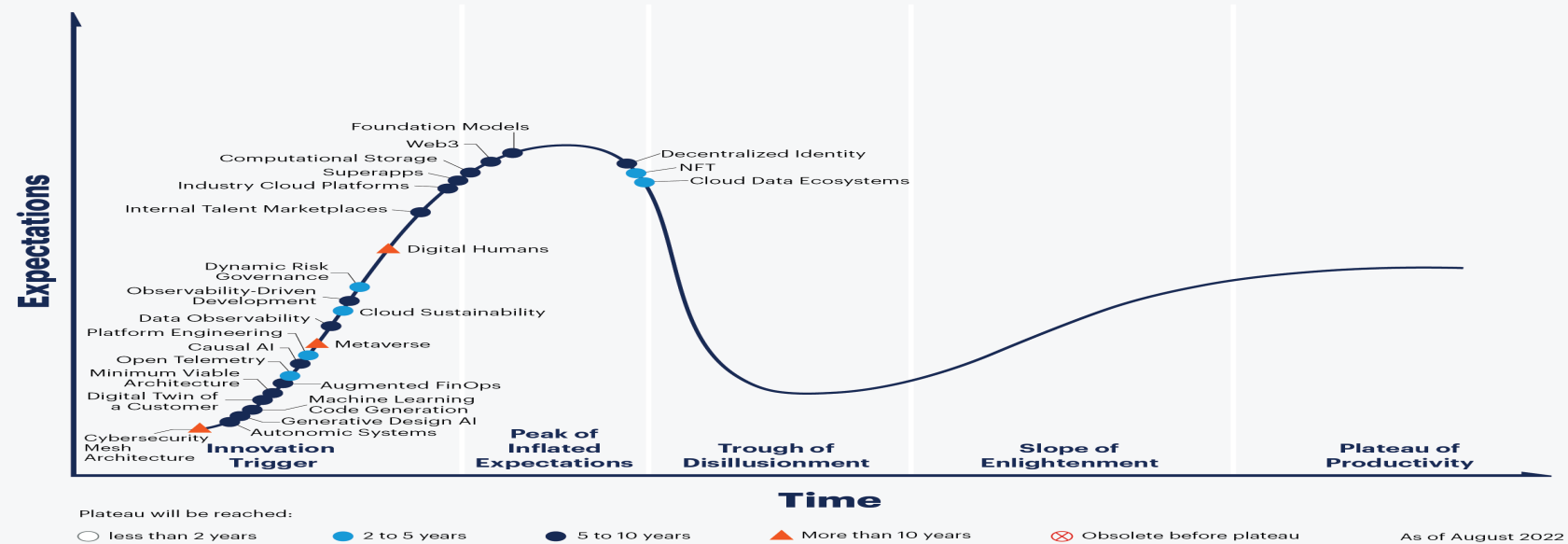
PRECISION IRRIGATION in  
SMART FARMING

**BASI DI DATI**

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT

# Introduzione ai DBMS NoSQL

## Hype Cycle for Emerging Tech, 2022



gartner.com

Source: Gartner  
© 2022 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner and Hype Cycle are registered trademarks of Gartner, Inc. and its affiliates in the U.S. 1893703

Gartner®

BASI DI DATI

PROF. MARCO DI FELICE – CORSO DI LAUREA IN INFORMATICA PER IL MANAGEMENT



# Introduzione ai DBMS NoSQL

---

Motivazioni legate alla diffusione dei database **NoSQL**:

- Gestione dei **Big-data**

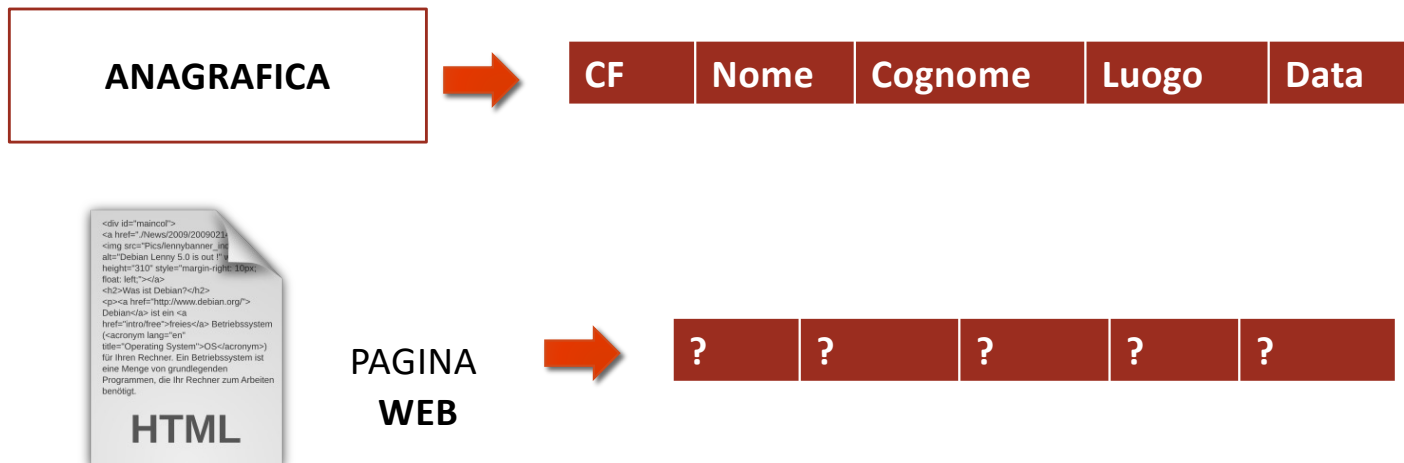


- Limitazioni del **modello relazionale**

- Teorema **CAP**

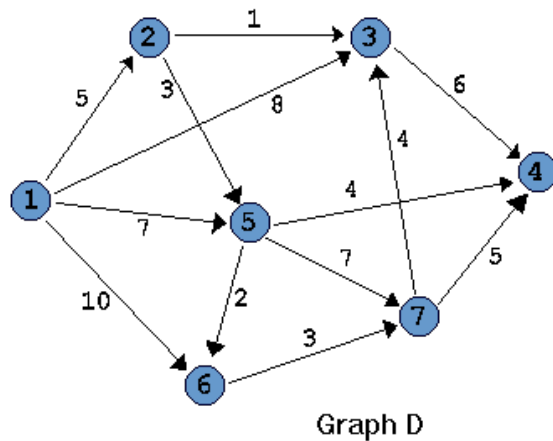
# Introduzione ai DBMS NoSQL

- (**LIMITAZIONE1**) Il modello relazionale presuppone implicitamente la presenza di **dati strutturati** in rappresentazione tabellare, che accade se i dati non si presentano in tale forma?.

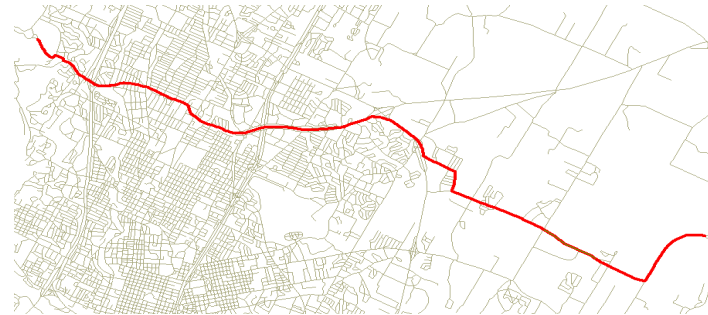


# Introduzione ai DBMS NoSQL

- ✧ (**LIMITAZIONE2**) Alcune operazioni non possono essere implementate in SQL (almeno, nella sua versione base vista fin qui).



Es. Memorizzazione di un grafo, e calcolo del percorso minimo tra due punti.



# Introduzione ai database NoSQL

---

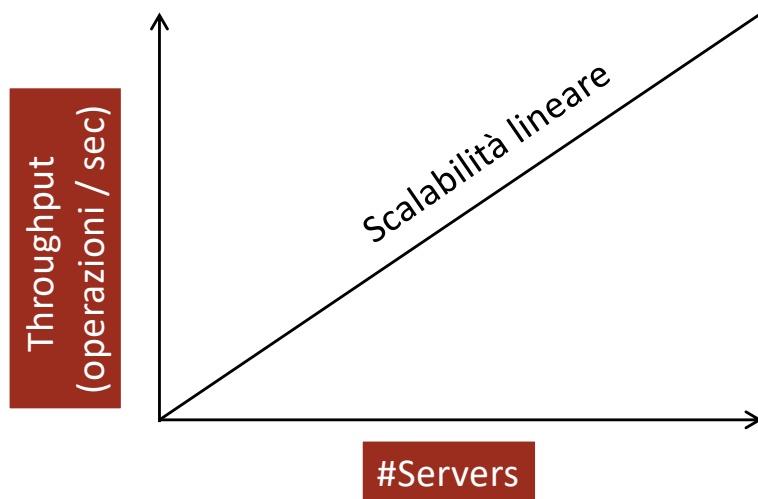
✧ (**LIMITAZIONE3**) Scalabilità *orizzontale* dei DMBS relazionali.

**Scalabilità** (def informale): capacità di un sistema di migliorare le proprie prestazioni per un certo carico di lavoro, quando vengono aggiunte *nuove risorse* al sistema.

- Scalabilità *verticale* → aggiungere più potenza di calcolo (RAM, CPU) ad i nodi che gestiscono il DB.
- Scalabilità *orizzontale* → aggiungere più nodi al cluster.

# Introduzione ai database NoSQL

✧ (**LIMITAZIONE3**) Scalabilità *orizzontale* dei DMBS relazionali.




## PROBLEMI DEI RDBMBS

- Gestione dei vincoli
- Repliche dei dati
- Gestione delle transazioni
- Soddisfacimento delle proprietà ACID

# Introduzione ai DBMS NoSQL

---

Motivazioni legate alla diffusione dei database **NoSQL**:

- ✧ Gestione dei **Big-data**
- ✧ Limitazioni del **modello relazionale**
-  ✧ Teorema **CAP**

# Introduzione ai DBMS NoSQL

---

Il teorema di Brewer (**CAP Theorem**) afferma che **un sistema distribuito può soddisfare al massimo solo due delle tre proprietà elencate sotto:**

- **Consistency:** Tutti i nodi della rete vedono gli stessi dati
- **Availability:** Il servizio è sempre disponibile
- **Partion Tolerance:** Il servizio continua a funzionare correttamente anche in presenza di perdita di messaggi o di partizionamenti della rete.

# Introduzione ai DBMS NoSQL

---

## CAP THEOREM PER DATABASE DISTRIBUITI:

Nel caso di un DB distribuito (gestito da un cluster), **è possibile soddisfare al massimo solo due** delle tre proprietà elencate sotto:

- **Consistency:** Se l'utente A modifica il dato X sul server 1, e B legge X dal server 2, B legge l'ultima versione disponibile di X.
- **Availability:** Se un utente effettua una query sul server A o B, la query restituisce un risultato.
- **Partion Tolerance:** Il servizio continua a funzionare correttamente anche in presenza di perdita di messaggi o di partizionamenti della rete.



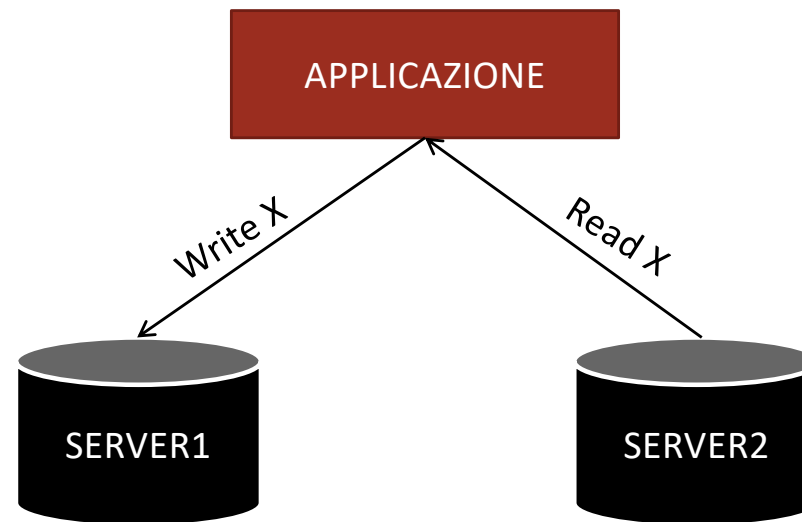
# Introduzione ai DBMS NoSQL

CAP THEOREM PER DATABASE DISTRIBUITI:

**CASO 1:** Consistency + Availability

## **NO Partition Tolerance**

Il sistema *non funziona correttamente* in caso di perdita di messaggi.

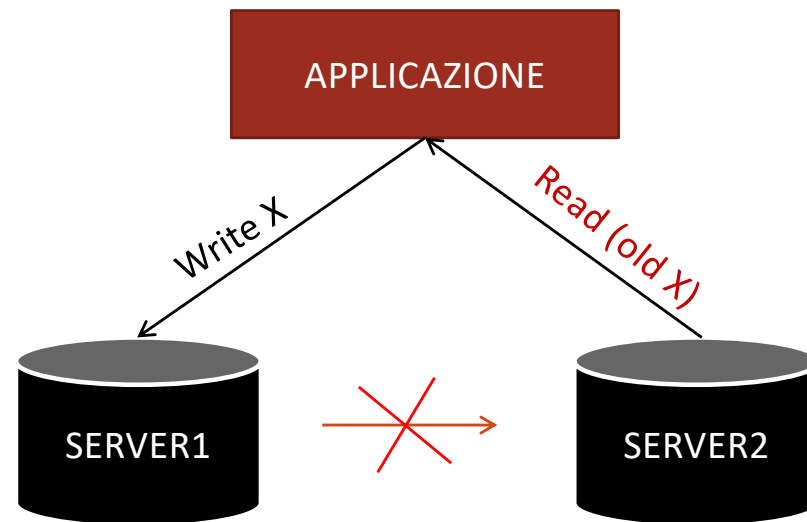


# Introduzione ai DBMS NoSQL

CAP THEOREM PER DATABASE DISTRIBUITI:

**CASO 2:** Availability + Partition Tolerance

**NO Consistency**  
Repliche del dato  
non aggiornate!

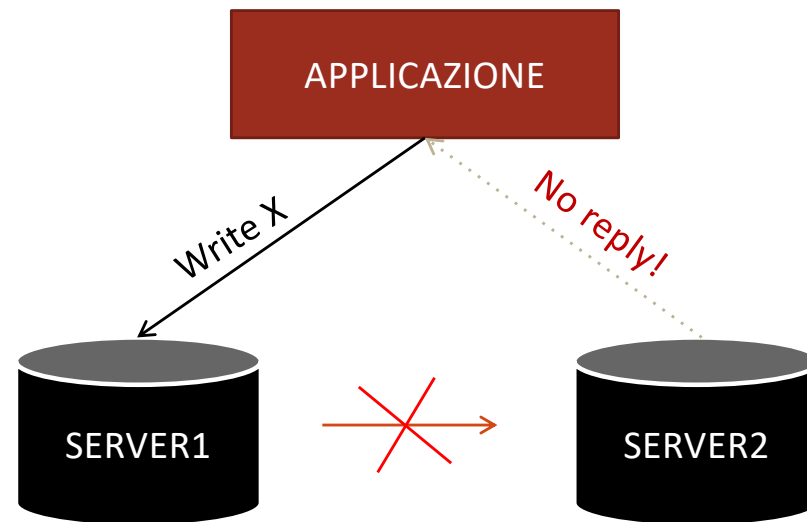


# Introduzione ai DBMS NoSQL

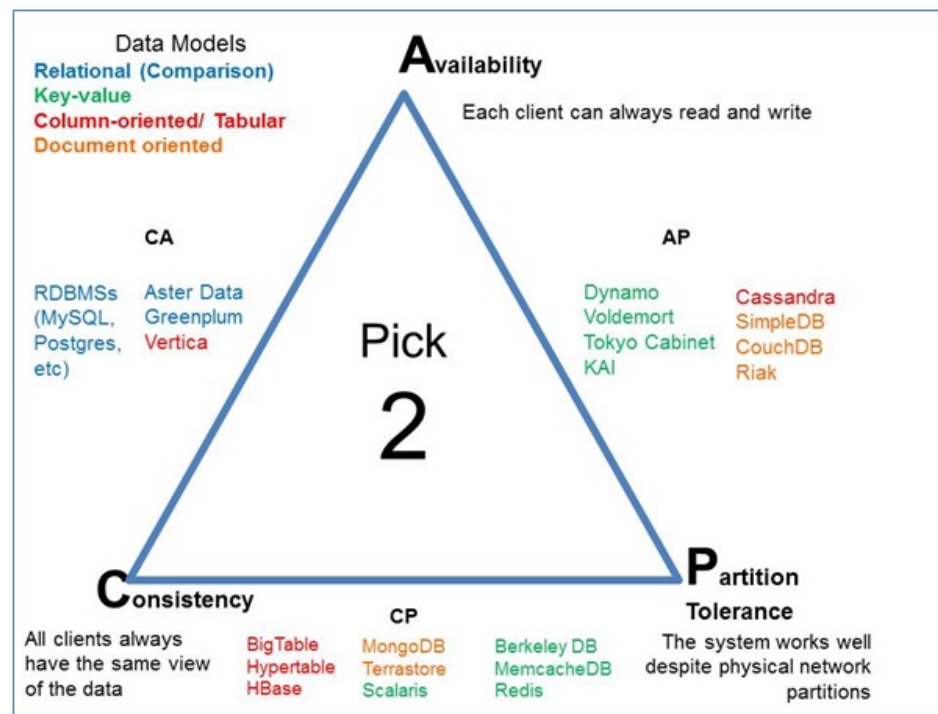
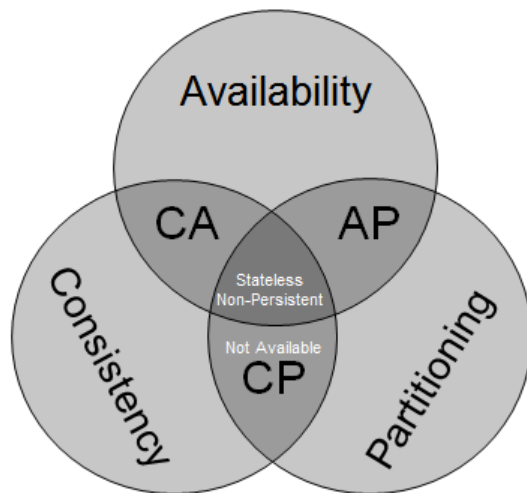
CAP THEOREM PER DATABASE DISTRIBUITI:

**CASO 3: Consistency+ Partition Tolerance**

**NO Availability**  
La query non  
produce risposta



# Introduzione ai DBMS NoSQL



# Introduzione ai DBMS NoSQL

---

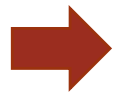
## PROPRIETA' BASE

- **Basically Available:** I nodi del sistema distribuito possono essere soggetti a guasti, ma il servizio è sempre disponibile.
- **Soft State:** La consistenza dei dati non è garantita in ogni istante.
- **Eventually Consistent:** Il sistema diventa consistente dopo un certo intervallo di tempo, se le attività di modifica dei dati cessano.

# Introduzione ai DBMS NoSQL

---

Il termine NoSQL identifica **una varietà di DBMS non relazionali**, basati su **modelli logici differenti**:



- Database **chiave/valore**
- Database **document-oriented**
- Database **column-oriented**
- Database **graph-oriented**

# Introduzione ai DBMS NoSQL

---


- Esempi: **BerkeleyDB, Project Voldemort**
- Dati di un DB come liste di coppie **chiave/valore** (array associativi o dizionari)
- **Chiave** → valore univoco per operazioni di ricerca
- **Valore** → qualsiasi cosa ...

Chiave	Valore
1	{Mario Rossi, 02311323}
2	{Mario Bianchi, 23}
3	{Dipartimento Informatica, Via Zamboni, 05143242}

# Introduzione ai DBMS NoSQL

---

Il termine NoSQL identifica **una varietà di DBMS non relazionali**, basati su **modelli logici differenti**:

- Database **chiave/valore**
-  ○ Database **document-oriented**
- Database **column-oriented**
- Database **graph-oriented**



# Introduzione ai DBMS NoSQL

---

- Esempi: **MongoDB**, **CouchDB**
- Gestione di dati **eterogenei** e **complessi** (semi-strutturati)
- Scalabili **orizzontalmente**, supporto per partizionamento (*sharding*) dei dati in sistemi distribuiti
- **Documenti** → coppie chiave/valore (JSON)
- Forniscono **funzionalità** per aggregazione/analisi dei dati (MapReduce)

# Introduzione ai DBMS NoSQL

---

Il termine NoSQL identifica **una varietà di DBMS non relazionali**, basati su **modelli logici differenti**:

- Database **chiave/valore**
- Database **document-oriented**

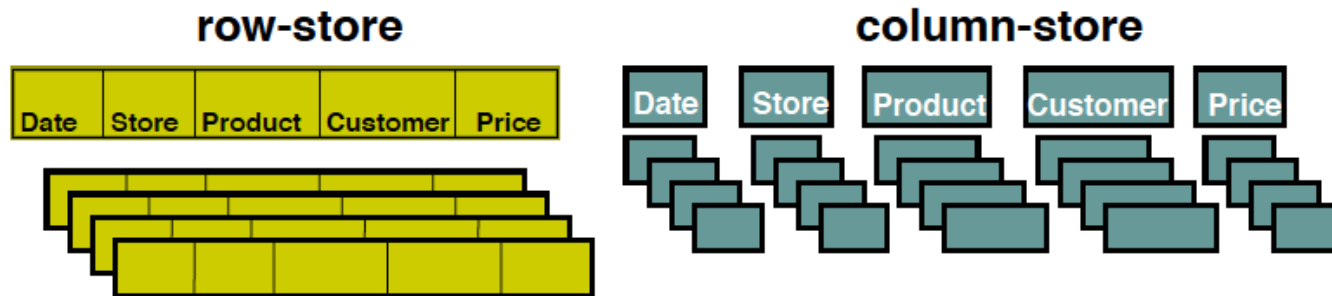


- Database **column-oriented**

- Database **graph-oriented**

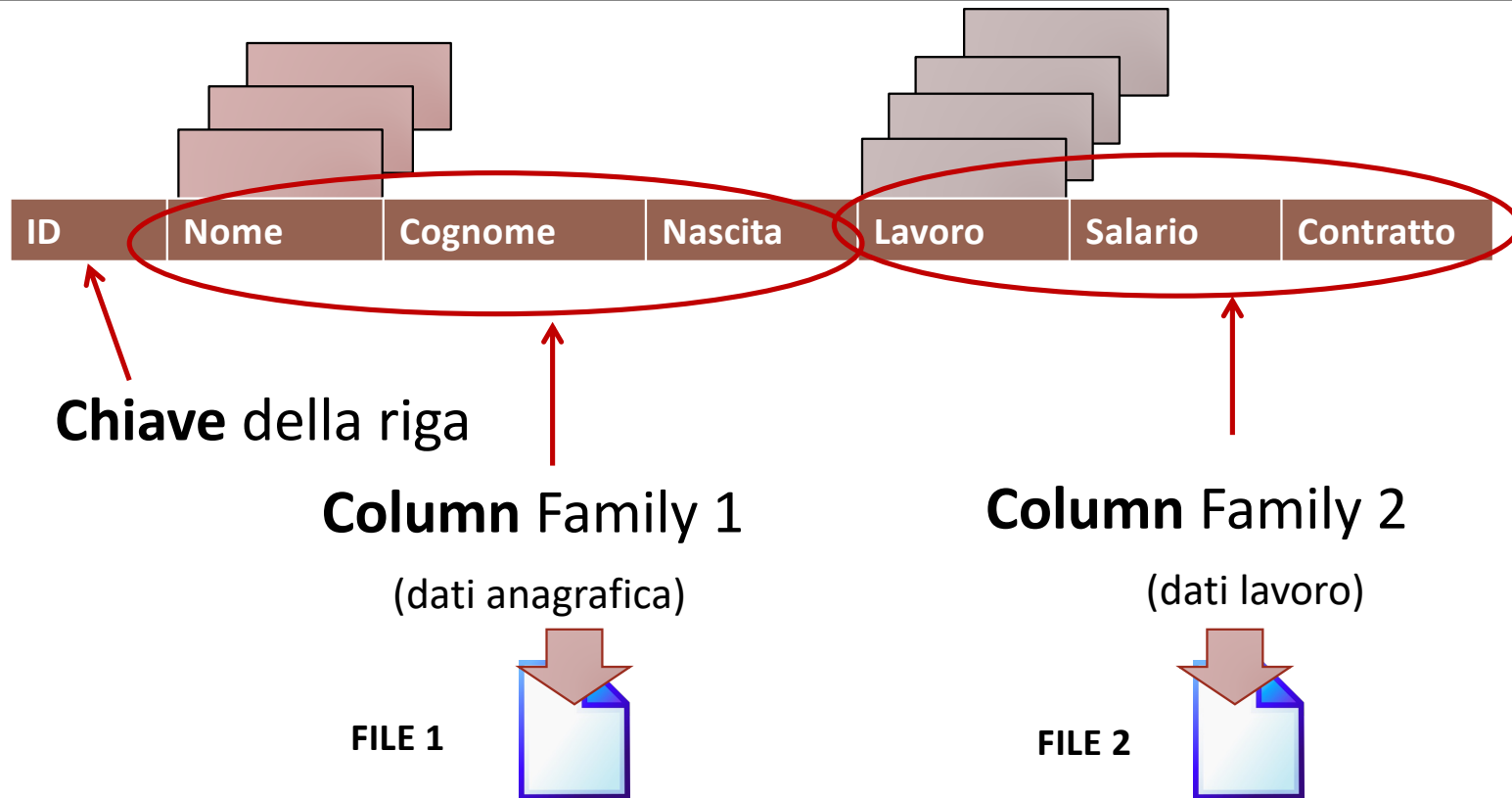
# Introduzione ai DBMS NoSQL

- Esempi: **HBase**, **Cassandra**
- Dati organizzati su colonne anziché su righe.
- **Column family** → contenitore di colonne. Ogni column family è scritta su un file diverso. Ogni riga dispone di una chiave primaria (**row key**).



† Stavros Harizopoulos, Daniel Abadi, Peter Boncz (2009)

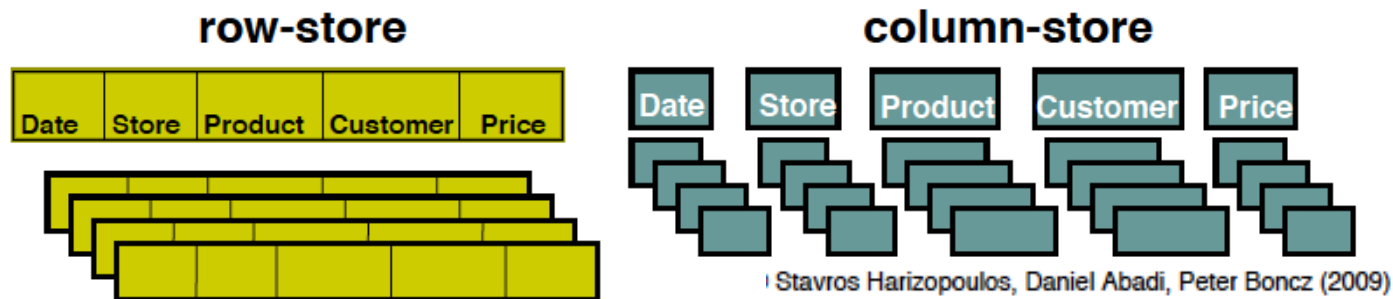
# Introduzione ai DBMS NoSQL



# Introduzione ai DBMS NoSQL

---

- Schema **flessibile**
- Maggiore efficienza nello **storage**
- Maggiore possibilità di **compressione dati**
- Usato in sistemi dati **read-oriented** (es. warehousing)



# Introduzione ai DBMS NoSQL

---

Il termine NoSQL identifica **una varietà di DBMS non relazionali**, basati su **modelli logici differenti**:

- Database **chiave/valore**
- Database **document-oriented**
- Database **column-oriented**



- Database **graph-oriented**

# Introduzione ai DBMS NoSQL

- Esempi: **Neo4J**, **Titan**
- Dati strutturati sotto forma di grafi: nodi → attributi/righe, archi → relazioni tra attributi/righe

