DBMS vs. DSMS

Event-Driven Architecture - **Longo Stefano**

Content

- What is a DBMS
  - What is a database
- What is a DSMS
  - What is a data stream
- Differences between DBMS and DSMS
- Limits of data stream model
- Differences on queries
- DBMS & DSMS – Case of use
- Conclusion

DBMS (DataBase Management System)

- A **Database** is an organized collection of data.

  - There are a lot of Database Models (Hierarchical, Relational, Semantic, XML, Object Oriented, NoSQL, …)

  - The most popular database systems since the 1980s have all supported the relational model as represented by the SQL language

- A **Database Management System** is a collection of programs that enables you to store, modify, and extract information from a database.

DSMS (Data Stream Management System)

What is a Data Stream?

- Large data volume, likely structured, arriving at a very high rate
- Not (only) what you see on youtube
- Definition (Golab and Ozsu, 2003):
  "A data stream is a real-time, continuous, ordered (implicitly by arrival time of explicitly by timestamp) sequence of items. It is impossible to control the order in which items arrive, nor it is feasible to locally store a stream in its entirety".

DSMS (Data Stream Management System)

- A DSMS is a computer program that permits to manage **continuous data streams** (assumed **infinite**).
- Data received from a DSMS is moving at **high pace**
- **Queries are continuous** (registered once, observed “forever”)
- Answer to queries in (nearly) **real-time required**
- For efficiency:
  - Probabilistic method
  - **Sliding window** (considering only a part of the stream)

Differences between DBMS and DSMS

- Fundamental difference: **data stream model**.
- In a data stream, data elements arrive **on-line** and stay only for a limited time period in memory.
- Consequently, the DSMS has to handle the data elements before the buffer is overwritten by new incoming data elements
- The size of **data streams** is potentially **unbounded** and can be thought of as an open-ended relation

Limits of Data Stream Model

Limits

- Stream data is unbounded..
  Memory is not unbounded, no way to store entire stream
- Query answer..
  Is not exact, we can only approximate
- To compute query results..
  Need to device algorithm with little memory consumption

Solutions

- Sliding Window: evaluate the query not over the entire past history of the data streams, but rather only over sliding windows of recent data from the streams
- Synopses: maintain only a synopsis of the data selecting random data points called sampling to summarization using histograms, wavelets or sketching
  (both methods cannot reflect the data accurately)
- Space used by the algorithm is important, although time required to process each stream is also relevant.

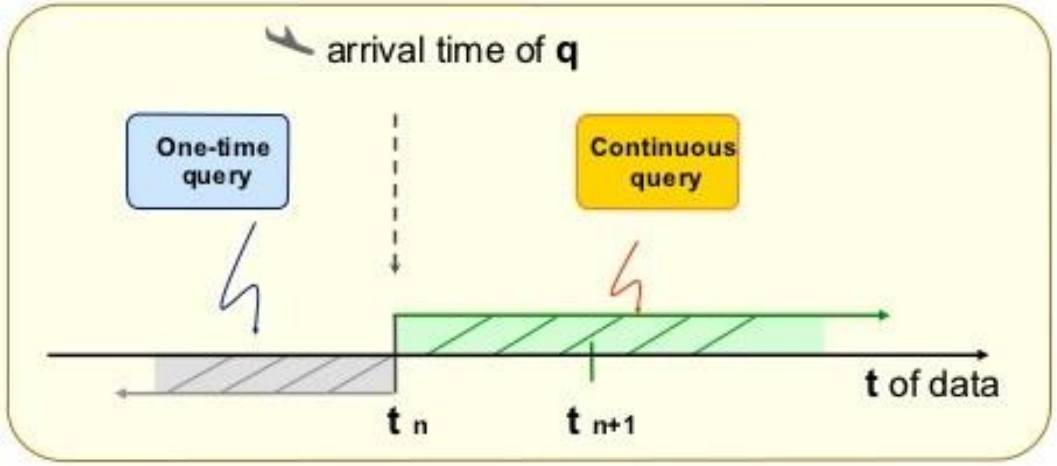
Differences on Queries

DBMS Queries (One-time Queries):

- Evaluated once over the data stored in the past in the database
- Queries is transient and the query answer is exact.

DSMS Queries (Continuous Queries):

- Waits for future incoming tuples
- Evaluated continuously as new tuples arrive
- Queries are persistent and the query answer is approximate.



source

Differences on Queries - example

DBMS

```
SELECT Name, Surname, Role, City
FROM Employees
WHERE city = ‘Berlin’
ORDER BY Surname, Name
```

Simple query that shows the name, surname, role and city of the company’ employees working in Berlin.
The output will be ordered by surname and name of the employees

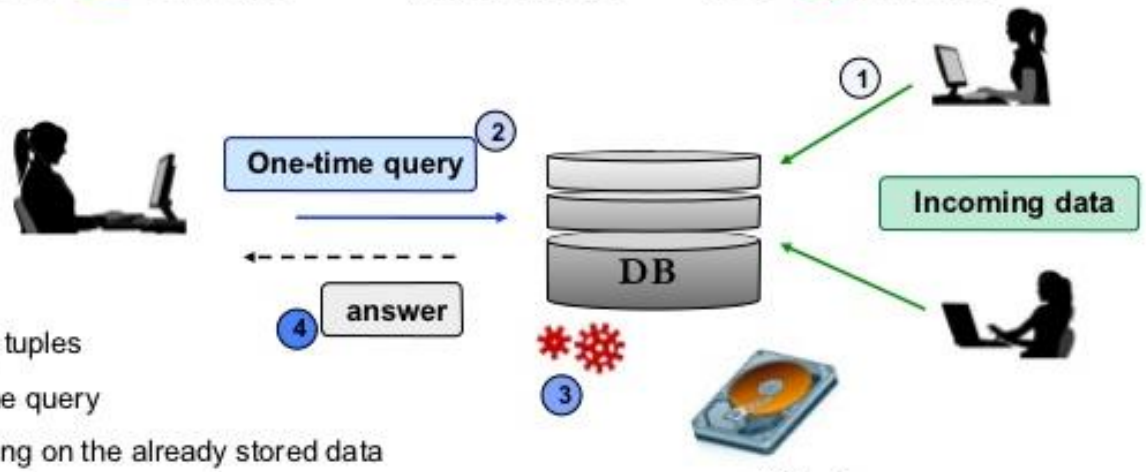
DSMS – Continuos Queries

```
SELECT Stream
    Rowtime,
    MIN(temp) OVER W1 AS Wmin_temp,
    MAX(temp) OVER W1 AS Wmax_temp,
    AVG(temp) OVER W1 AS Wavg_temp
FROM Weatherstream
WINDOW W1 AS ( RANGE INTERVAL '1' SECOND
PRECEDING );
```

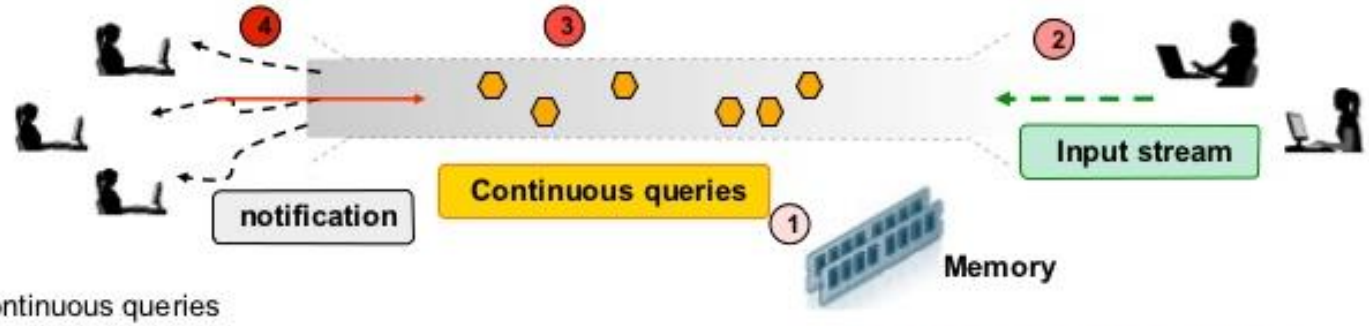
The query aggregates a sensor stream from a weather monitoring system.
It aggregates the minimum, maximum and average temperature values.
Window clause create a window of one second duration showing a stream of incrementally updated results with zero result latency.

CWI Centrum Wiskunde & Informatica

DBMS versus DSMS



1. Store incoming tuples
2. Submit one-time query
3. Query processing on the already stored data
4. Create answer



1. Submit continuous queries
2. Incoming streams
3. Input stream is processed on the fly
4. The produced results are continuously delivered to the clients

A *data stream* is a never ending sequence of tuples

source

DBMS - Case of use

Why to use a DBMS?

- Data independence and efficient access.
- Reduced application development time.
- Data integrity and security.
- Uniform data administration.
- Concurrent access, recovery from crashes.
- User-friendly declarative query language.

Database Applications:

- Banking: all transactions
- Airlines: reservations, schedules
- Universities: registration, grades
- Sales: customers, products, purchases

DSMS – Case of use

- Financial real-time analysis
- Video streaming
- Network monitoring and traffic engineering
- Security applications
- Telecom call records
- Web logs and click-streams
- Sensor networks
- Manufacturing processes

Conclusion

Database management system (DBMS)	Data stream management system (DSMS)
Persistent data (relations)	Volatile data streams
Random access	Sequential access
“Unbounded” disk store	Bounded main memory
One-time queries	Continuous queries (CQs)
Plannable query processing	Variable data arrival and data characteristics
Relatively low update rate	Potentially extremely high update rate