香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

SCHOOL OF
DATA SCIENCE
數據科學學院

CSC3170
Tutorial 10

School of Data Science

The Chinese University of Hong Kong, Shenzhen

Overview

Processing Models

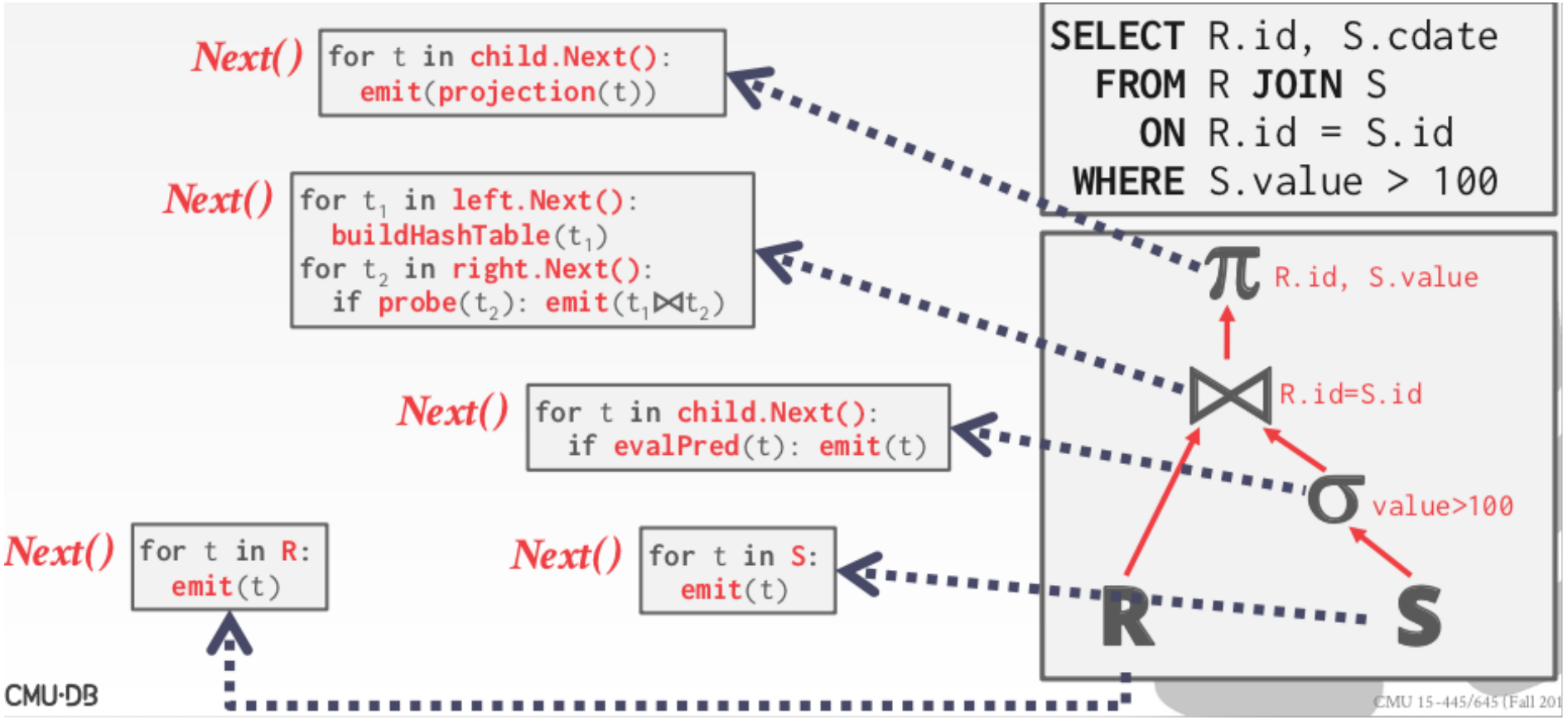
- Iterator model
- materialization model
- Vectorization model

Access Methods

- Sequential scan and its optimizations
  - Zone MAPS
  - LATE MATERIALIZATION
- Index scan
  - Multi-Index Scan

Processing Model: Iterator model

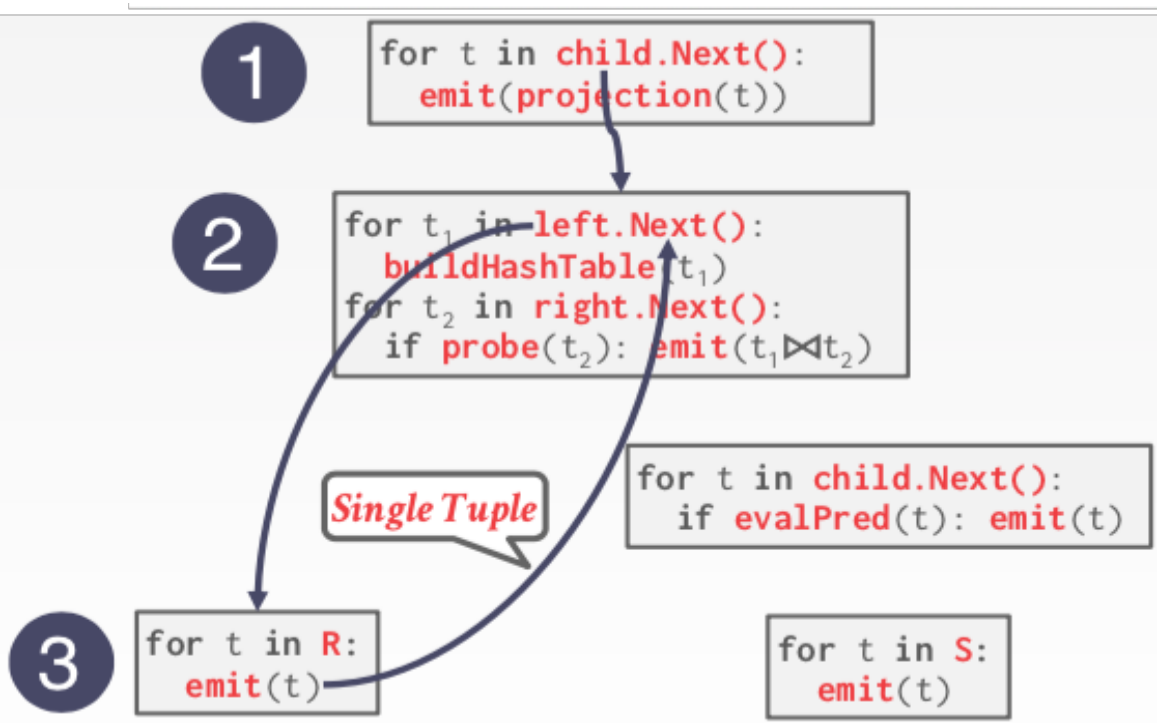
- Each operator in the query plan implements a next function.
- When called, the next function returns either a tuple or null, with null indicating that all data has been traversed.
- Each operator has an internal loop that repeatedly calls the next function of its child operators to retrieve the next piece of data for its own processing.



Processing Model: Iterator model

SCHOOL OF DATA SCIENCE

Some operators remain blocked until their children return all tuples; these operators are known as **pipeline breakers**. Examples include **Joins**, **Subqueries**, and **Order By** operations.



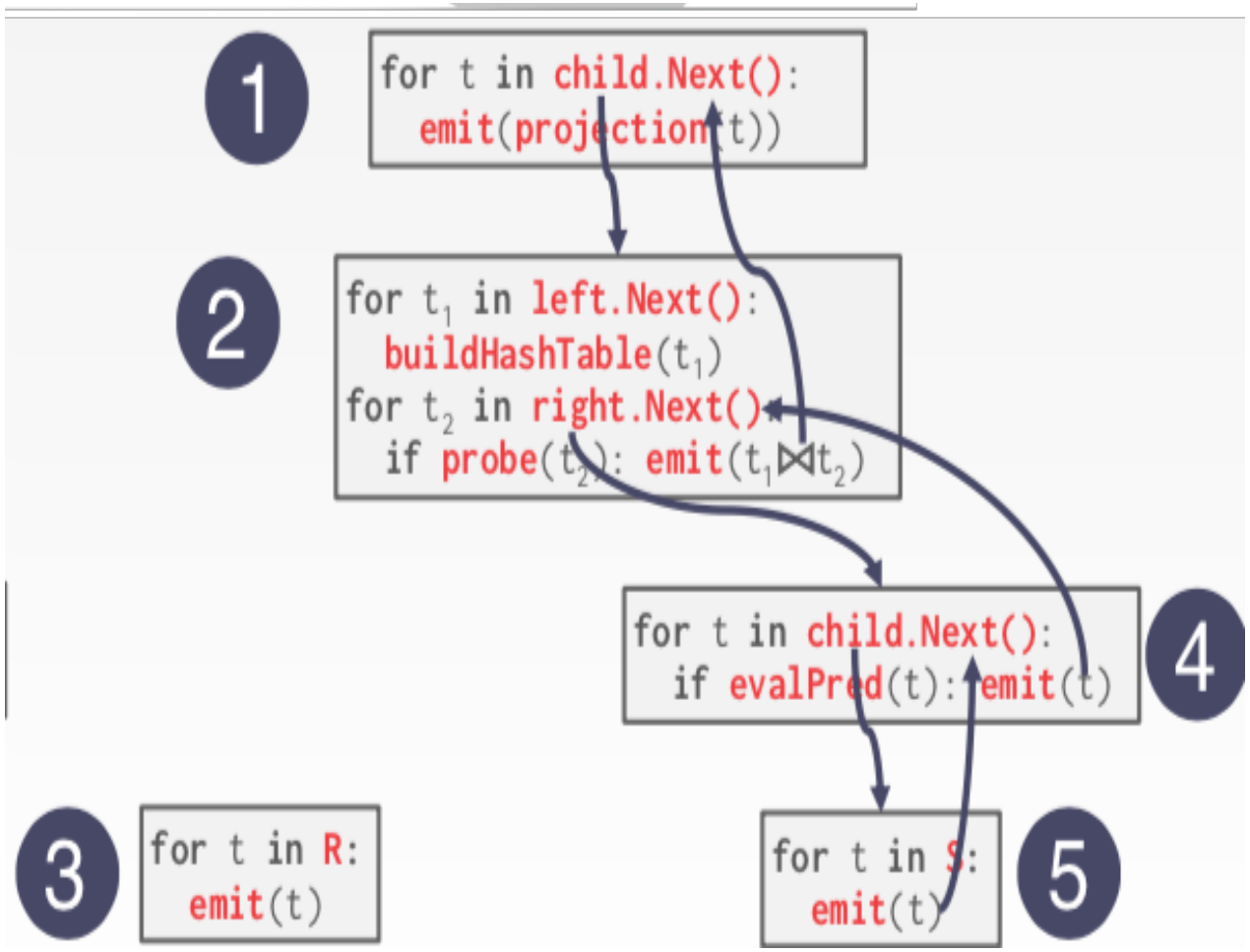
Initial Call to the Join Operator's Next():

- The top-level operator (Projection) invokes child.Next() on the Join operator to retrieve the first result.
- The Join operator needs to execute its join logic, and for a Hash Join, it must first build a hash table. Therefore, it gets **blocked** during the build phase.

Building the Hash Table:

- During the build phase, the Join operator calls left.Next() (corresponding to the scan operator for R).
- Each call to left.Next() retrieves a single tuple.
- For each record, the R.id value is added to the hash table until all records from the left subtree (R) are retrieved, completing the construction of the hash table.

Processing Model: Iterator model



Entering the Probe Phase:

- After building the hash table, the Join operator enters the Probe Phase.
- It calls the Next() method of its right child (scan operator for S) to retrieve records.

Processing Filter Condition:

For each record from S, it checks if S.value > 100. If not, the record is discarded. If satisfied, the probe() function checks for a matching R.id in the hash table. If a match exists, the records from R and S are joined, and the result is emitted.

Subsequent Calls:

Further calls to child.Next() do not rebuild the hash table. The Join operator continues probing the right subtree (S) and emits results for matches.

Completion:

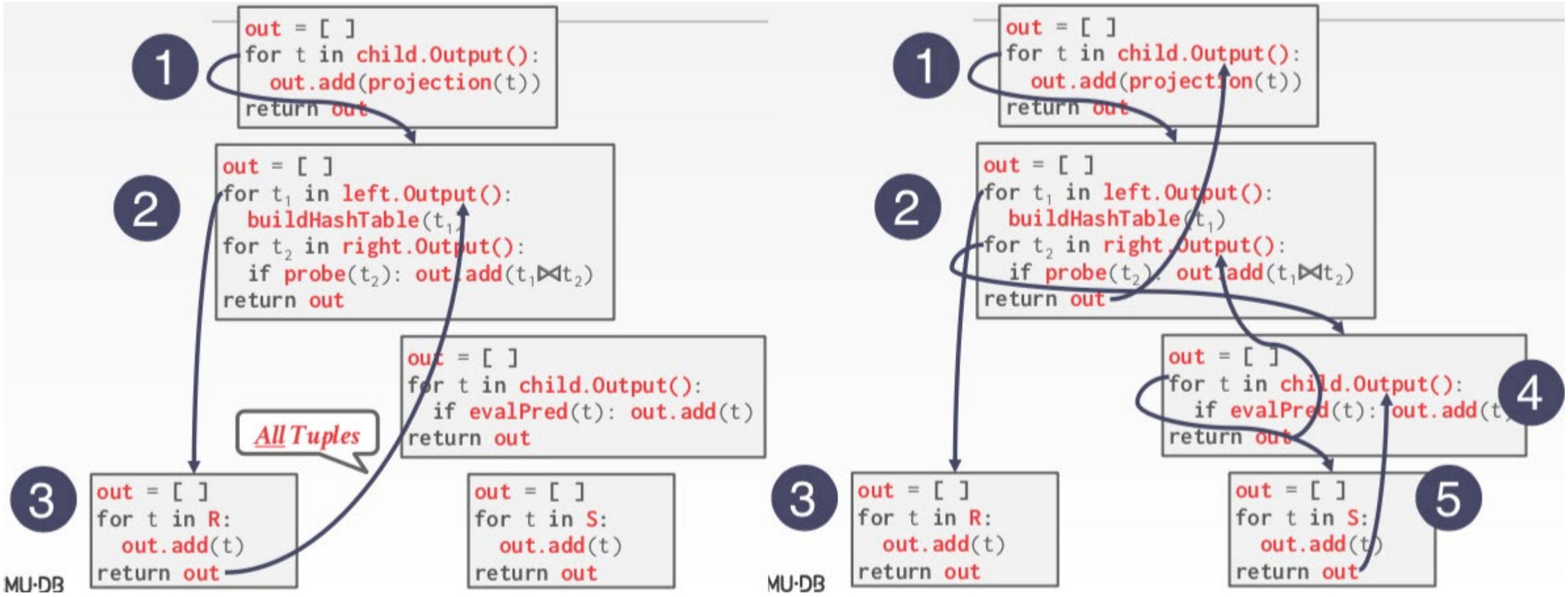
When all records from S are processed, child.Next() returns null, signaling the Join operator has finished execution.

Processing Model: Materialization Model

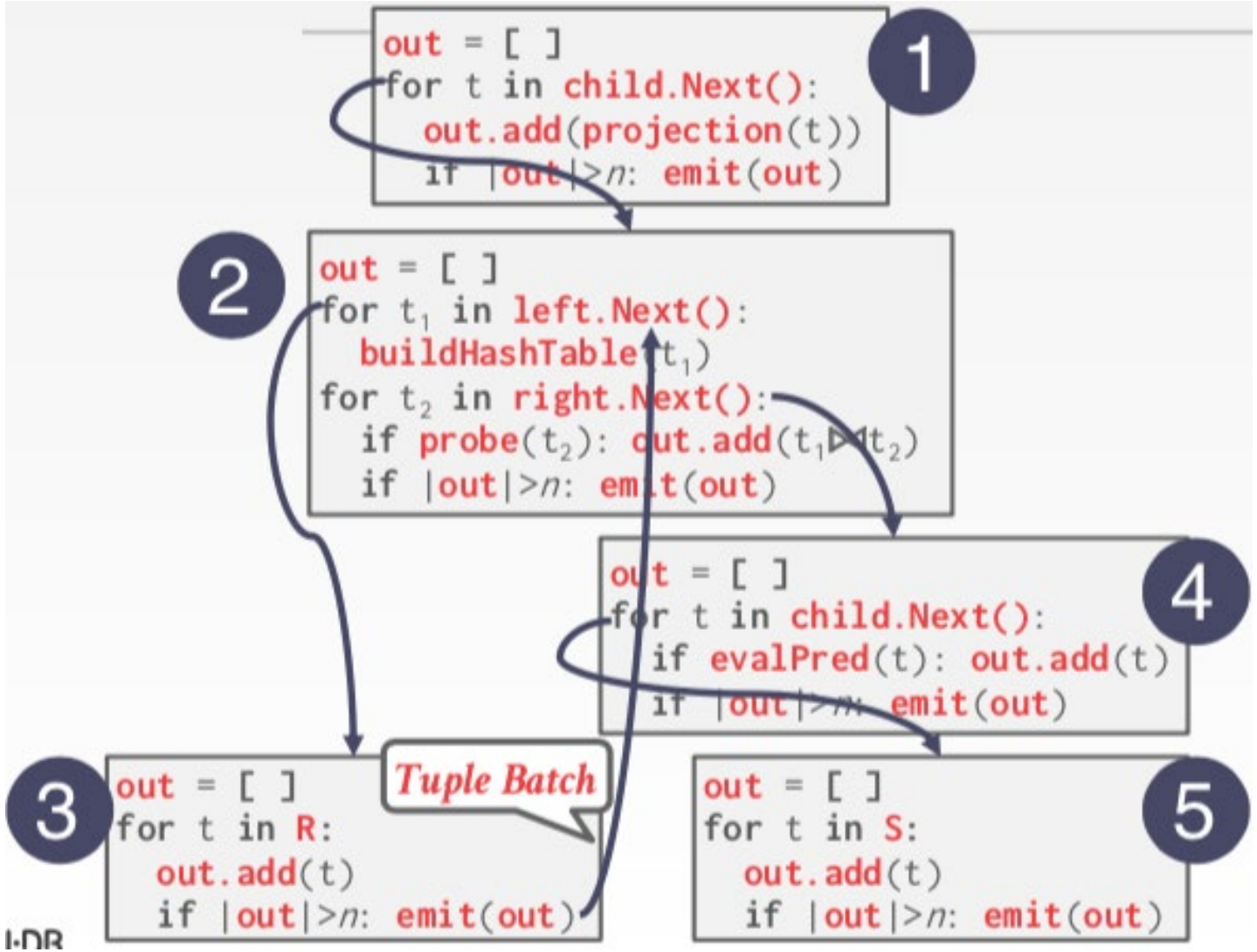
SCHOOL OF DATA SCIENCE

Each query plan operator implements an Output function:

- The operator processes all the tuples of its children at once.
- The return result of this function is all the tuples that the operator will emit.



Processing Model: Vectorization Model



- Like the Iterator Model where each operator implements a Next() function, but …
- Each operator emits a batch single tuple
  - The operator's internal loop processes multiple tuples at a time.
  - The size of the batch can vary based on hardware or query properties

Access Methods: Sequential Scan and Optimization

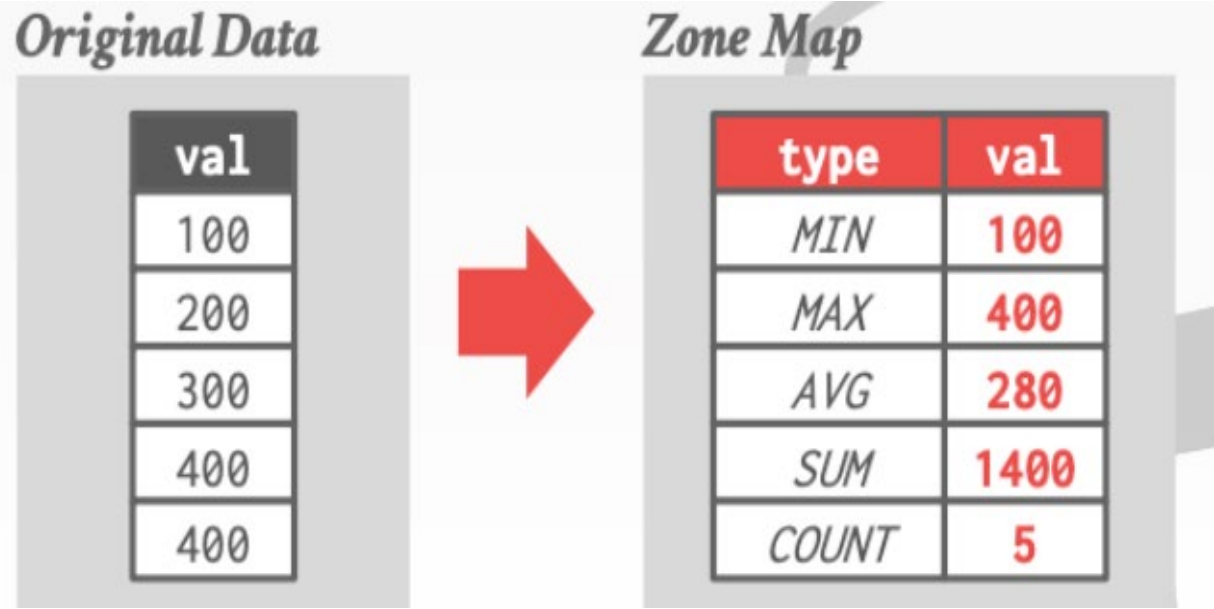
SCHOOL OF DATA SCIENCE

- A **sequential scan** is the simplest way to retrieve data—it reads tuples sequentially from the pages where the table is stored.
- The DBMS uses a cursor to track the position it last accessed (such as the page and slot).
- While this method works, it's often considered the **worst-case strategy** because it can be very inefficient.

```
for page in table.pages:
    for t in page.tuples:
        if evalPred(t):
            # do something
```

1.Zone MAPS

Pre-computed aggregates for the attribute values in a page. DBMS checks the zone map first to decide whether it wants to access the page.



SELECT * FROM TABLE WHERE val > 500

Access Methods: Sequential Scan and Optimization

SCHOOL OF DATA SCIENCE

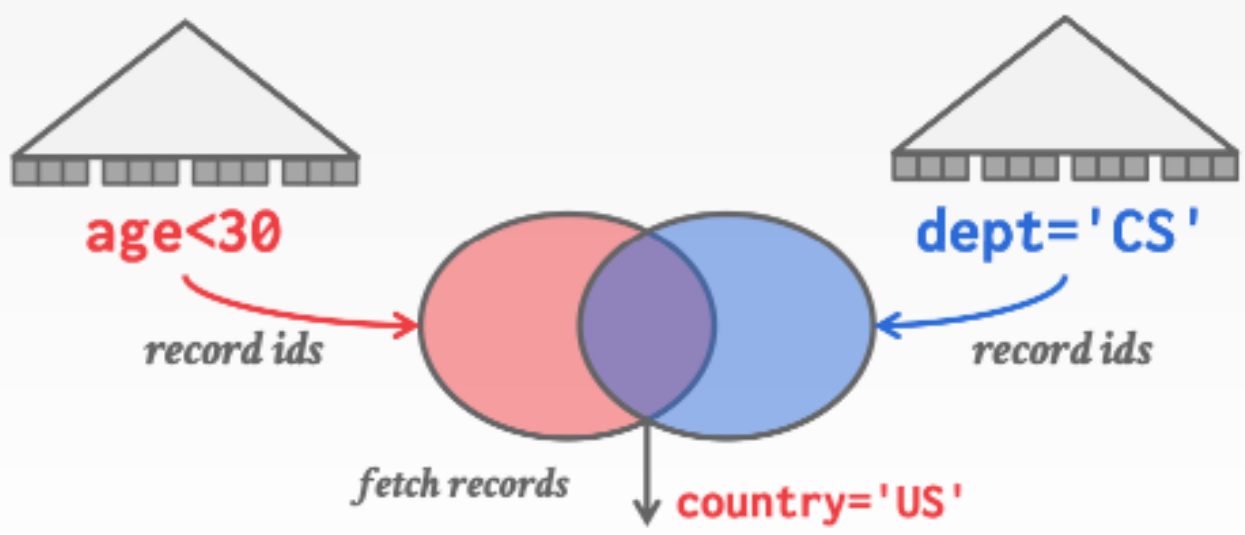
2. LATE MATERIALIZATION

- For columnar DBMS, we can delay passing data from one operator to another.
- If a certain column's data is not required further up in the query tree, we only need to pass the **offset** or the **column ID**.
- This approach allows us to retrieve the necessary data later, only when it is actually needed.

Access Methods: Multi-index Scan

SCHOOL OF DATA SCIENCE

- If there are multiple indexes that the DBMS can use for a query:
- Compute sets of Record IDs using each matching index.
- Combine these sets based on the query's predicates (union vs. intersect).
- Retrieve the records and apply any remaining predicates




```sql
SELECT * FROM students
 WHERE age < 30
   AND dept = 'CS'
   AND country = 'US';
```

Q&A