Database Administration

José Orlando Pereira

Departamento de Informática
Universidade do Minho

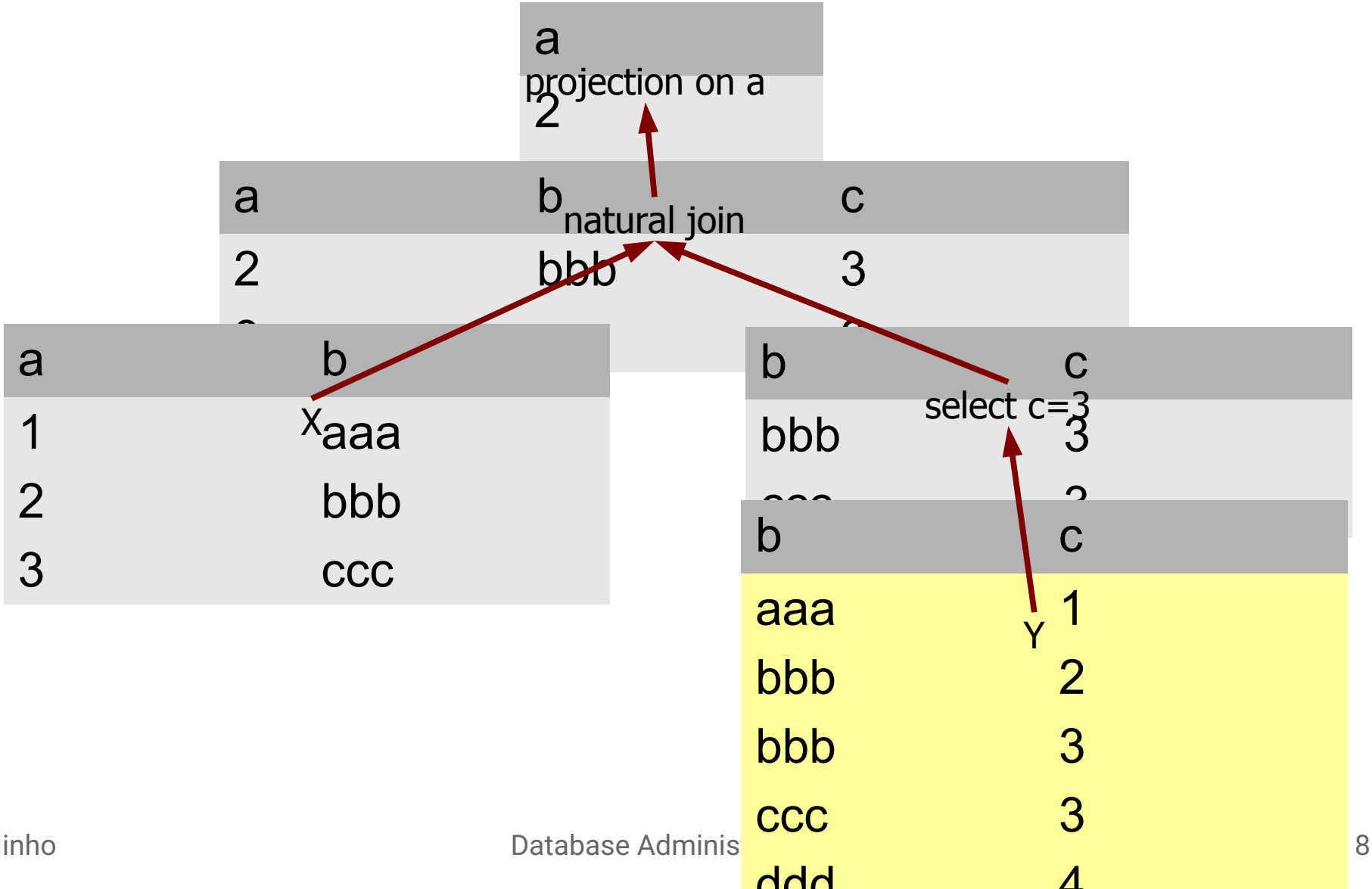
Roadmap

- What physical operators exist for each logical operation?
- How are physical operators implemented and composed?
- Later: How are physical operators selected?

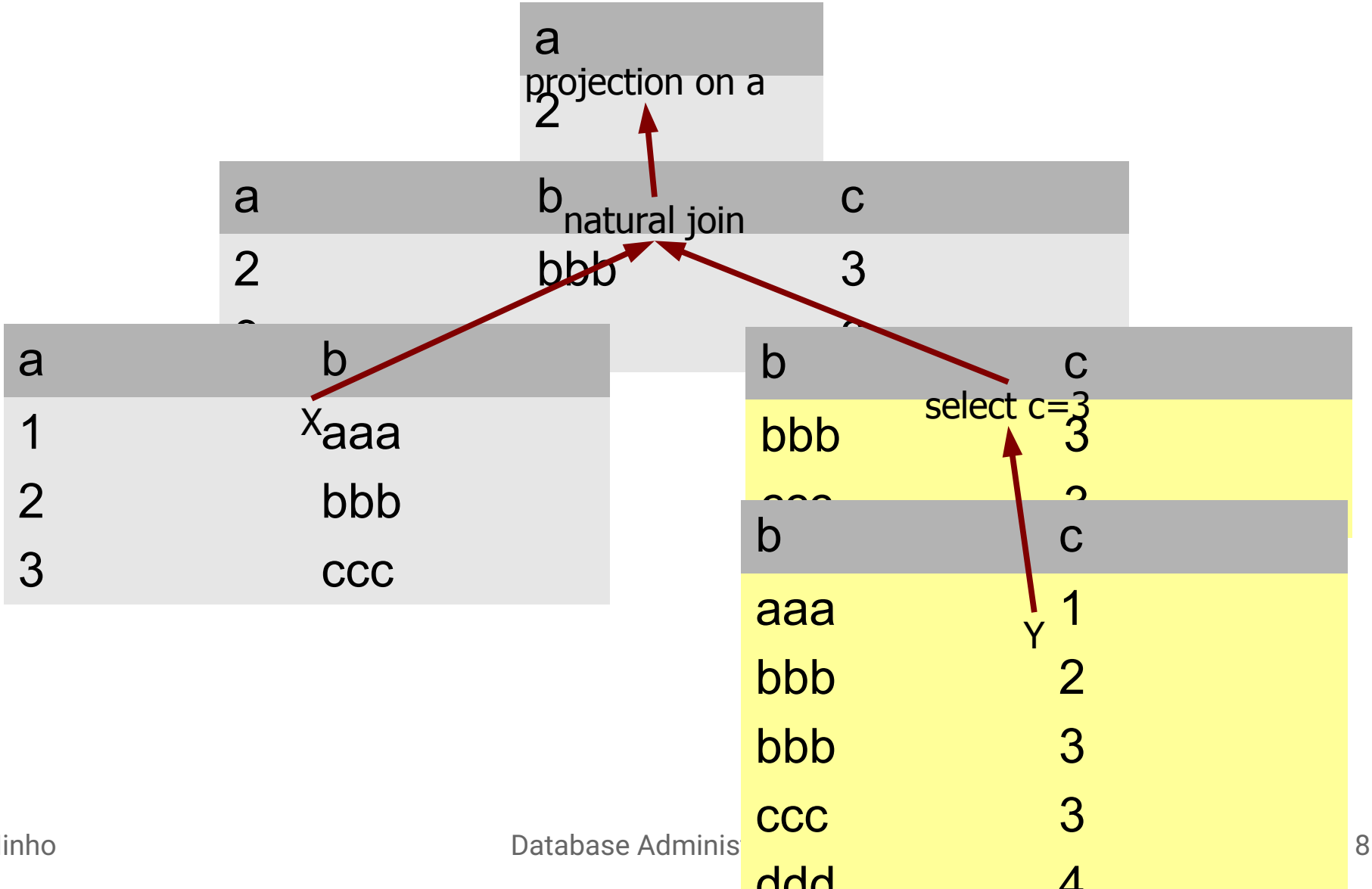
Execution with materialization

- Each operator is a function:
  - Returns a relation
  - Parameters are other relations (possibly, returned from operators)
- Computation order:
  - From leaves to root

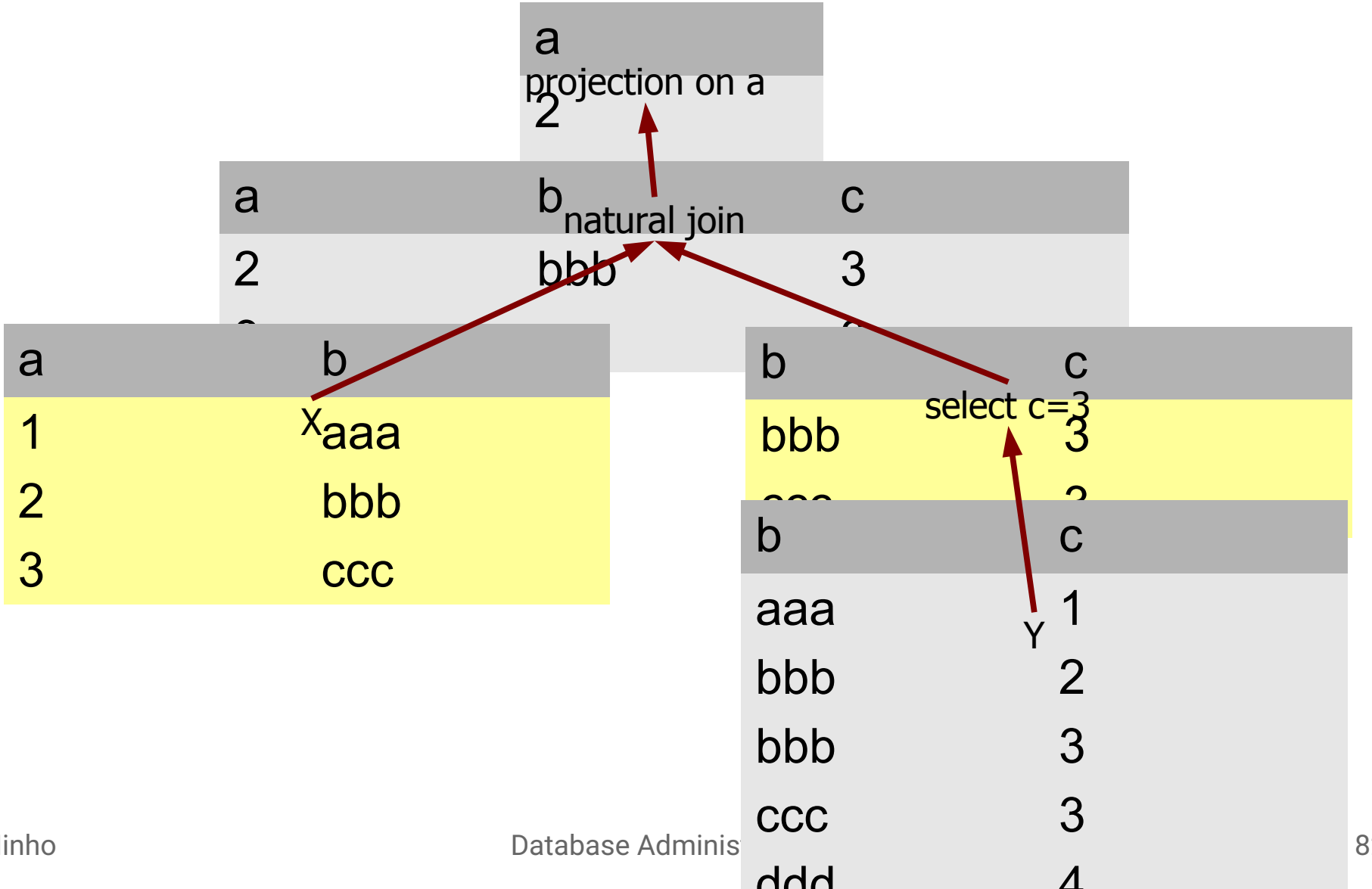
Execution with materialization



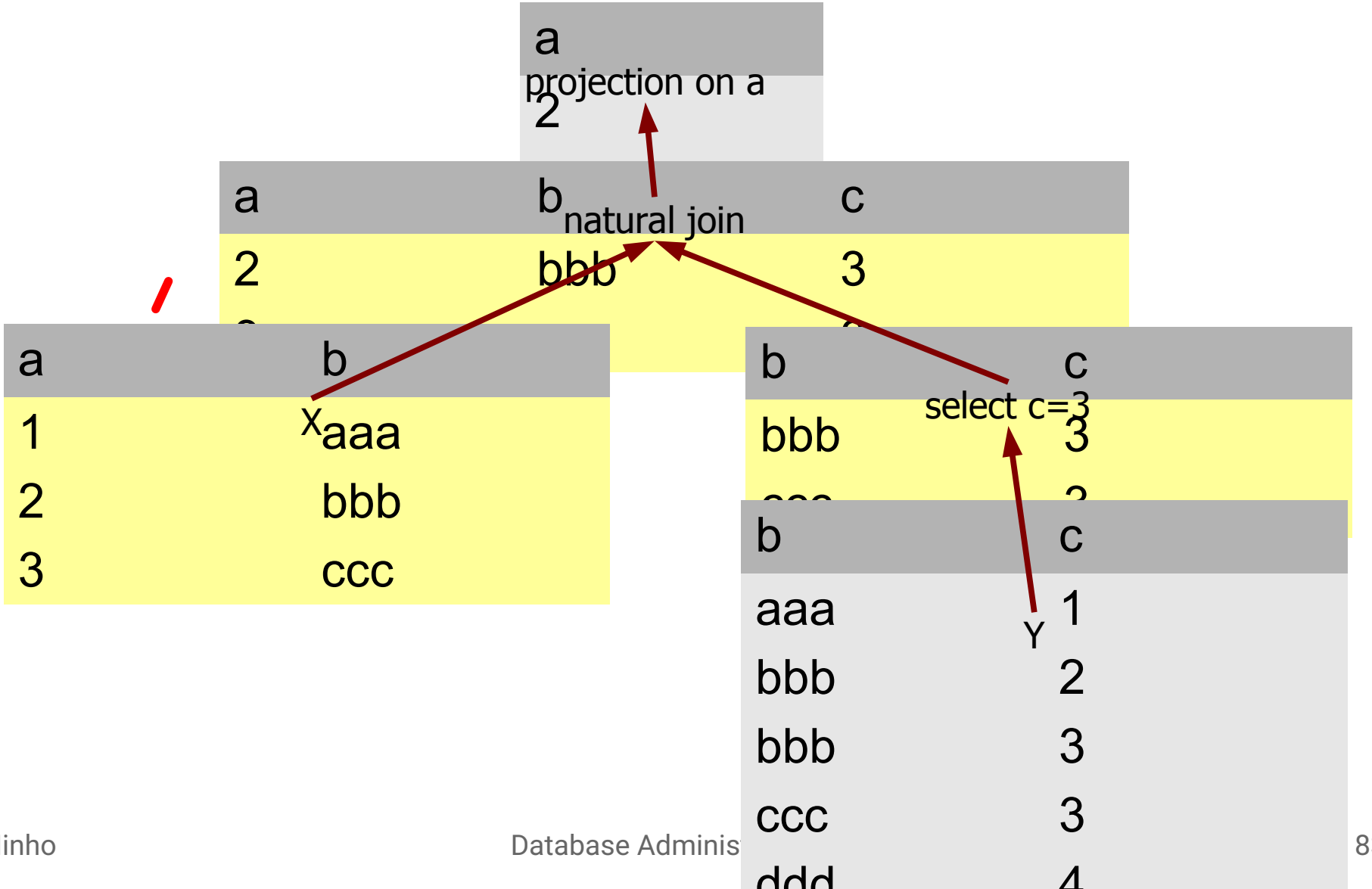
Execution with materialization



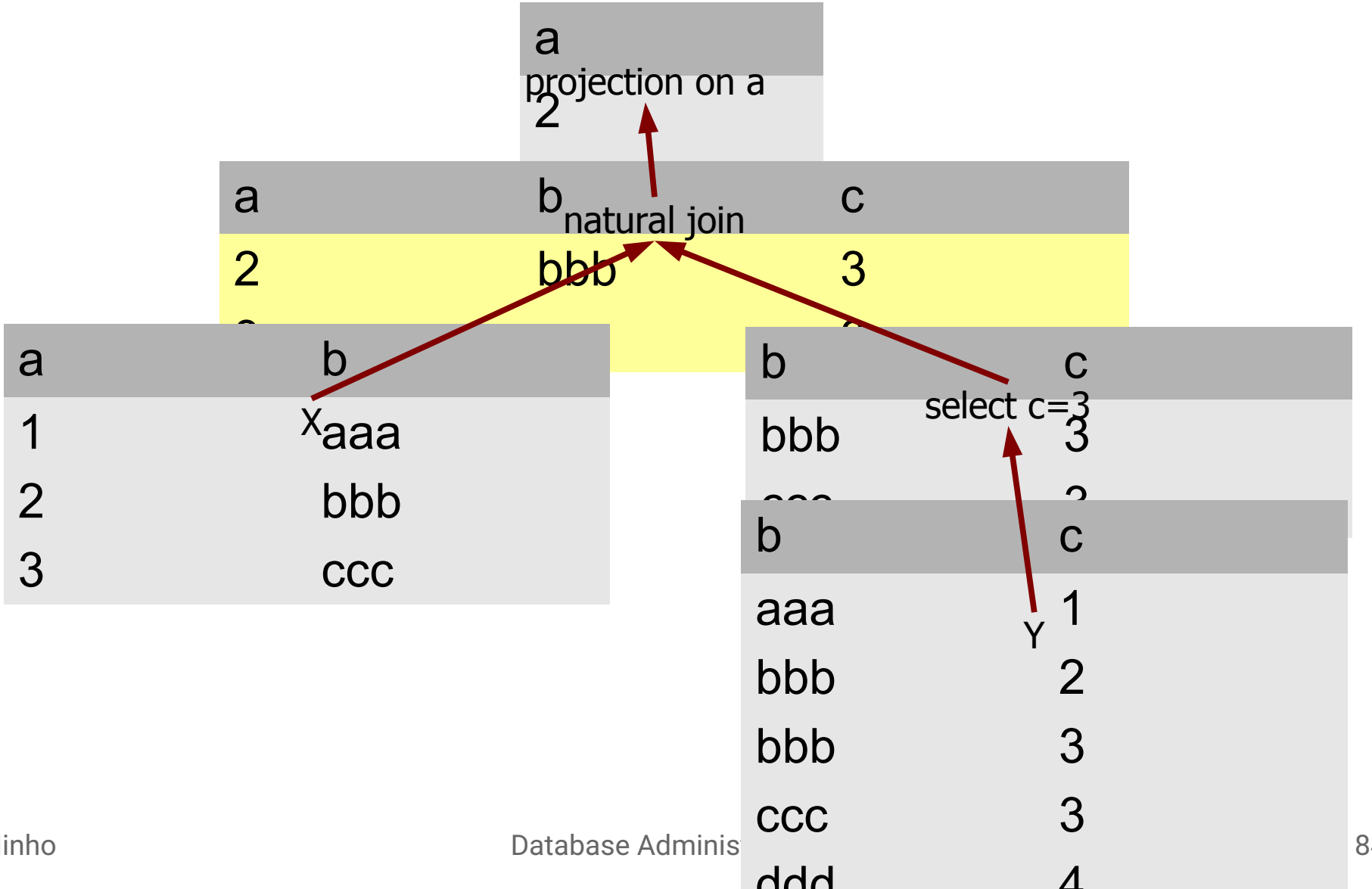
Execution with materialization



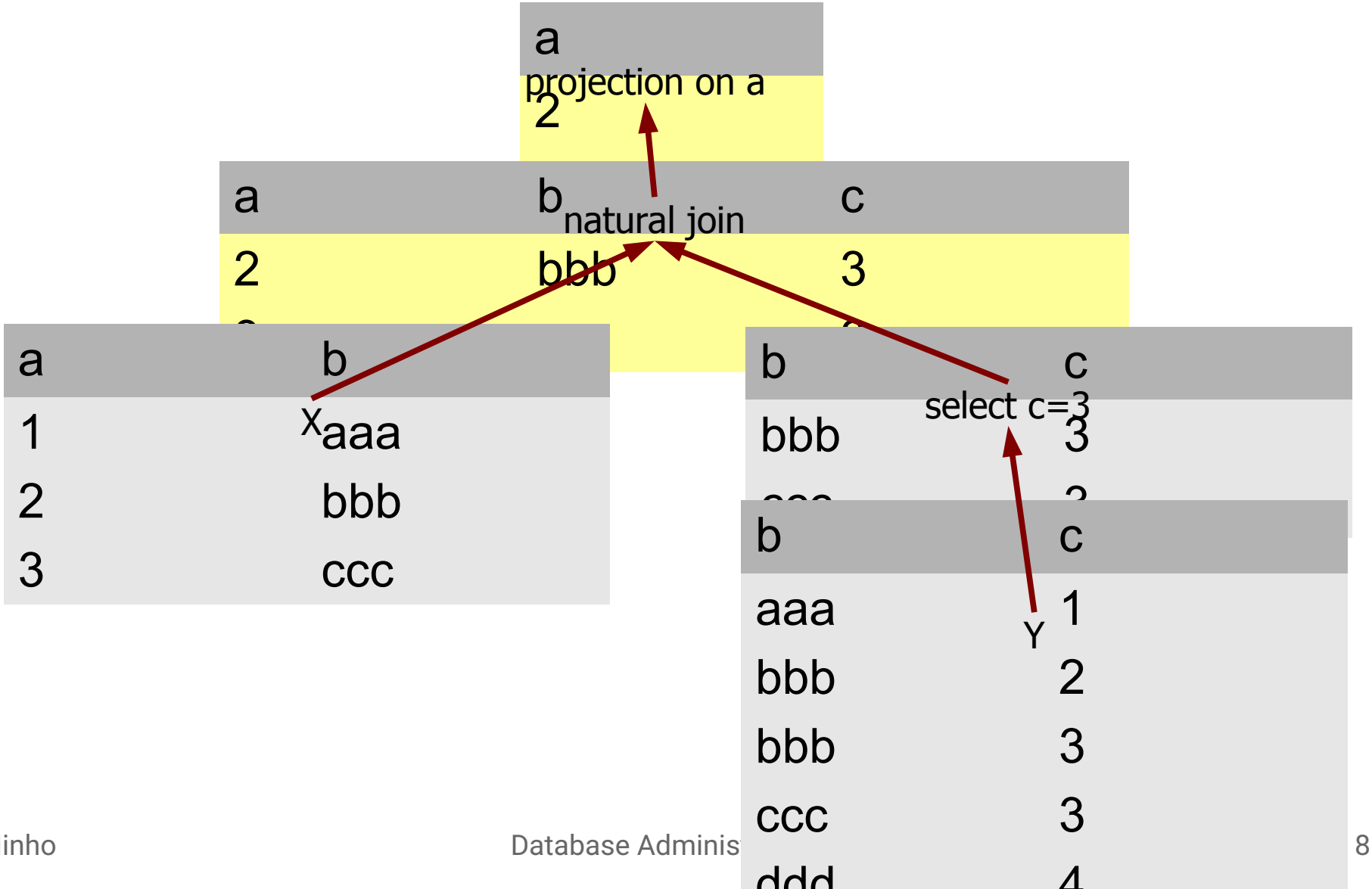
Execution with materialization



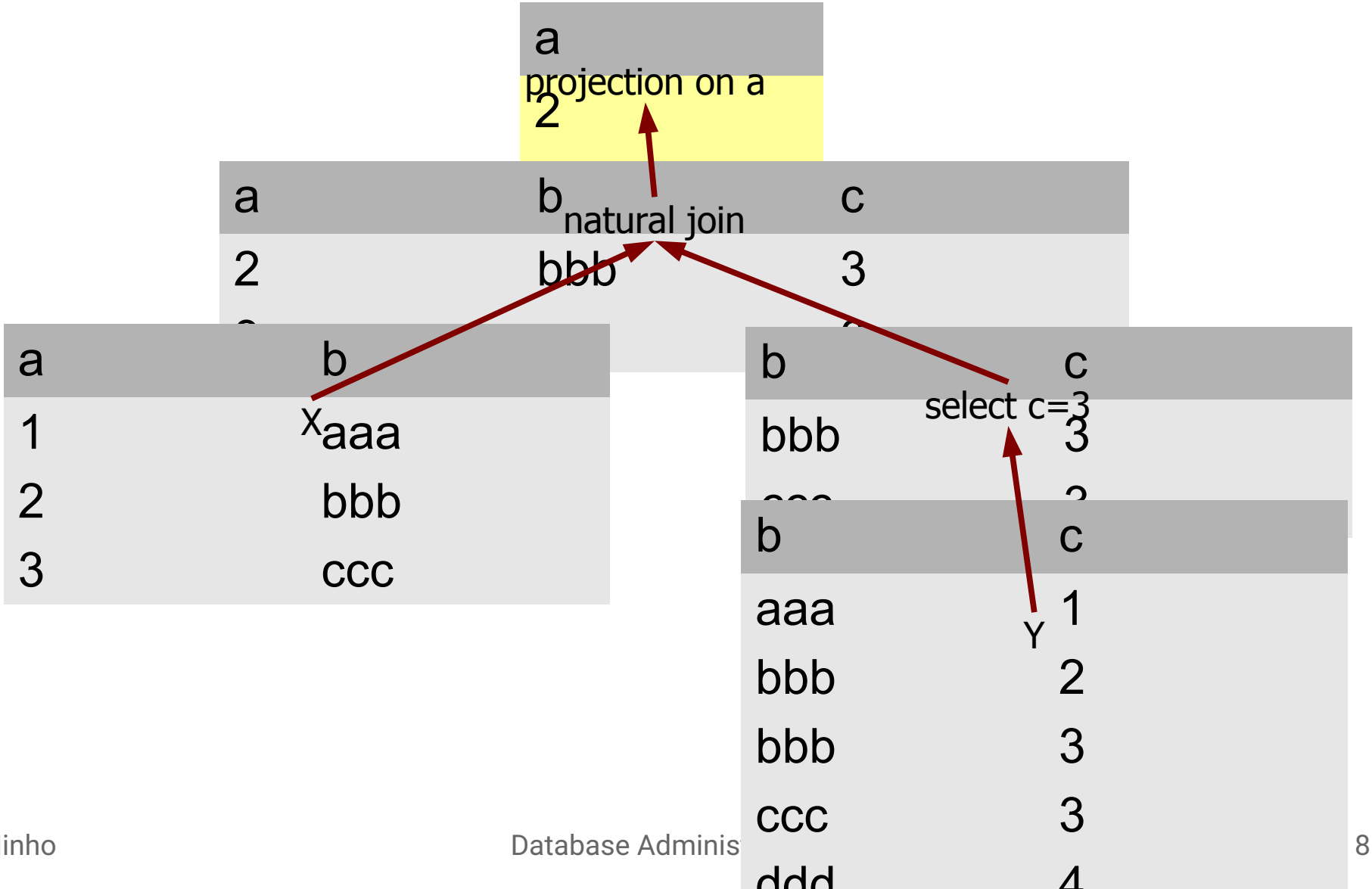
Execution with materialization



Execution with materialization



Execution with materialization



Consequences

- Large amount of intermediate results that need to be stored
  - Might not fit completely in memory
- Potentially wasted work
  - e.g. SELECT … LIMIT 10

Challenges

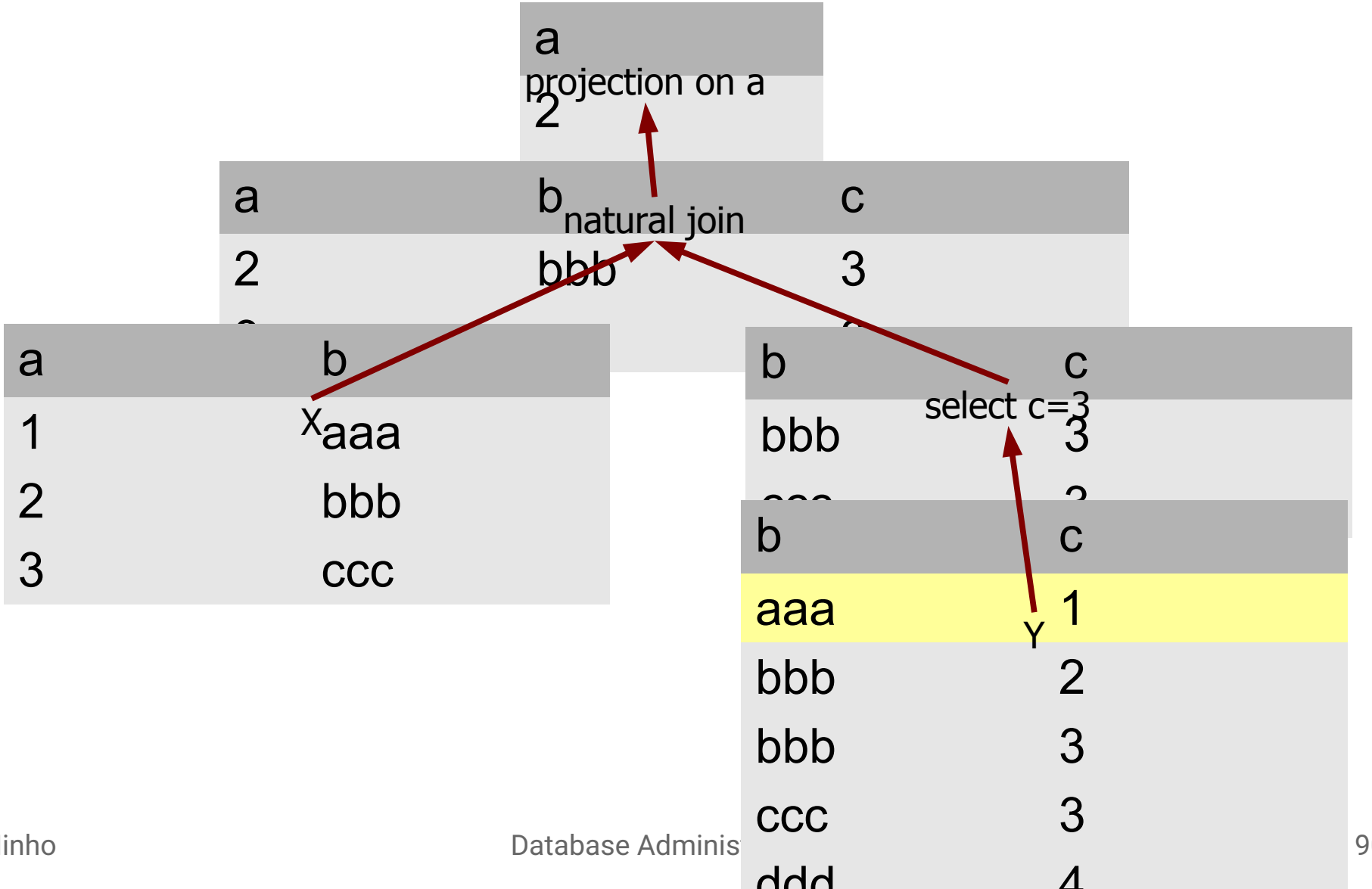
- How to minimize data movement while executing?

- How to achieve the best performance given current computer architectures?

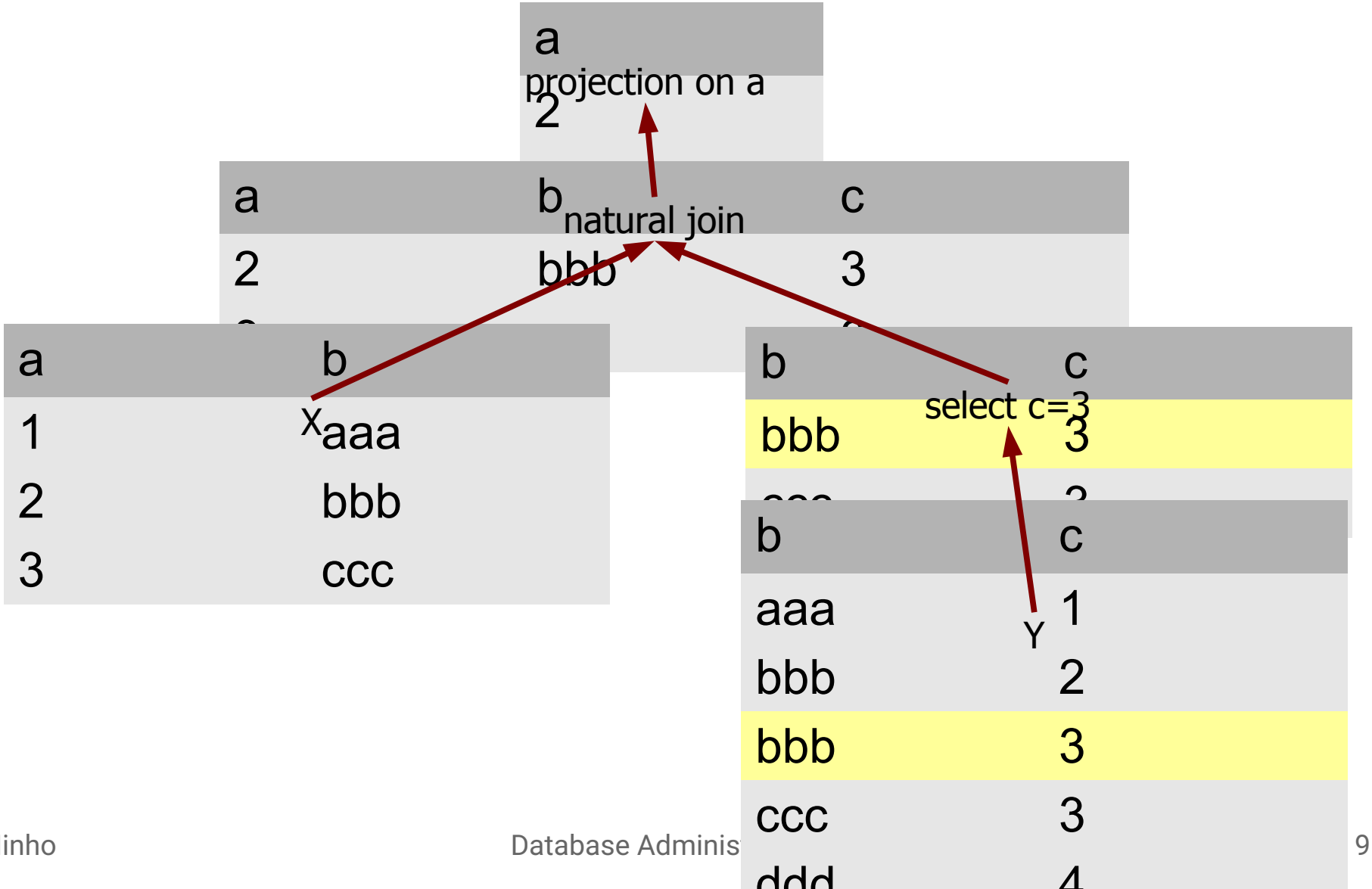
Iteration (pull model)

- Each operator is an object:
  - Interface similar to java.util.Iterator:
    - open() - get ready to return first record
    - next() - return next record
    - close() - no more records required
  - Constructor parameters:
    - Other operator objects
- Computation order:
  - From leaves to root, for each record

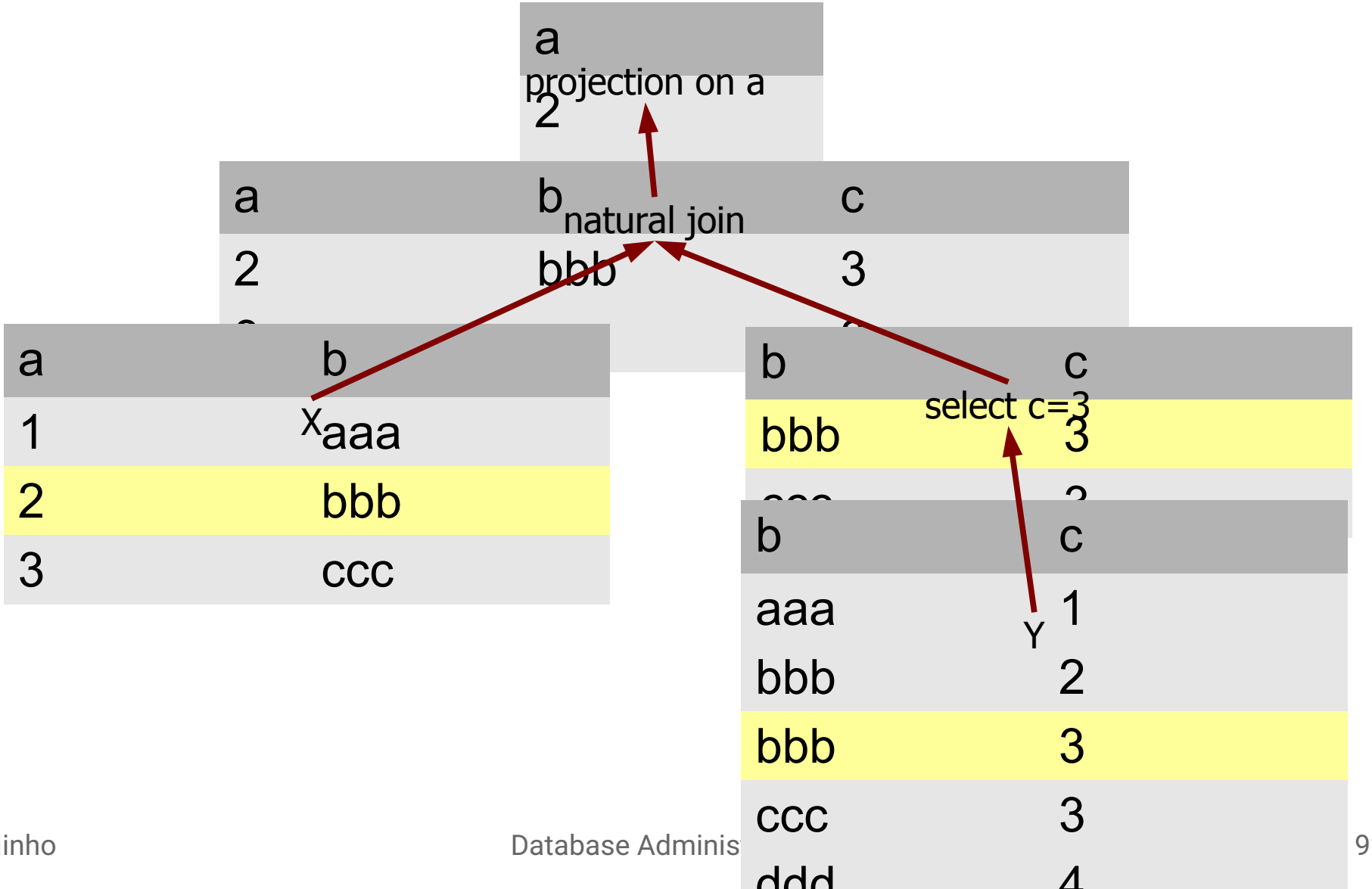
Execution with iteration



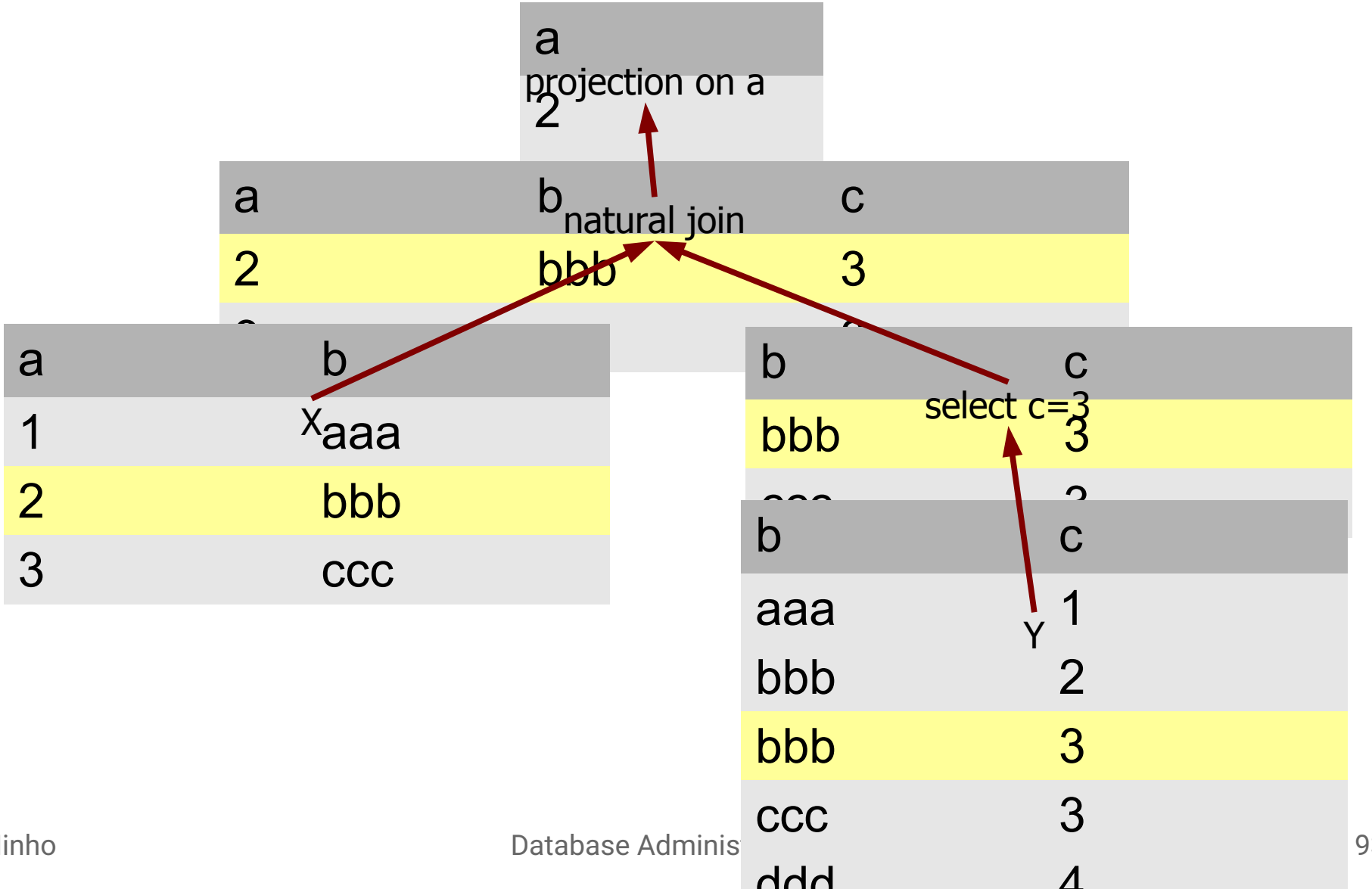
Execution with iteration



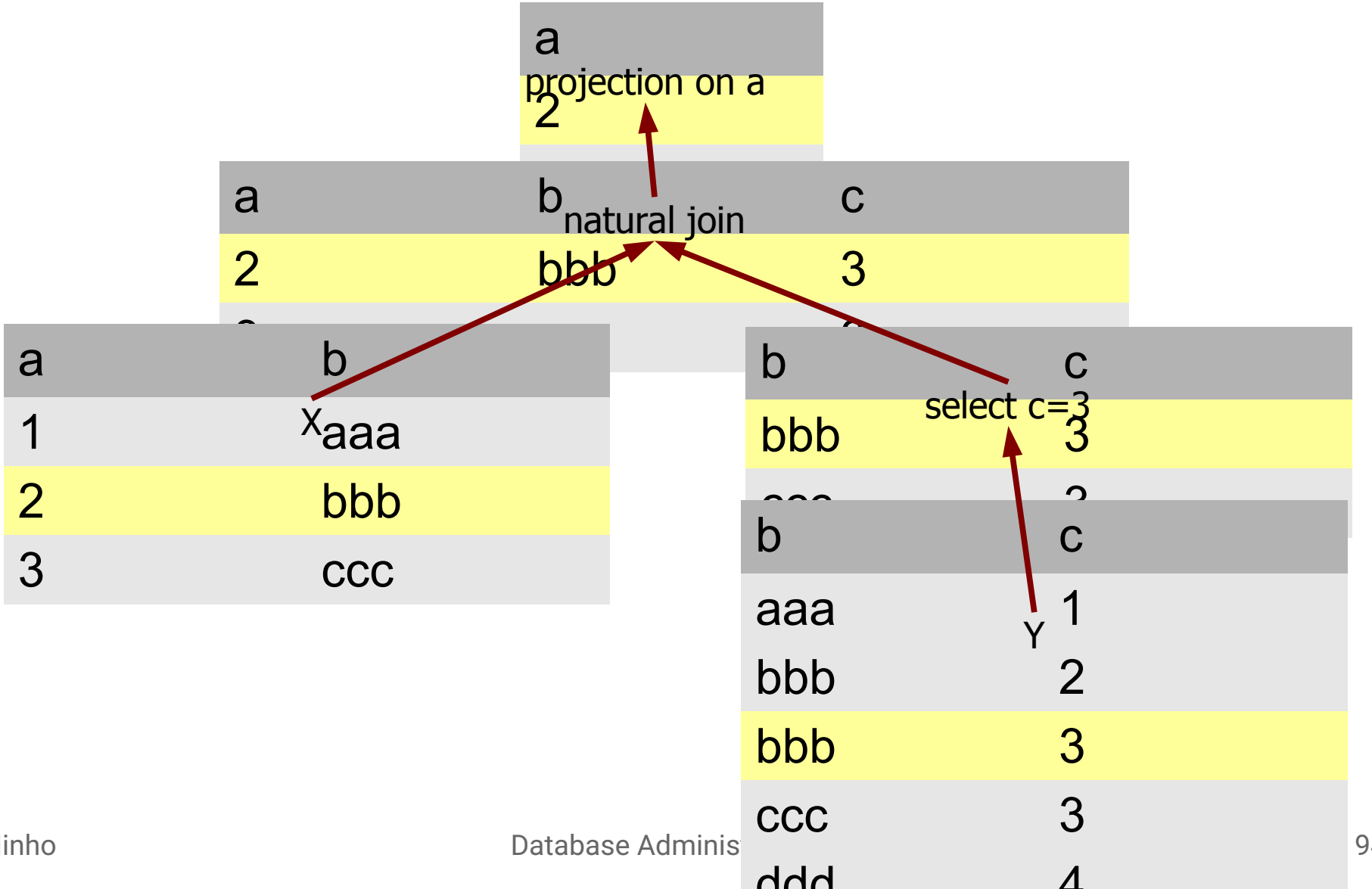
Execution with iteration



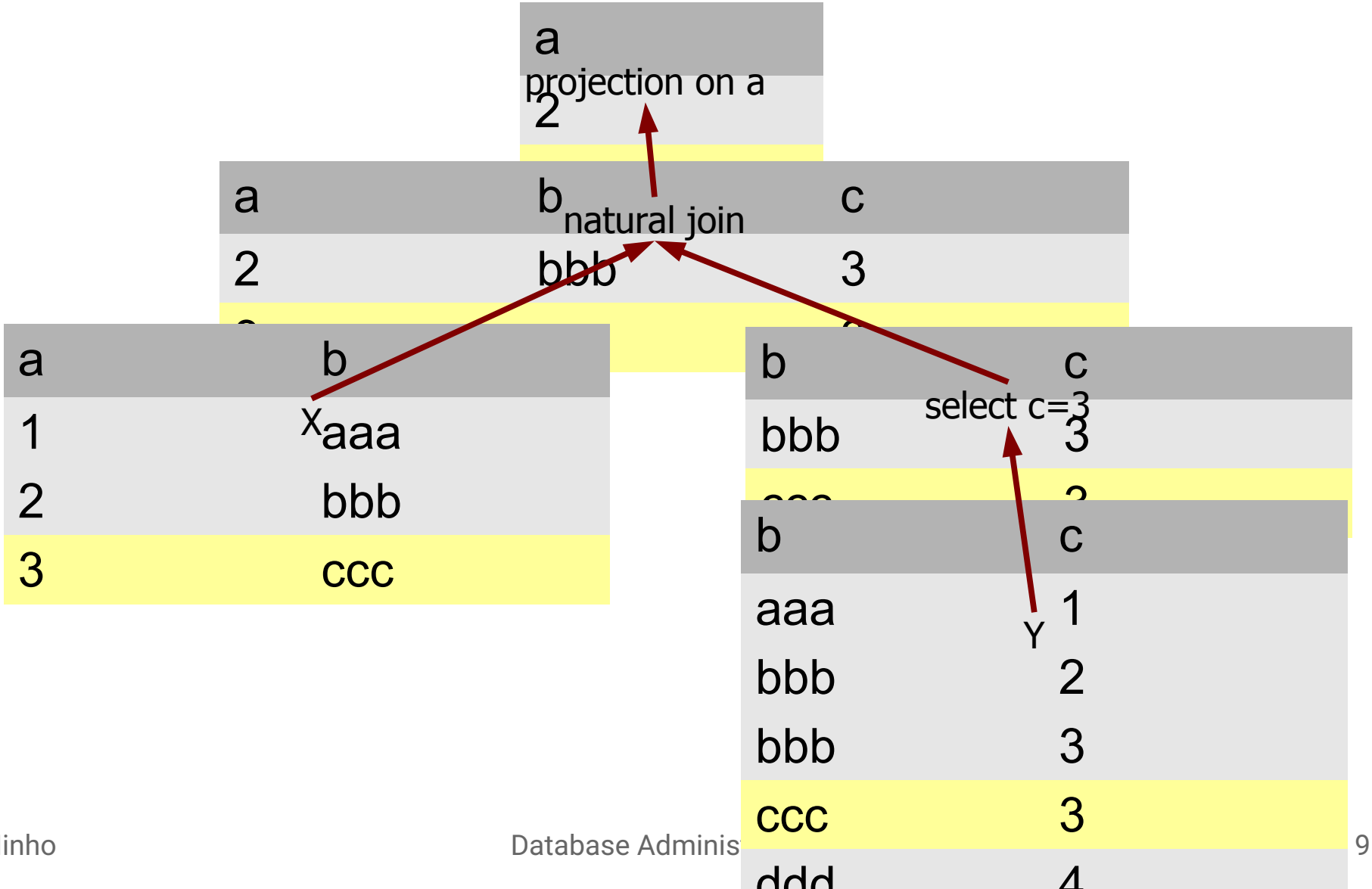
Execution with iteration



Execution with iteration



Execution with iteration



Consequences

- Not applicable to operators that must observe all rows before knowing what is the first to output
  - ORDER BY
  - GROUP BY on an unsorted input
  - …
  - In these cases, the operator executes all the underlying plan on open() or the first invocation to next()
- Minimizes work with LIMIT clause