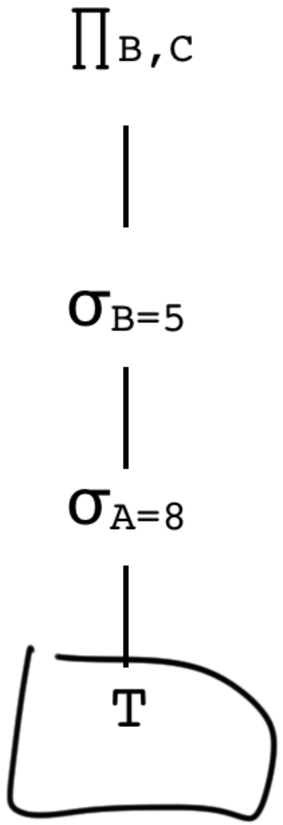
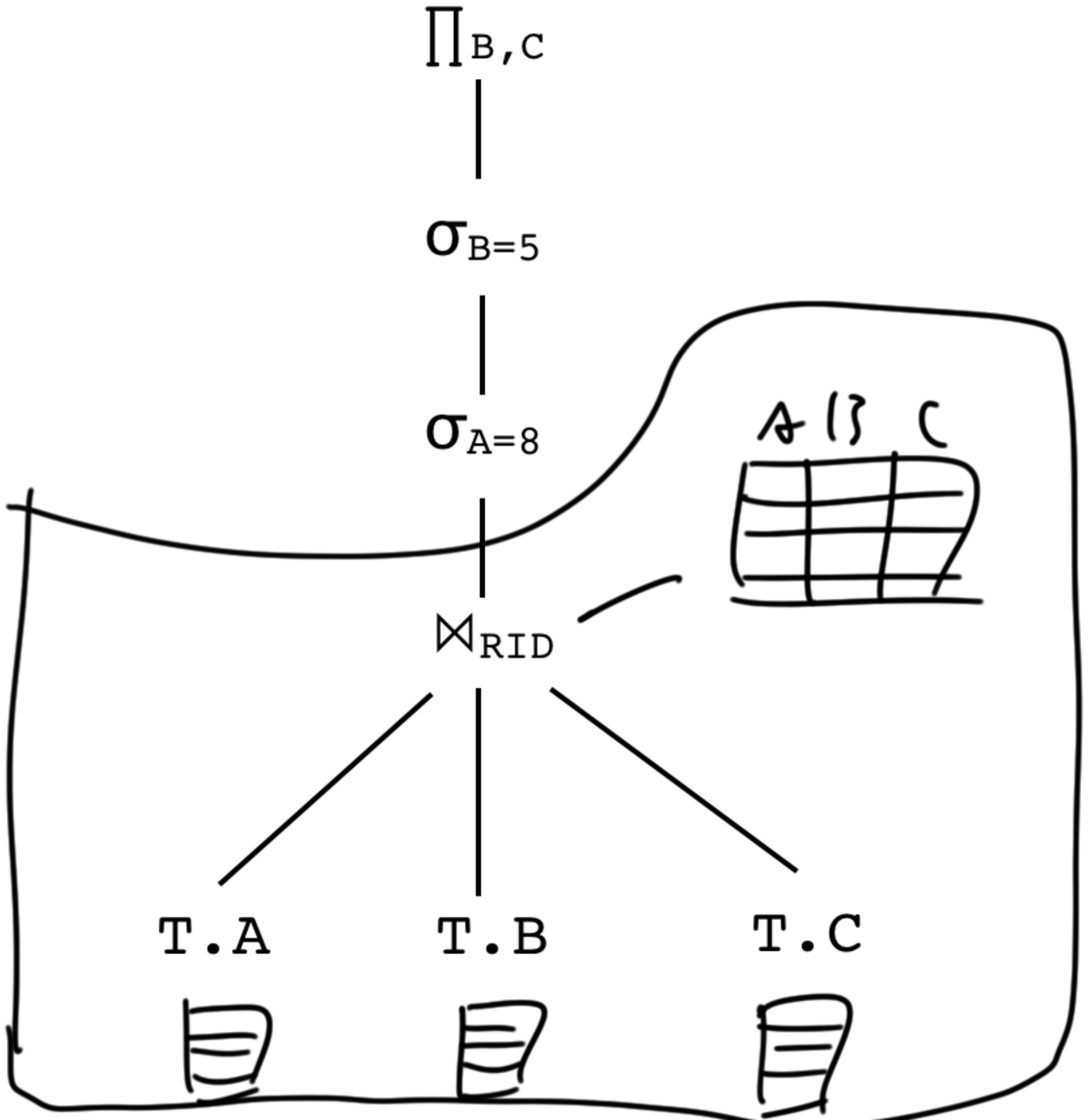
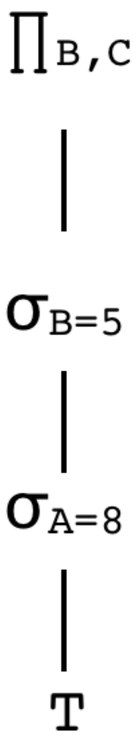
Early Materialization

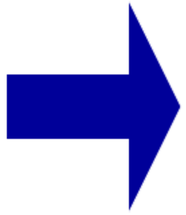


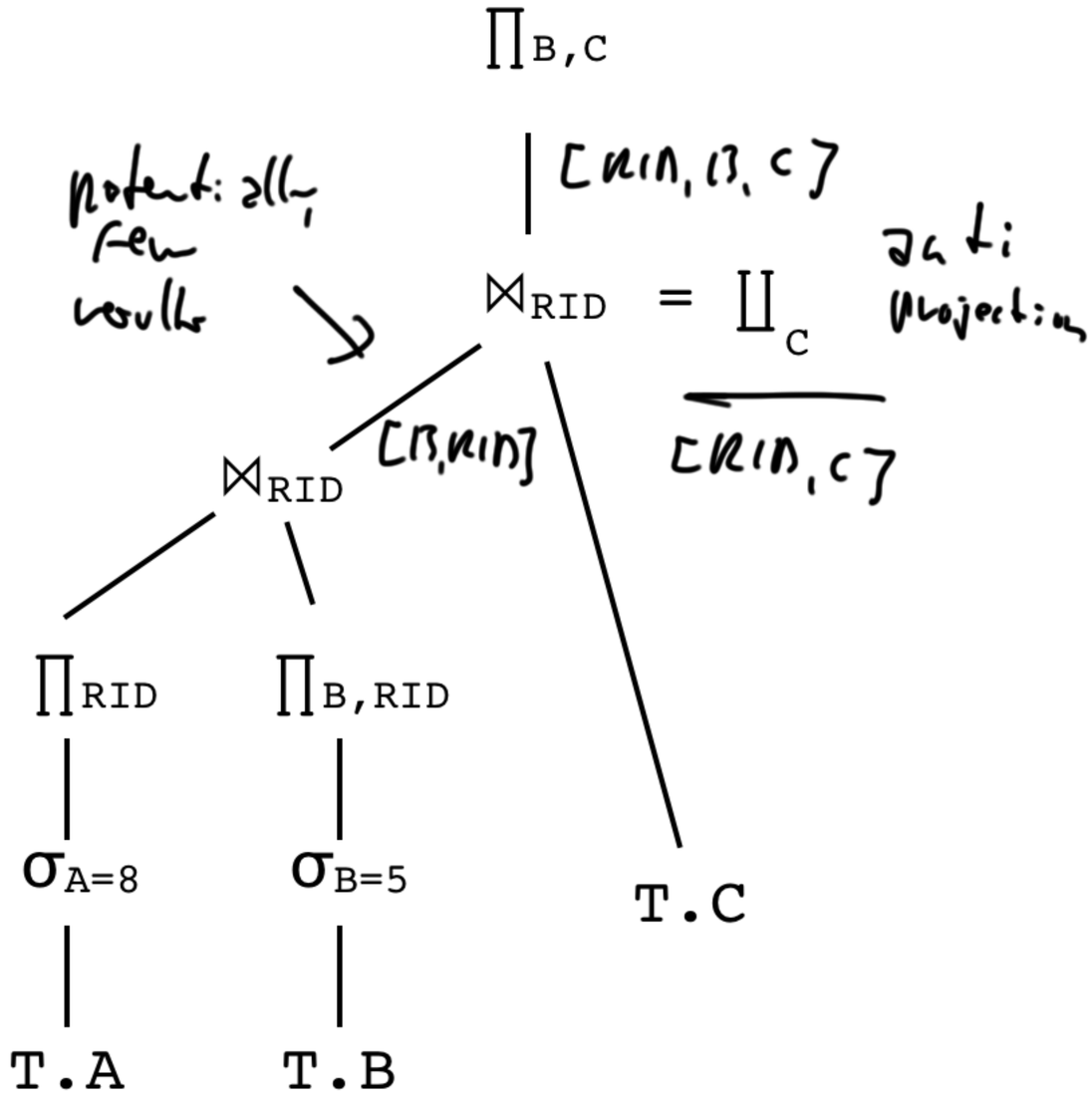



```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```

(Partially) Late Materialization

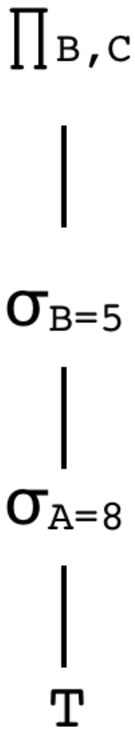


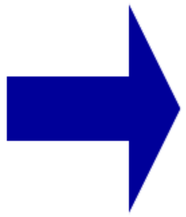


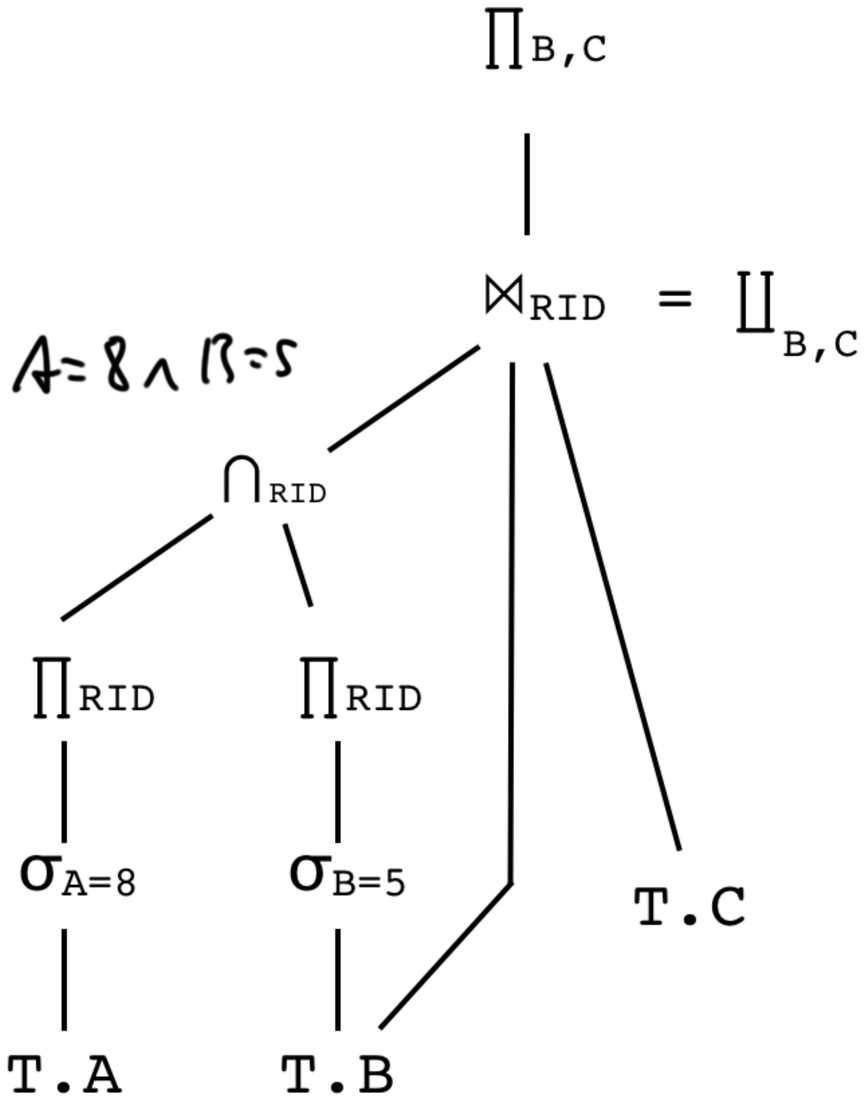



```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```

(Really) Late Materialization



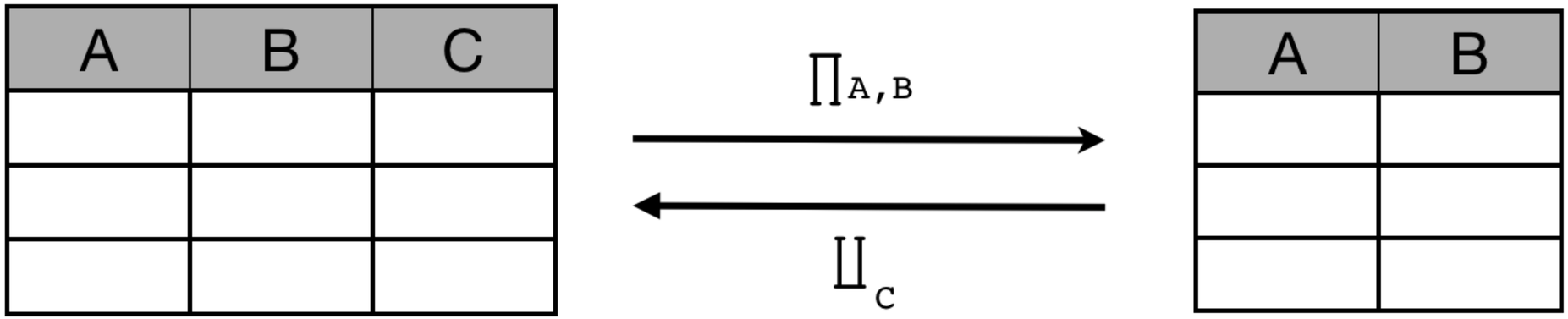





```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```

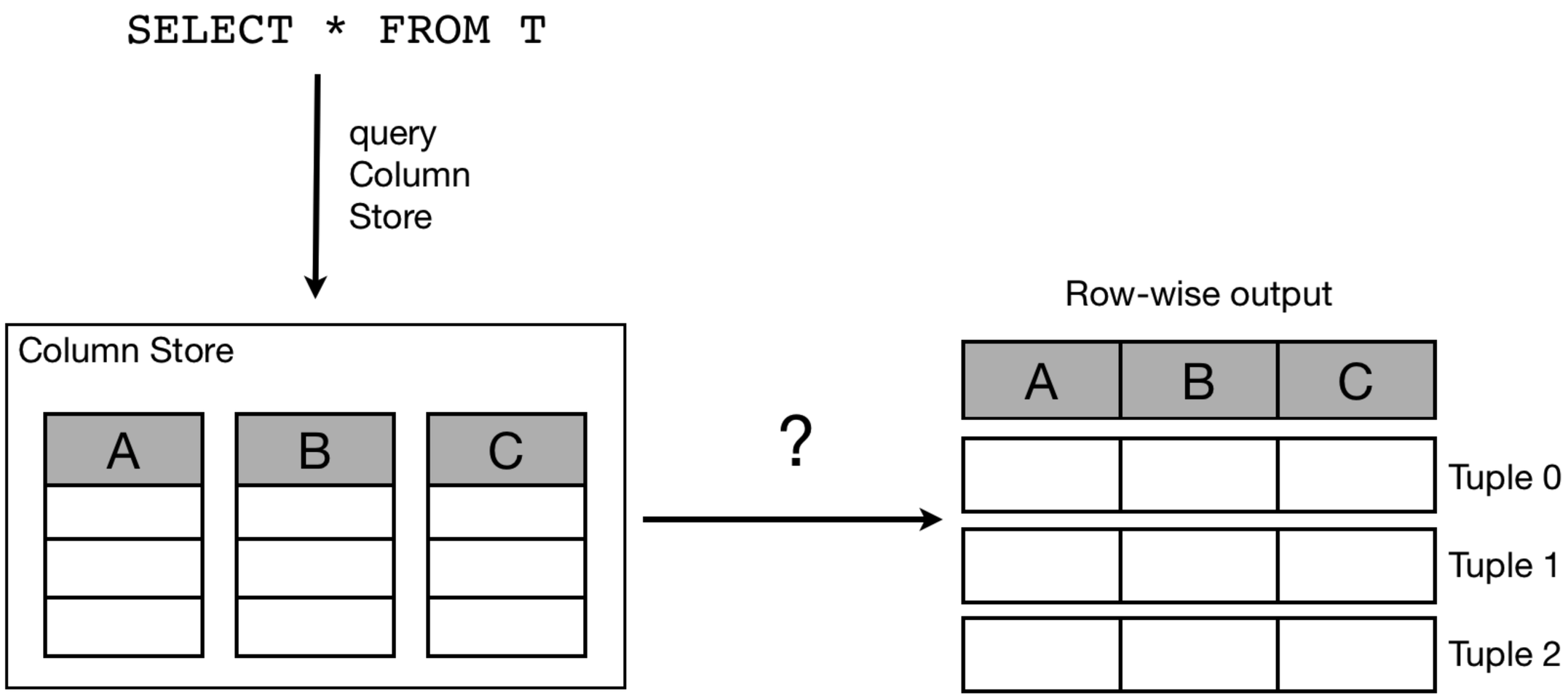
Projection vs “Anti-Projection“

When to narrow tuples?



When to widen tuples?

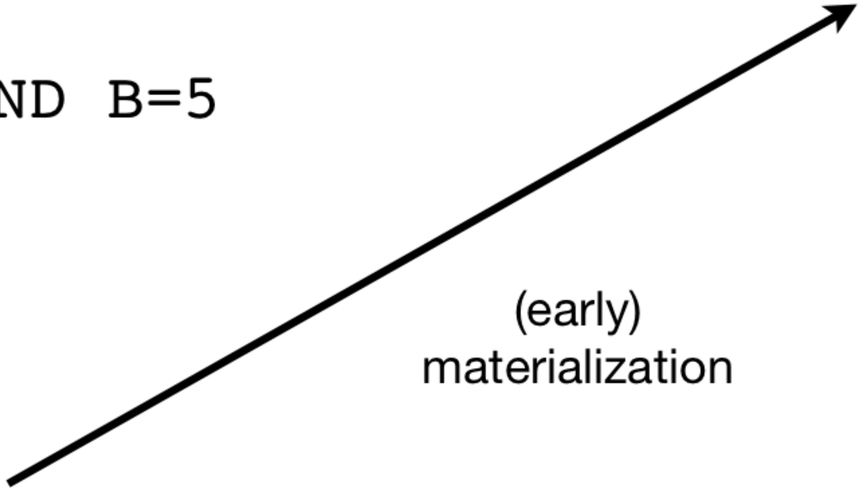
Example: Tuple Reconstruction in Column Stores



Implementing Early Materialization

```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```

query request



intermediate representation

A	B	C
3	2	7
8	5	3
8	2	9

$\sigma_{\texttt{A=8 AND B=5}}$

$\prod_{\texttt{B,C}}$

A	B	C
3	2	7
8	5	3
8	2	9

column store

B	C
5	3

output

Implementing Late Materialization (1)

```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```

query request

A	B	C
3	2	7
8	5	3
8	2	9

column store

selection
on A and B

A
3
8
8

B
2
5
2

input columns

label

$\sigma_{A=8}$
0
1
1

$\sigma_{B=5}$
0
1
0

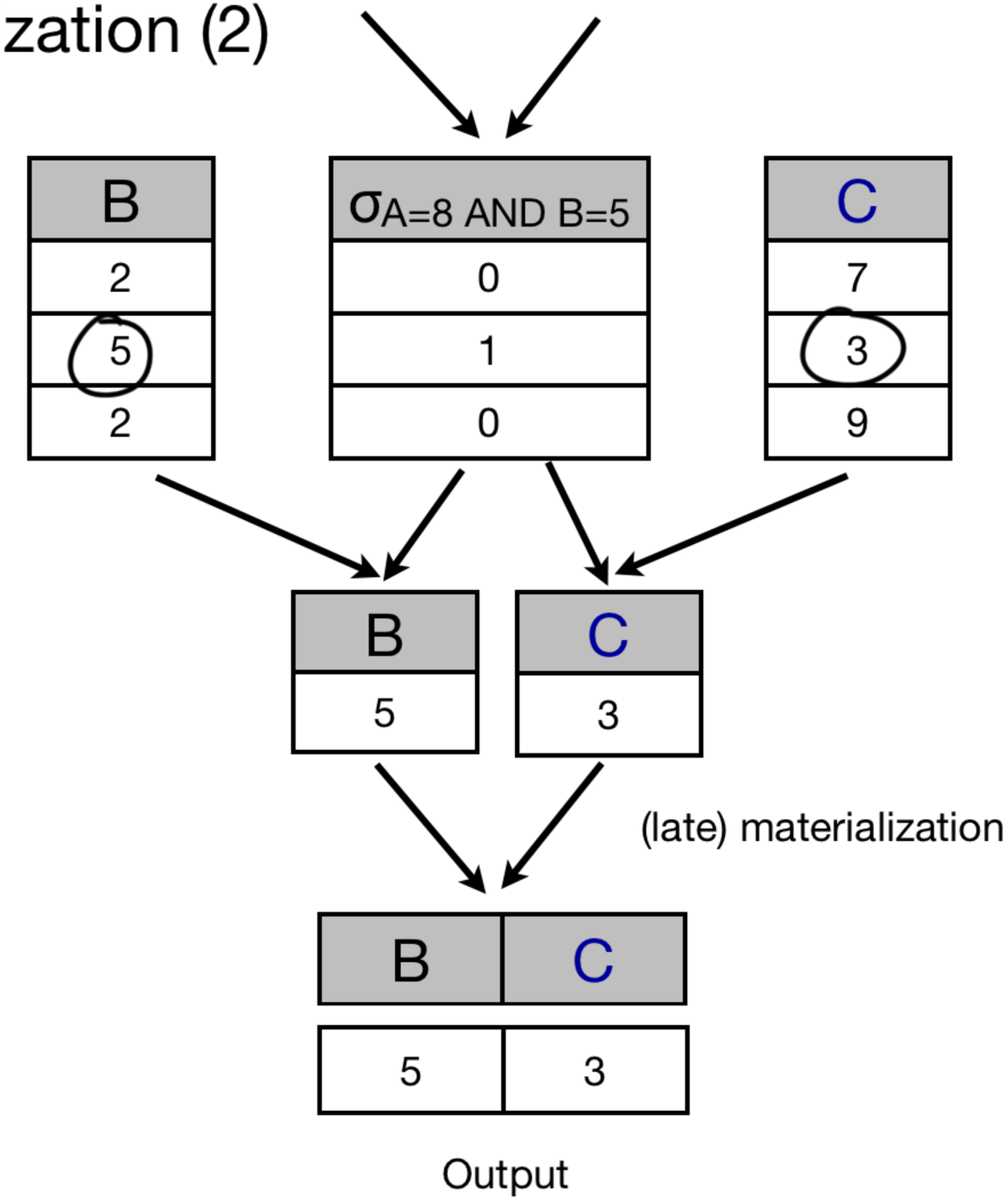
bitvector
marking
qualifying
entries

$\sigma_{A=8 \text{ AND } B=5}$
0
1
0

conjunction
bitvector

Implementing Late Materialization (2)

```
SELECT B,C
FROM T
WHERE A=8 AND B=5
```



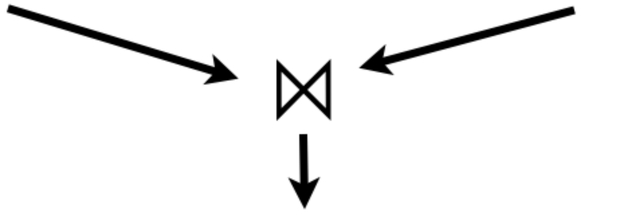
Joins using Early Materialization

```
SELECT T.B,S.C
FROM   T,S
WHERE  T.A = S.A
```

Input to join has been materialized already!

T	
B	A
7	3
5	4
6	7
4	4
3	2
1	2

S	
A	C
2	3
3	4
1	7
1	4
9	2
8	2



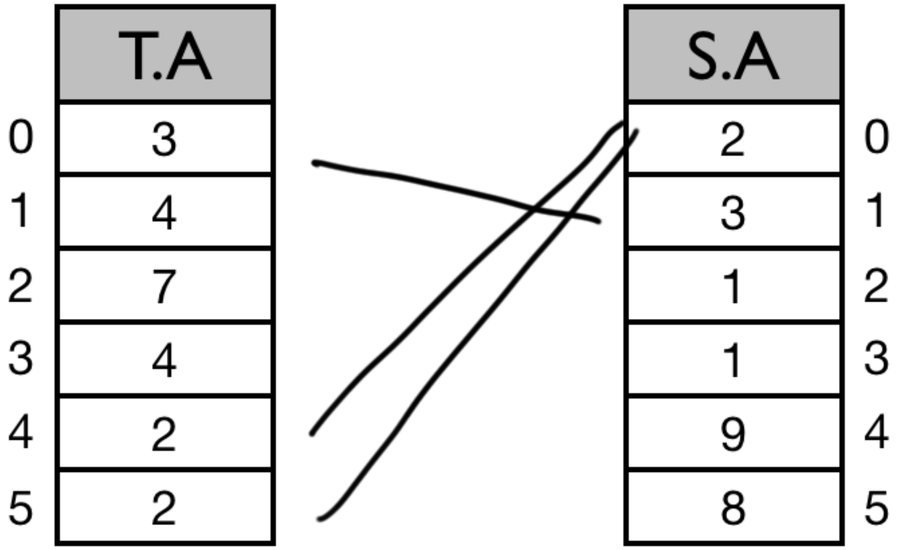
A	B	C
3	7	4
2	3	3
2	1	3

$\prod_{B,C}$

B	C
7	4
3	3
1	3

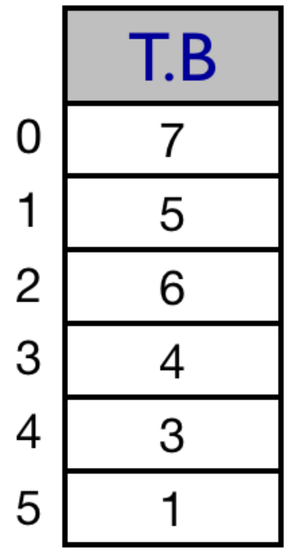
Joins using Late Materialization

```
SELECT T.B,S.C
FROM   T,S
WHERE  T.A = S.A
```

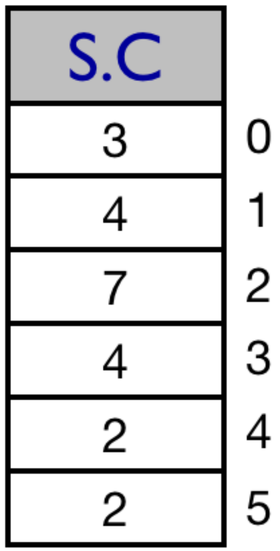


	T.A
0	3
1	4
2	7
3	4
4	2
5	2

S.A	
2	0
3	1
1	2
1	3
9	4
8	5



	T.B
0	7
1	5
2	6
3	4
4	3
5	1



S.C	
3	0
4	1
7	2
4	3
2	4
2	5

$\prod_{B,C}$

removing RID

$\prod_{B,C}(\coprod_{B,C})$

join index

Pos(T.A)	Pos(S.A)
0	1
4	0
5	0

T.B	S.C
7	4
3	3
1	3

Early Materialization

Advantages:

no re-access of columns necessary → bottom-up

Early Materialization

Advantages:

- no re-access of columns necessary
- easier planning

Disadvantages:

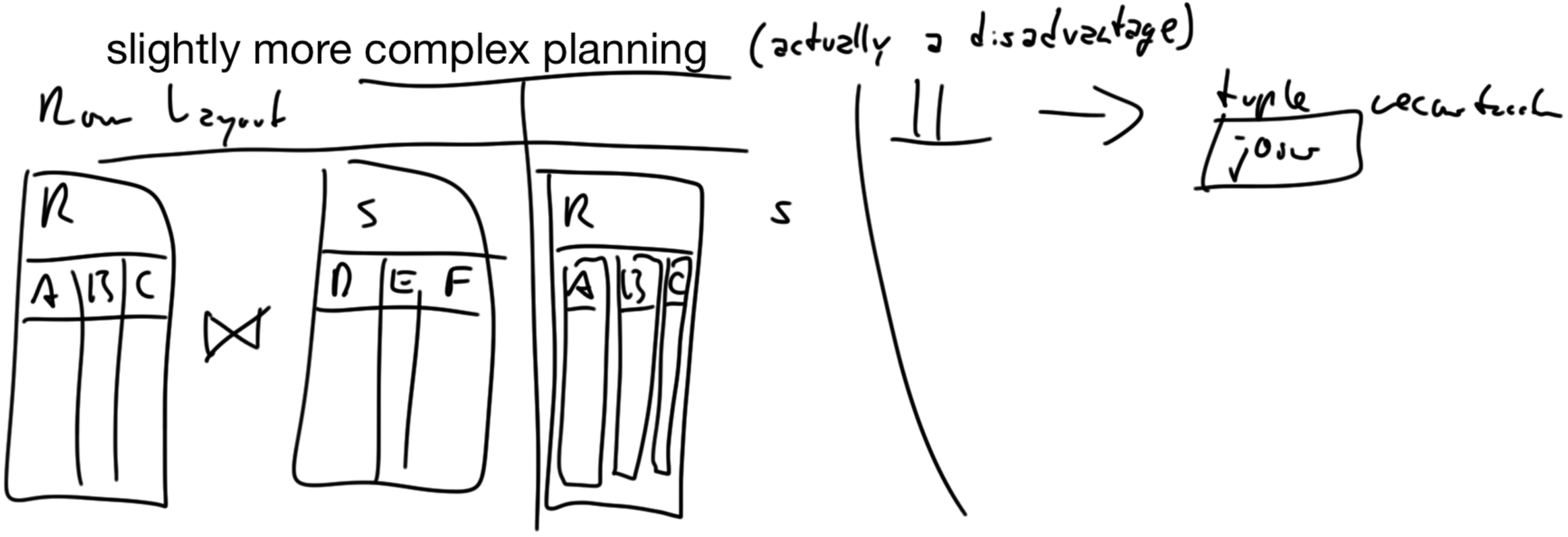
- possible generation of wide intermediate results

Late Materialization

Advantages:

constructing tuples only when necessary

slightly more complex planning (actually a disadvantage)



Late Materialization

Advantages:

- constructing tuples only when necessary
- slightly more complex planning

Disadvantages:

- re-access of columns possible

When to prefer what?

	Early Materialization	Late Materialization
Selectivity	Low (many entries selected)	High (few entries selected)
Compression	No	Yes
Aggregation	No	Yes