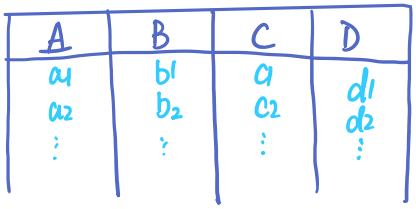
Partitioning

Friday, May 24, 2019 12:18 PM

- Vertical partitioning



R1(A, B, C), R2(A, D)

  - CREATE VIEW Name AS

```
SELECT R1.A, R1.B, R1.C, R2.D
FROM R1, R2
WHERE R1.A = R2.A;
```

  - difference with with...as: create view doesn't copy the data.
  - Advantage: Speeds up queries that touch only a small faction of columns;

    Single column can be compressed effectively, reducing disk I/O;
  - Disadvantage:

    Updates expensive

    Need many joins to access many columns

    Repeated key columns add overhead

- Horizontal Partitioning

```
CREATE VIEW dSet AS
    (SELECT A, B, C, d1 AS dName
     FROM Rhasd1)
     UNION ALL
    (SELECT A, B, C, d2 AS dName
     FROM Rhasd2)
     UNION ALL
       :

⇒ SELECT A
  FROM Rhasd2
```

o Application

- Performance optimization

  → data warehousing

- Distributed and parallel databases