

データベース技術

- Network Database, Hierarchical Database
1950年代～
- 関係データベース
1970年代～
- オブジェクトデータベース
1980年代～

関係データベース

- 関係データベースのデータの単位：**テーブル**
表の形式
- テーブルの各行：**タプル** (tuple)
1つのデータのまとめ
- テーブルの各列：
同じタイプのデータ（属性）が並んでいる。

関係データベースの例

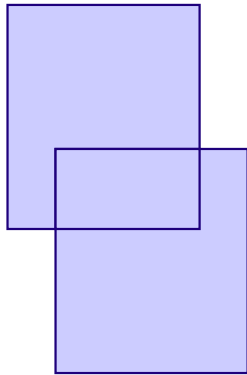
- スキーマ例

(COMP, SSN, FNAME, LNAME, STREETNUM, STREETNAME, CITY, STATE, ZIP)

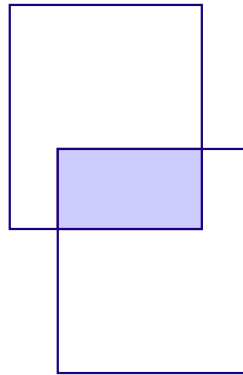
COMP	SSN	FNAME	LNAME	STREETNUM	STREETNAME	CITY	STATE	ZIP
ABC Corp.	99278	John	Simth	27	Canal St.	Fairfax	VA	22087
ABC Corp.	28745	Denia	Jones	786	Baker St.	Manassas	VA	22185
ABC Corp.	54892	Jane	Fox	1224	Cowper Dr.	Bethesda	MD	20984
ABC Corp.	98374	Kisa	Baenea	17	Edgar St.	Rockville	MD	20887
ABC Corp.	18946	Jill	Devia	26	Canal St.	Fairfax	VA	22087
XYZ Corp.	19847	Bill	Bosco	11	Lake Dr.	Richmond	VA	23876
XYZ Corp.	83746	Bill	Dashell	45	Forest St.	Baltimore	MD	24533
XYZ Corp.	19374	David	Johns	581	Lugar Dr.	Rockville	MD	20845
XYZ Corp.	19328	Jim	Hatch	2374	Whitman Dr.	Fairfax	VA	22087
XYZ Corp.	19374	Tina	Budge	198	Wallis St.	Bethesda	MD	20984

関係代数のオペレータ (1 / 2)

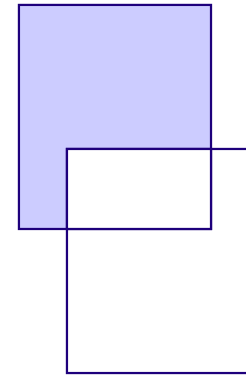
Union



Intersection



Difference



Cartesian Product

a1	b1
a2	b2

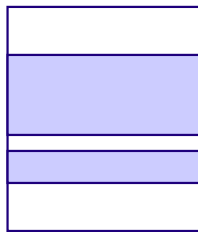
c1	d1
c2	d2



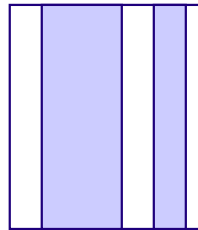
a1	b1	c1	d1
a2	b2	c1	d1
a1	b1	c2	d2
a2	v2	c2	d2

関係代数のオペレータ (2 / 2)

Selection



Projection



Natural (Join)

a1	b1
a2	b2

b1	e1
b2	e2



a1	b1	e1
a2	b2	e2

Divide

a
b
c

a	x
a	y
a	z
b	x

x
z



x

Relation Calculus

- データベース問い合わせは、Relation Calculus を使っても書ける

例 $\{ t \mid (t \in \text{bank}) \ \& \ (t.\text{LNAME} = \text{Smith}) \}$
テーブル bank から、LNAME が Smith である
ようなタプルをすべて求めよ

$$\{ t1 \mid (\exists t2) ((t1 \in \text{bank}) \ \& \ (t2 \in \text{emp}) \\ \& \ (t1.\text{FNAME} = t2.\text{FNAME}) \\ \& \ (t1.\text{LNAME} = t2.\text{LNAME}) \\ \& \ (t2.\text{CITY} = \text{Fairfax})) \}$$

SQLと関係データベース

- SQLは、関係データベースのための、標準的なデータベース言語
 - 検索： 欲しいデータを取り出す機能
 - 操作： データの更新, 挿入, 削除の機能

SQL

典型的なSQL文の例

SELECT attr1, attr2, ... , attrn

FROM R1<V1>, R2<V2>, ... , Rk<Vk>

<WHERE F>

R1, R2, ..., Rk: テーブル名

V1, V2, ... , Vk : タプル変数 (省略してもよい)

WHERE句は、省略してもよい

SQLでの更新, 挿入, 削除

テーブル 社員 (氏名, 自宅番号) に対して

- 挿入

```
INSERT INTO 社員 (氏名, 自宅番号)
```

```
VALUES ("金子邦彦", "092-642-4XXX")
```

- 更新

```
UPDATE 社員
```

```
SET 自宅番号 = "092-643-0XXX"
```

```
WHERE 氏名 = "金子邦彦"
```

- 削除

```
DELETE FROM 社員
```

```
WHERE 氏名 = "金子邦彦"
```

SQL と Relational Calculus

```
SELECT attr1, attr2, ... , attrn  
FROM R1<V1>, R2<V2>, ... , Rk<Vk>  
<WHERE F>
```

Relational Calculusで書くと

$$\{ t \mid (V_1 \in R_1) \& \dots \& (V_k \in R_k) \& F \}$$

t のスキーマは、 (attr1, attr2, ... , attrn)

関係代数で書くと

$$\pi_{attr1, \dots, attrn} (\sigma_G (R_1 \times \dots \times R_k))$$

SQLの例 (1/2)

```
SELECT COMP,CITY  
FROM emp
```

```
SELECT TRANS,DAY,MTH,YR  
FROM emp  
WHERE FNAME=Jone & LNAME =  
Smith  
  
& AMOUNT > 6000
```

SQLの例 (2/2)

```
SELECT  FNANE, LNAME  
FROM    emp.E, birth B  
WHERE   E.SSN = B.SSN
```