Division Usage: Review Example

• Query: "List the usernames of people who have in common all Susan's friends on Facebook."

Relation Friends

$$SF \leftarrow \sigma_{Username = `Susan'}(Friends)$$

$$SSF \leftarrow Friends \div \pi_{FN,LN}(SF)$$

 $RSLT \leftarrow \sigma_{Username \neq `Susan'}(SSF)$

Username	FNFriend	LNFriend
Susan	Alex	L
Mark	Alex	L
Kirk	Mary	K
Shi	Alex	L
Susan	Mary	K
Shi	Lory	М
Kirk	Chia	S
Kirk	Alex	L

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Extended Relational Operations

- □ Extended set and Relational operations:
 - Outer Union
 - Outer Joins
- Aggregate operations:
 - MAX, MIN, AVG, SUM
 - Count
 - Subset: grouping
- Arithmetic operations and other functions:

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Outer Union

- it is defined on partially union compatible relations
 - Non union-compatible attributes are kept in r ∪* s
 - Non union-compatible attributes without value are set
 - Tuples are "matched" over common named attributes like in natural join $r \cup * s$

relation r

relation **s**

FN	LN	MJ
a	b	cs
d	а	ce
С	b	cs

f q а

FN	LN	MJ	CL
а	b	CS	Null
d	а	ce	sr
С	b	CS	Null
b	g	Null	f

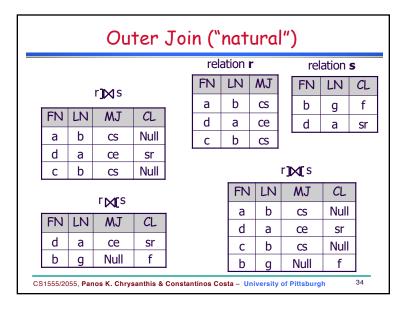
what about outer intersection or outer difference?

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Outer Join

- □ Join selects only tuples satisfying the join-condition
- □ Outer Join:
 - Left outer join (r 🔀 s) keeps every tuple in the left
 - Right outer join (r 🛛 s) keeps every tuple in the right relation
 - Full outer join (r x s) keeps every tuple
- □ Attributes of tuples with no matching tuples are set to **NULL**
- □ With out a join-condition they behave like natural join

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh



Aggregate Functions

- Mathematical and Statistical aggregate functions on collections of values
 - SUM, MAXIMUM, MINIMUM, AVERAGE
 - COUNT number of tuples (cardinality)

f <function list> (< relation>)

- Function list is a list of pairs (< function, attribute >)
- E.g., \$\(\operage\) count SID, AVERAGE GPA (STUDENT)

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

0.5

Aggregate Functions: Example

□ RSLT \leftarrow \$\frac{1}{2}\$ count SID, AVERAGE GPA (STUDENT)

Student

SID	Name	Age	GPA
546007	Susan	18	3.8
546100	Bob	19	3.65
546500	Bill	20	3.7

RSLT

Count_SID	AVERAGE_GPA
3	3.72

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Example of Aggregation Query

- Q: Find the students with the highest GPA.
- □ Student (SID, Name, Age, GPA)
- □ A:

 $MG(MGPA) \leftarrow f_{MAX GPA}$ (Student);

 $RSLT \leftarrow MG \bowtie MGPA = GPA (Student);$

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

37

Grouping Grouping the tuple in a relation Grouping attributes \$ < function list> (< relation>) Tuples are grouped based on the values of grouping attributes E.g., Q1: major \$ count SID, AVERAGE GPA (STUDENT) major Count_SID AVERAGE_GPA Q2: class, major \$ count SID, AVERAGE GPA (STUDENT) ? class major Count_SID AVERAGE_GPA CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Recursive closure

- □ It is applied to a recursive relationship between tuples of the same relation
- E.g., find all the ancestors or descendants
- □ How do we express it?
- What about the join operation?
- Need control statements...iteration

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa - University of Pittsburgh

Write Queries in Relational Algebra

- Deletion:
 - r ← r Relational_Expression
 - STUDENT ← STUDENT − (σ Dept = 'CSD' ∧ QPA<2.5 (STUDENT))
- □ Insertion:
 - $r \leftarrow r \cup Relational Expression$
 - STUDENT ← STUDENT ∪ {(365, `Smith', `John')}
- Updating:
 - $r \leftarrow \Pi$ attributes-to-be-updated (r)
 - STUDENT $\leftarrow \Pi$ Dept = 'CSD' (σ Dept = 'CS' (STUDENT))

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa – University of Pittsburgh

Discussion

- □ The relational algebra is *procedural*
- The queries in relational algebra specify *how* to produce a result, BUT...
- ☐ The *how* should be the responsibility of the system
- User queries should be *declarative* specifying what is to be retrieved
 - Textual query languages (SQL, QUEL)
 - Graphical query languages (QBE)
 - Visual iconic languages (QBI)
- Other formal query languages:
 - Relational tuple calculus
 - Relational domain calculus

CS1555/2055, Panos K. Chrysanthis & Constantinos Costa – University of Pittsburgh

41

