

العمليات الإضافية وحساب القضايا

د. مادلين عبود

محتوى مجاني غير مخصص للبيع التجاري



04/05/2023

10

قواعد معطيات 1

RB Informatics;

تحدثنا في المحاضرة السابقة عن البنى الأساسية في النموذج العلاقتي وعن العمليات الأساسية في الجبر العلاقتي...
أما في هذه المحاضرة سنكمل الحديث عن العمليات الإضافية لندخل في حساب القضايا.

1. العمليات الإضافية:

■ عمليات مشتقة من العمليات الأساسية وبالتالي لا تضيف أي قوة للجبر العلاقتي.
■ تستخدم لتبسيط الكتابة.

■ العمليات المشتقة: » التقاطع set intersection

» الدمج الطبيعي Natural join

» القسمة Division

» الإسناد Assignment



التقاطع Set Intersection Operation

■ تجرى هذه العملية بين علاقيتين ويجب أن تحقق هاتان العلاقتان الشروط المذكورة في عملية الاجتماع.
■ نتيجة عملية التقاطع هي علاقة تحوي مجموعة الحدوديات الموجودة ضمن العلاقتين.

Notation: $r \cap s$ Defined as: $r \cap s = \{t | t \in r \text{ and } t \in s\}$ $r \cap s = r - (r - s)$.Note: عدد الوصفات في (r, s) هو نفسهمجال الوصفة رقم i في r هو نفسه مجال الوصفةرقم i في s وتكون هذه الوصفة متجانسة

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

Relations r, s :



A	B
α	2

: $r \cap s$

الدمج الطبيعي Natural-join Operation

غالبا ما نرغب في تبسيط بعض الاستعلامات التي تحتاج إلى جداء ديكارتي ومعظم عمليات الاستعلام التي تحوي جداء ديكارتيًا تحوي عملية اختيار من نتيجة الجداء.

فعملية الدمج هي عملية ثنائية تسمح بتركيب عملية الاختيار والجداء الديكارتي بعملية واحدة

لنأخذ كتعريف لهذه العملية العلاقتين $r(R)$ و $s(S)$ ونقول أنّ دمج العلاقتين هي علاقة مخططها هو اجتماع مخططي العلاقتين ومعرف بالشكل:

$$r \bowtie s = \Pi_{RIS} \left(\sigma_{r.A_1=s.A_1 \wedge r.A_2=s.A_2 \wedge \dots r.A_n=s.A_n} (r \times s) \right)$$

حيث: $RIS = \{A_1, A_2, A_3, \dots, A_n\}$

Notation: $r \bowtie s$

لتكن لدينا r و s علاقات على المخططات R و S بشكل متتالي تكون النتيجة هي علاقة على المخطط الذي

تم الحصول عليه من خلال t_s من s و t_r من r

إذا كان t_r و t_s نفس القيمة لكل واصفة في $R \cap S$ إذاً هناك t tuple ستضاف إلى النتيجة بحيث:

t has the same value as t_r on r

t has the same value as t_s on s

Example1:

R=(A,B,C,D)

S=(E,B,D)

Result schema=(A,B,C,D,E)

 $r \bowtie s$ is Defined as:

$$\Pi_{r.A, r.B, r.D, r.C, s.E} \left(\sigma_{r.B=s.B \cap r.D=s.D} (r \times s) \right).$$



Example2:

لإيجاد أسماء جميع الأفرع التي لزيائنها حساب في المصرف وتعيش في مدينة Harrison نكتب:

$$\Pi_{branch-name} (\sigma_{customer-city="harrison"} (customer \bowtie account \bowtie depositor)).$$

حالات خاصة:

- إذا كان $RIS = \phi \Rightarrow r \bowtie s = r \times s$ وكانت: \ominus هناك قضية على الواصفات في مخطط العلاقة الناتجة عن الدمج نكتب: $r \bowtie \ominus \leftarrow \sigma_{\ominus}(r \times s)$
- :Relations r, s

A	B	C	D
α	1	α	a
β	2	γ	a
γ	4	β	b
α	1	γ	a
δ	2	β	b

r

B	D	E
1	a	α
3	a	β
1	a	γ
2	b	δ
3	b	ϵ

s

$r \bowtie s$



A	B	C	D	E
α	1	α	a	α
α	1	α	a	γ
α	1	γ	a	α
α	1	γ	a	γ
δ	2	β	b	δ

عملية القسمة Division operation

- هذه العملية مناسبة للاستعلامات التي تحوي كلمة "لأجل كل" فإيجاد مجموعة الزبائن الذين لديهم حسابات مصرفية في جميع الأفرع الموجودة في مدينة "brooklyn" نقوم بما يلي:

» نستخرج مجموعة الفروع الموجودة في مدينة "brooklyn" بكتابة التعبير التالي:

$$r_1 = \Pi_{branch-name} (\sigma_{branch-city="brooklyn"}(branch))$$

» ثم نستخرج مجموعة الزبائن والفروع الذين لديهم حسابات فيها بكتابة التعبير:

$$r_2 = \Pi_{customer-name, branch-name}(depositor \bowtie account)$$

» ونحصل على النتيجة المطلوبة بكتابة:

$$r_2 \div r_1$$

:Relations r, s

A	B	C	D	E
α	a	α	a	1
α	a	γ	a	1
α	a	γ	b	1
β	a	γ	a	1
β	a	γ	b	3
γ	a	γ	a	1
γ	a	γ	b	1
γ	a	β	b	1

r

D	E
a	1
b	1

s

A	B	C
α	a	γ
γ	a	γ

: $r \div s$

A	B
α	1
α	2
α	3
β	1
γ	1
δ	1
δ	3
δ	4
ϵ	6
ϵ	1
β	2

r

B
1
2

s

:Relation r, s

A
α
β

: $r \div s$

لتكن لدينا r و s علاقات على المخططات R و S بشكل متتالي بحيث:

- ❖ $R = (A_1, \dots, A_m, B_1, \dots, B_n)$
- ❖ $S = (B_1, \dots, B_n)$

❖ إن نتيجة $r \div s$ هي علاقة على المخطط:

$$R - S = (A_1, \dots, A_m)$$

$$r \div s = \{t | t \in \Pi_{R-S}(r) \wedge \forall u \in s (t_u \in r)\}$$



■ $q \times s \subseteq r$ حيث $q = r \div s$ هي أكبر علاقة تحقق

■ Let $r(R)$ and $s(S)$ be relation and let $S \subseteq R$

■ $r \div s = \Pi_{R-S}(r) - \Pi_{R-S}((\Pi_{R-S}(r) \times s) - \Pi_{R-S,S}(r))$

■ $\Pi_{R-S,S}(r)$: ترتب واصفات r

■ $:\Pi_{R-S}((\Pi_{R-S}(r) \times s) - \Pi_{R-S,S}(r))$

gives those tuples t in $\Pi_{R-S}(r)$ such that for some tuple $u \in s, t_u \notin r$

النسب Assignment Operation

■ وتسمى بالإسناد ويرمز للعملية بـ " \leftarrow "

■ تعمل هذه العملية بكيفية مشابهة لعملية الإسناد في لغات البرمجة وتعتبر طريقة مناسبة للتعبير عن استعلامات معتمدة.

■ تجري دوما هذه العملية لمتغير علاقة مؤقتة

Example: write $r \div s$ as:

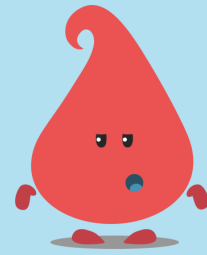
$$\text{Temp1} \leftarrow \Pi_{R-S}(r)$$

$$\text{Temp2} \leftarrow \Pi_{R-S}((\text{temp1} \times s) - \Pi_{R-S,S}(r))$$

$$\text{Result} = \text{temp1} - \text{temp2}.$$

Relation variable \leftarrow the result

■ May use variable in subsequent expressions.



أمثلة عن استفسارات:

Find all customers who have an account from at least the "Downtown" and "uptown" branches.

❖ Query1

$$\Pi_{CN}(\sigma_{BN="Downtown"}(\text{depositor} \bowtie \text{account})) \cap \Pi_{CN}(\sigma_{BN="uptown"}(\text{depositor} \bowtie \text{account}))$$

CN=customer-name

BN=branch-name

❖ Query2

$$\Pi_{CN, BN}(\text{depositor} \bowtie \text{account}) \div P_{\text{temp}(BN)}(\{("Downtown"), ("Uptown")\})$$

Find all customers who have an account at all branches located in Brooklyn city.

$$\Pi_{CN,BN}(\text{depositor} \bowtie \text{account}) \div \Pi_{BN}(\sigma_{\text{branch-city}="Brooklyn"}(\text{branch}))$$

2. حساب القضايا predicate calculus

لغة القضايا: هي لغة استعلام غير إجرائية تسمح بوصف المعلومات المرغوب الحصول عليها دون تبيان الإجراء اللازم لذلك.

الاستعلام بهذه اللغة له الشكل التالي: $\{t \mid \rho(t)\}$ الذي يعني:

"مطلوب الحصول على جميع الحدوديات t التي تحقق القضية ρ "

المتحول الحدودي هو متحول حر ما عدا المتحول المسبوق بمعامل "يوجد \exists " أو "مهما يكن \forall "

يستخدم في هذه اللغة الرموز التالية:

$t[A]$: للدلالة على قيمة الحدودية t في الواصفة A .

$t \in r$: للدلالة على أن الحدودية t هي في العلاقة r (متحول حدودي).

والقضية ρ مبنية على تركيب من عناصر لها أحد الأشكال التالية:

» $s \in r$ حيث s متحول حدودي و r علاقة.

» $S[x] \circ u[y]$ حيث S, u متحويلات حدودية، x واصفة في S و y واصفة في u و \circ عملية مقارنة

ويجب أن يكون مجالا الواصفين x و y متساويين.

» $s[x] \circ c$ حيث c ثابت في مجال تعريف الواصفة x .

ويجري بناء القضايا باستخدام القواعد التالية:

» أي عنصر معرف بأحد الأشكال السابقة هو قضية.

» P قضية $\Leftarrow (\rho)$ و $(\neg p)$ هما قضيتان.

» p_1 و p_2 قضيتان $\Leftarrow p_1 \wedge p_2, p_1 \vee p_2, p_1 \rightarrow p_2$ جميعها قضايا.

» إذا كان $p_1(s)$ قضية حيث s متحول حر و r علاقة فإن:

$\exists s \in r(p_1(s))$, $\forall s \in r(p_1(s))$ قضايا.



نعود لمثال قاعدة معطيات المصرف:

- Branch (branch_name, branch_city, assets).
- Customer (customer_name, customer_street, customer_only).
- Account (account_number, branch_name, balance).
- Loan (loan_number, branch_name, amount).
- Depositor (customer_name, account_number).
- Borrower (customer_name, loan_number).

Find the loan-number, branch-name and amount for loans of over 1200\$

✓ $\{t \mid t \in \text{loan} \wedge t[\text{amount}] > 1200\}$



- Find the loan-number for each loan of an amount greater than 1200\$
 $\checkmark \{t | \exists s \in \text{loan}(t[\text{loan} - \text{number}] = s[\text{loan} - \text{number}] \wedge s[\text{amount}] > 1200)\}$
 Notices that a relation on schema [customer-name] is implicitly defined by the query
- Find the names for all customers having a loan, an account, or both at the bank.
 $\checkmark \{t | \exists s \in \text{borrower}(t[\text{customer} - \text{name}] = s[\text{customer} - \text{name}]) \vee \exists u \in \text{depositor}(t[\text{customer} - \text{name}] = u[\text{customer} - \text{name}])\}$
- Find the names for all customers who have a loan and an account at the bank.
 $\checkmark \{t | \exists s \in \text{borrower}(t[\text{customer} - \text{name}] = s[\text{customer} - \text{name}]) \wedge \exists u \in \text{depositor}(t[\text{customer} - \text{name}] = u[\text{customer} - \text{name}])\}$
- Find the names of all customers having a loan at the *perryridge* branch.
 $\checkmark \{t | \exists s \in \text{borrower}(t[\text{customer} - \text{name}] = s[\text{customer} - \text{name}] \wedge \exists u \in \text{loan}(u[\text{branch} - \text{name}] = \text{"perryridge"} \wedge u[\text{loan} - \text{number}] = s[\text{loan} - \text{number}]))\}$
- Find the names for all customers who have a loan at the *perryridge* branch but no account at any branch of the bank.
 $\checkmark \{t | \exists s \in \text{borrower}(t[\text{customer} - \text{name}] = s[\text{customer} - \text{name}] \wedge \exists u \in \text{loan}(u[\text{branch} - \text{name}] = \text{"perryridge"} \wedge u[\text{loan} - \text{number}] = s[\text{loan} - \text{number}])) \wedge \text{not } \exists v \in \text{depositor}(v[\text{customer} - \text{name}] = t[\text{customer} - \text{name}])\}$
- Find the names of all customers having a loan from the *perryridge* branch and the cities they live in.
 $\checkmark \{t | \exists s \in \text{loan}(s[\text{branch} - \text{name}] = \text{"perryridge"} \wedge \exists u \in \text{borrower}(u[\text{loan} - \text{number}] = s[\text{loan} - \text{number}] \wedge t[\text{customer} - \text{name}] = u[\text{customer} - \text{name}] \wedge \exists v \in \text{customer}(u[\text{customer} - \text{name}] = v[\text{customer} - \text{name}] \wedge t[\text{customer} - \text{city}] = v[\text{customer} - \text{city}]))))\}$
- Find the names for all customers who have an account at all branches located in Brooklyn:
 $\checkmark \{t | \exists c \in \text{customer}(t[\text{customer} - \text{name}] = c[\text{customer} - \text{name}]) \wedge \forall s \in \text{branch}(s[\text{branch} - \text{city}] = \text{"Brooklyn"} \Rightarrow \exists u \in \text{account}(s[\text{branch} - \text{name}] = u[\text{branch} - \text{name}] \wedge \exists v \in \text{depositor}(t[\text{customer} - \text{name}] = v[\text{customer} - \text{name}] \wedge v[\text{account} - \text{number}] = u[\text{account} - \text{number}]))))\}$

نتائج بعض الاستفسارات "للمحاضرة السابقة والمحاضرة الحالية"

Result of $(\sigma_{branch-name="Perryridge"}(loan))$

Loan number	Branch-name	amount
L-15	perryridge	1500
L-16	perryridge	1300



Loan number and the amount of the loan

Loan number	amount
L-11	900
L-14	1500
L-15	1500
L-16	1300
L-17	1000
L-23	2000
L-93	500

Names of all customers who have either a loan or an account

Customer-name

Addams
Curry
Hayes
Jackson
Jones
Smith
Williams
Lindsay
Johnson
Turner

Customers with both an account and a loan at the bank

Customer-Name

Hayes
Jones
Smith

Customer with an account but no loan:

Customer-name

Johnson
Lindsay
Turner

Result of $\Pi_{customer-name}$

Customer-name

adams
hayes

Largest Account Balance in the Bank

Balance

900

Result of the subexpression

Balance
500
400
700
750
350

Customers Who live on the same street and in the same city as smith

Customer-name
Curry
Smith

Result of $\Pi_{branch-name} \left(\sigma_{customer-city="harrison"}(customer \bowtie account \bowtie depositor) \right)$

Branch-name
Brighton
Perryridge

Result of $\Pi_{branch-name} \left(\sigma_{branch-city="Brooklyn"}(branch) \right)$

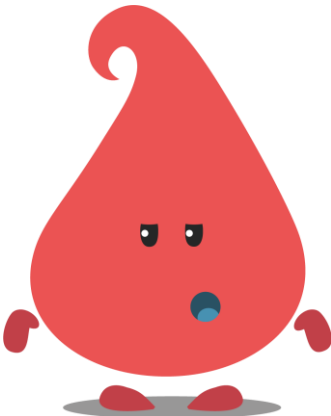
Branch-name
Brighton
Downtown

Result of $\Pi_{(customer-name,branch-name)}(depositor \bowtie account)$

Customer-name	Branch-name
Hayes	Perryridge
Johnson	Downtown
Johnson	Brighton
Jones	Brighton
Lindsay	Redwood
Smith	Mianus
Turner	Round hill

Result of $borrower \times loan$

<i>customer-name</i>	<i>borrower. loan-number</i>	<i>loan. loan-number</i>	<i>branch-name</i>	<i>amount</i>
Adams	L-16	L-11	Round Hill	900
Adams	L-16	L-14	Downtown	1500
Adams	L-16	L-15	Perryridge	1500
Adams	L-16	L-16	Perryridge	1300
Adams	L-16	L-17	Downtown	1000
Adams	L-16	L-23	Redwood	2000
Adams	L-16	L-93	Mianus	500
Curry	L-93	L-11	Round Hill	900
Curry	L-93	L-14	Downtown	1500
Curry	L-93	L-15	Perryridge	1500
Curry	L-93	L-16	Perryridge	1300
Curry	L-93	L-17	Downtown	1000
Curry	L-93	L-23	Redwood	2000
Curry	L-93	L-93	Mianus	500
Hayes	L-15	L-11		900
Hayes	L-15	L-14		1500
Hayes	L-15	L-15		1500
Hayes	L-15	L-16		1300
Hayes	L-15	L-17		1000
Hayes	L-15	L-23		2000
Hayes	L-15	L-93		500
...
...



Result of $\sigma_{branch-name="perryridge"}(borrower \times loan)$

<i>customer-name</i>	<i>borrower. loan-number</i>	<i>loan. loan-number</i>	<i>branch-name</i>	<i>amount</i>
Adams	L-16	L-15	Perryridge	1500
Adams	L-16	L-16	Perryridge	1300
Curry	L-93	L-15	Perryridge	1500
Curry	L-93	L-16	Perryridge	1300
Hayes	L-15	L-15	Perryridge	1500
Hayes	L-15	L-16	Perryridge	1300
Jackson	L-14	L-15	Perryridge	1500
Jackson	L-14	L-16	Perryridge	1300
Jones	L-17	L-15	Perryridge	1500
Jones	L-17	L-16	Perryridge	1300
Smith	L-11	L-15	Perryridge	1500
Smith	L-11	L-16	Perryridge	1300
Smith	L-23	L-15	Perryridge	1500
Smith	L-23	L-16	Perryridge	1300
Williams	L-17	L-15	Perryridge	1500
Williams	L-17	L-16	Perryridge	1300

Result of $\Pi_{customer-name, loan-number, amount}(borrower \bowtie loan)$

customer-name	loan-number	amount
Adams	L-16	1300
Curry	L-93	500
Hayes	L-15	1500
Jackson	L-14	1500
Jones	L-17	1000
Smith	L-23	2000
Smith	L-11	900
Williams	L-17	1000

ملاحظة هامة:

في الجبر العلاقتي دوماً نحذف التكرار من أي عملية نودّ القيام بها على عكس ما سنتعلمه في قسم الـ SQL



THE END