General Overview

- relational model
- Formal query languages
  - relational algebra
  - rel. tuple calculus
  - rel. domain calculus

Overview - detailed

- rel. tuple calculus
  - dfn
  - details
  - equivalence to rel. algebra
- rel. domain calculus + QBE

Rel. domain calculus (RDC)

- Q: why?
- A: slightly easier than RTC, although equivalent - basis for QBE.
- ideal domain variables (w/ F.O.L.) - eg:
  - "find STUDENT record with ssn=123"

Rel. Dom. Calculus

- find STUDENT record with ssn=123'

\{ s, n, a \mid s, n, a \in \text{STUDENT} \land s = 123 \}

Details

- Like R.T.C - symbols allowed:
  \( \land, \lor, \rightarrow \Rightarrow \geq, \leq, = \)
  \( (, ), \in \)
- quantifiers
Details

- but: domain (= column) variables, as opposed to tuple variables, eg:

\[ <s, n, a> \in STUDENT \]

- \text{ssn} \quad \text{name} \quad \text{address}

Reminder: our Mini-U db

\begin{array}{|c|c|c|}
\hline
\text{SSN} & \text{Name} & \text{Address} \\
\hline
123 & smith & main st \\
234 & jones & forbes ave \\
\hline
\end{array}

\begin{array}{|c|c|c|c|}
\hline
\text{c-id} & \text{c-name} & \text{units} \\
\hline
15-413 & s.e. & 2 \\
15-412 & o.s. & 2 \\
\hline
\end{array}

\begin{array}{|c|c|c|}
\hline
\text{SSN} & \text{c-id} & \text{grade} \\
\hline
123 & 15-413 & A \\
234 & 15-413 & B \\
\hline
\end{array}

Examples

- find all student records

\[ \{ <s, n, a> \in STUDENT \} \]

\text{RtC:} \quad \{ t \in \text{STUDENT} \}

Examples

- (selection) find student record with ssn=123

\[ \{ 123, n, a \in STUDENT \} \]

Examples

- (projection) find name of student with ssn=123

\[ <n > \quad 123, n, a \in STUDENT \}

\text{RtC:} \quad \{ t \in \text{STUDENT} \wedge t_{\text{ssn}} = 123 \}
Examples
- (projection) find name of student with ssn=123
  \(<n>\exists a(<123,n,a \in STUDENT>)\}
  \uparrow \text{need to restrict “s”}
  RTC: \{t | \exists s \in STUDENT(s[ssn]=123 \land t[name]=s[name])\}

Examples cont’d
- (union) get records of both PT and FT students
  RTC: \{t | t \in FT \_STUDENT \lor t \in PT \_STUDENT\}

Examples
- difference: find students that are not staff
  RTC: \{t | t \in STUDENT \land t \notin STAFF\}

Examples cont’d
- (union) get records of both PT and FT students
  \(<s,n,a> \land s \in FT \_STUDENT \lor s \in PT \_STUDENT\}

Examples
- difference: find students that are not staff
  \(<s,n,a> \land s \in STUDENT \land s \notin STAFF\}
Cartesian product

- eg., dog-breeding: MALE x FEMALE
- gives all possible couples

<table>
<thead>
<tr>
<th>MALE</th>
<th>FEMALE</th>
</tr>
</thead>
<tbody>
<tr>
<td>spike</td>
<td>lassie</td>
</tr>
<tr>
<td>spot</td>
<td>shiba</td>
</tr>
</tbody>
</table>

\[
\text{MALE} \times \text{FEMALE} = \{m \times f | m \in \text{MALE}, f \in \text{FEMALE}\}
\]

Cartesian product

- find all the pairs of (male, female) - RTC:

\[
\{t | \exists m \in \text{MALE} \wedge \\
\exists f \in \text{FEMALE} \\
\quad t[m \rightarrow \text{name} = m[\text{name}] \wedge \\
\quad t[f \rightarrow \text{name} = f[\text{name}]\}
\]

Cartesian product

- find all the pairs of (male, female) - RDC:

\[
\{<m, f> | m \in \text{MALE} \wedge \\
\quad f \in \text{FEMALE}\}
\]

‘Proof’ of equivalence

- rel. algebra <-> rel. domain calculus
- rel. tuple calculus

Overview - detailed

- rel. domain calculus
  - why?
  - details
  - examples
  - equivalence with rel. algebra
  - more examples: ‘safety’ of expressions

More examples

- join: find names of students taking 15-415
Reminder: our Mini-U db

<table>
<thead>
<tr>
<th>STUDENT</th>
<th>CLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSN</td>
<td>SSN</td>
</tr>
<tr>
<td>Name</td>
<td>id</td>
</tr>
<tr>
<td>Address</td>
<td>c-name</td>
</tr>
<tr>
<td>123</td>
<td>15-113</td>
</tr>
<tr>
<td>smith</td>
<td>15-113</td>
</tr>
<tr>
<td>main st</td>
<td>i.s.a.</td>
</tr>
<tr>
<td>234</td>
<td>15-12</td>
</tr>
<tr>
<td>jones</td>
<td>o.s.a.</td>
</tr>
<tr>
<td>forbes</td>
<td></td>
</tr>
<tr>
<td>ave</td>
<td></td>
</tr>
</tbody>
</table>

TAKES

<table>
<thead>
<tr>
<th>SSN</th>
<th>id</th>
<th>grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>15-113</td>
<td>A</td>
</tr>
<tr>
<td>234</td>
<td>15-12</td>
<td>B</td>
</tr>
</tbody>
</table>

More examples

- join: find names of students taking 15-415 in RTC
  \[t \exists s \in STUDENT \land \exists e \in TAKES (s[ssn]=e[ssn] \land t[name]=s[name] \land e[c-ld]=15-415)\]

More examples

- join: find names of students taking 15-415 in RDC
  \[\langle n \rangle \exists s \exists t (s[n]=t[n] \land s[ssn] \in STUDENT \land t[ssn]=TAKES)\]

Sneak preview of QBE:

- very user friendly
- heavily based on RDC
- very similar to MS Access interface

Sneak preview of QBE:

- 3-way join: find names of students taking a 2-unit course in RTC:
  \[t \exists e \in STUDENT \land \exists t \in TAKES \exists c \in CLASS (s[ssn]=e[ssn] \land e[c-ld]=c[c-ld] \land t[name]=s[name] \land c[units]=2)\]

More examples

- join: find names of students taking 15-415 in RTC
  \[t \exists s \in STUDENT \land \exists e \in TAKES (s[ssn]=e[ssn] \land t[name]=s[name] \land e[c-ld]=15-415)\]
More examples

- 3-way join: find names of students taking a 2-unit course

\[
\{<n> | \exists a, c, g, cn \in \text{STUDENT} \land \exists s, n, a \in \text{TAKES} \land \exists c, n, 2 \in \text{CLASS}\}
\]

Even more examples:

- self-joins: find Tom’s grandparent(s)

\[
\{t | \exists p \in \text{PC} \land \exists q \in \text{PC} \\
( p[c-id] = q[p-id] \land \exists \text{Tom} = q[p-id] \land q[c-id] = "Tom") \}
\]

Even more examples:

- self-joins: find Tom’s grandparent(s)

\[
\{t | \exists p \in \text{PC} \land \exists q \in \text{PC} \\
( p[c-id] = q[p-id] \land \exists \text{Tom} = q[p-id] \land q[c-id] = "Tom") \}
\]
Even more examples:

- self-joins: find Tom’s grandparent(s)

\[
\langle \text{g} \rangle \exists \text{p}(\langle \text{g}, \text{p} \rangle \in \text{PC} \land \langle \text{p}, \text{"Tom"} \rangle \in \text{PC})
\]

Hard examples: DIVISION

- find suppliers that shipped all the ABOMB parts

\[
\begin{align*}
\text{SHIPMENT} & \quad \text{ABOMB} \\
\text{BAD} & \quad \text{BAD} \\
s_1 & \quad p_1 \\
s_2 & \quad p_1 \\
s_1 & \quad p_2 \\
s_3 & \quad p_1 \\
s_5 & \quad p_3 \\
\end{align*}
\]

Hard examples: DIVISION

- find suppliers that shipped all the ABOMB parts

\[
\begin{align*}
\{ t \forall p(p \in \text{ABOMB} \Rightarrow ( \\
\exists s \in \text{SHIPMENT}( \\
\quad d,s\# = s[s\#] \land \\
\quad s[p\#] = p[p\#])))
\}
\end{align*}
\]

Hard examples: DIVISION

- find suppliers that shipped all the ABOMB parts

\[
\begin{align*}
\{ t \forall p(p \in \text{ABOMB} \Rightarrow ( \\
\exists s \in \text{SHIPMENT}( \\
\quad d,s\# = s[s\#] \land \\
\quad s[p\#] = p[p\#]))
\}
\end{align*}
\]

More on division

- find students that take all the courses that ssn=123 does (and may be even more)

\[
\{ o \forall t((t \in \text{TAKES} \land t[\text{ssn}] = 123) \Rightarrow \\
\exists r \in \text{TAKES}( \\
\quad t[c \cdot \text{id}] = r[c \cdot \text{id}] \land \\
\quad t[\text{ssn}] = o[\text{ssn}])))
\}
\]

More on division

- find students that take all the courses that ssn=123 does (and may be even more)

\[
\{ s \forall c(\exists g(123, c, g \in \text{TAKES}) \Rightarrow \\
\exists g'(s, c, g') \in \text{TAKES})))
\}
\]
Safety of expressions

- similar to RTC
- FORBIDDEN:

\[ \langle s, n, a \rangle \not\subset s, n, a \subseteq \text{STUDENT} \]

Overview - detailed

- rel. domain calculus + QBE
  - def
  - details
  - equivalence to rel. algebra

\[ \{ (x, y) \mid x \in \text{STUDENT} \land (y \in \text{TAKES}) \} \]

Table

<table>
<thead>
<tr>
<th>Hi</th>
<th>There</th>
<th>For</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>And</td>
<td>For</td>
</tr>
<tr>
<td>Hat</td>
<td>This</td>
<td>is</td>
</tr>
</tbody>
</table>