# QBE: A Query Language Based on Domain Calculus (Appendix C)

- This language is based on the idea of giving an example of a query using “example elements” which are nothing but domain variables.
- Notation: An example element stands for a domain variable and is specified as an example value preceded by the underscore character.
- P. (called P dot) operator (for “print”) is placed in those columns which are requested for the result of the query.
- A user may initially start giving actual values as examples, but later can get used to providing a minimum number of variables as example elements.

# QBE: A Query Language Based on Domain Calculus (Appendix C)

- The language is very user-friendly, because it uses minimal syntax.
- QBE was fully developed further with facilities for grouping, aggregation, updating etc. and is shown to be equivalent to SQL.
- The language is available under QMF (Query Management Facility) of DB2 of IBM and has been used in various ways by other products like ACCESS of Microsoft, and PARADOX.
- For details, see Appendix C in the text.

# QBE Examples

- QBE initially presents a relational schema as a “blank schema” in which the user fills in the query as an example:

# Example Schema as a QBE Query Interface

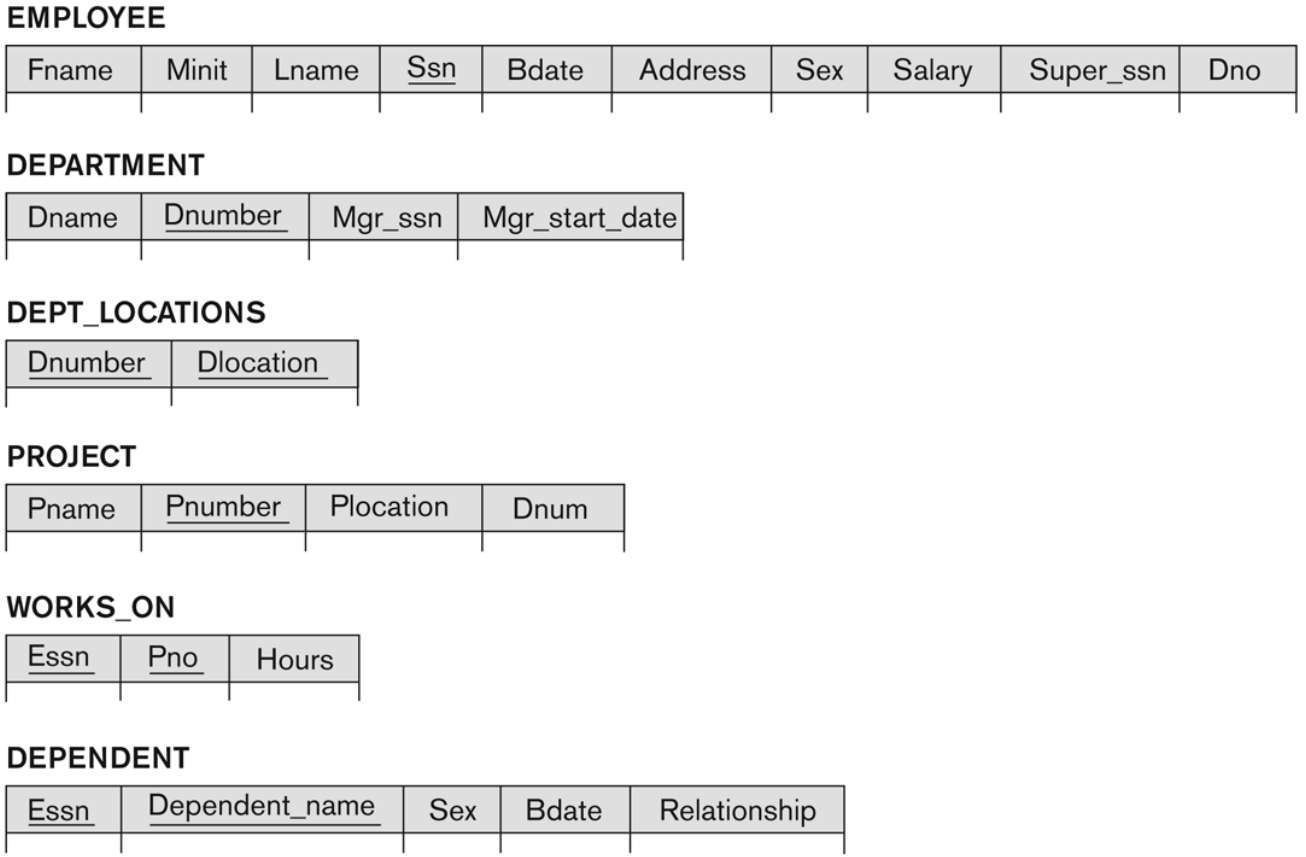


**Figure C.1**
The relational schema of Figure 5.5 as it may be displayed by QBE.

# QBE Examples

- The following domain calculus query can be successively minimized by the user as shown:
- Query :

**{uv | (∃ q) (∃ r) (∃ s) (∃ t) (∃ w) (∃ x) (∃ y) (∃ z)**
**(EMPLOYEE(qrstuvwxyz) and q='John' and r='B' and s='Smith')}**

# Four Successive Ways to Specify a QBE Query

**(a)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| John | B | Smith | _123456789 | P._9/1/60 | P._100 Main, Houston, TX | _M | _25000 | _123456789 | _3 |

**(b)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| John | B | Smith | | P._9/1/60 | P._100 Main, Houston, TX | | | | |

**(c)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| John | B | Smith | | P._X | P._Y | | | | |

**(d)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| John | B | Smith | | P. | P. | | | | |

**Figure C.2**
Four ways to specify the query Q0 in QBE.

# QBE Examples

- Specifying complex conditions in QBE:
- A technique called the “condition box” is used in QBE to state more involved Boolean expressions as conditions.
- The C.4(a) gives employees who work on either project 1 or 2, whereas the query in C.4(b) gives those who work on both the projects.

# Complex Conditions with and without a condition box as a part of QBE Query

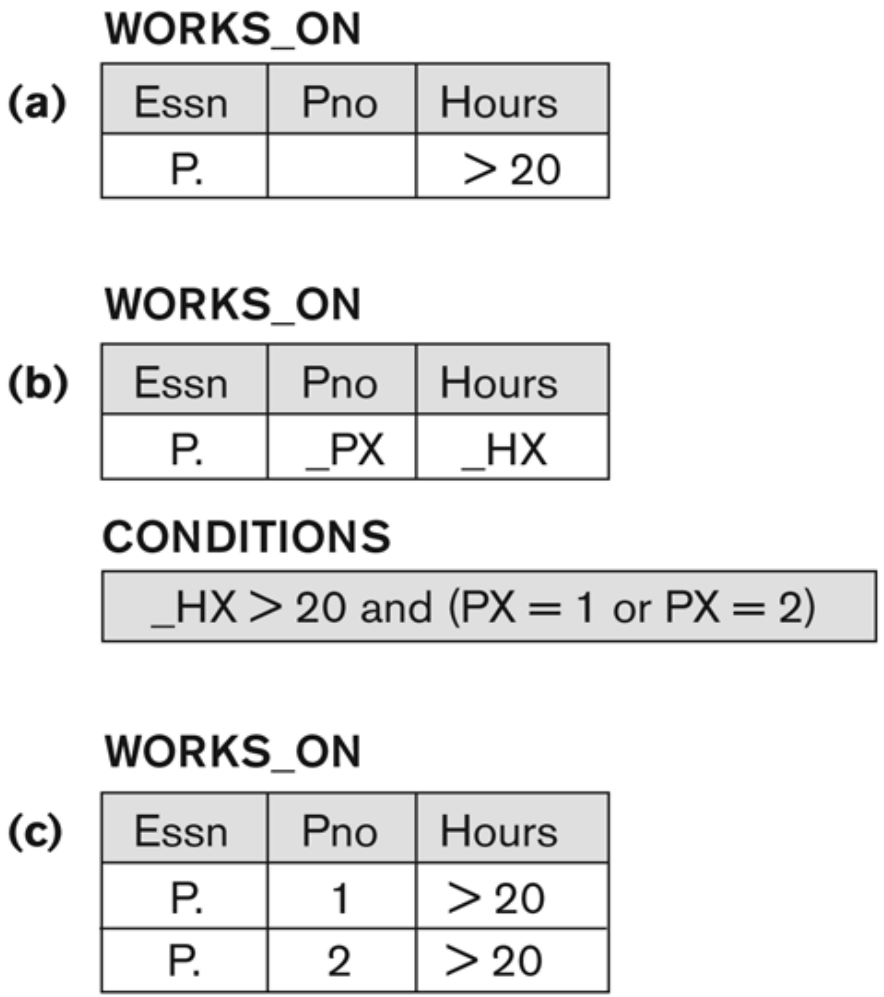


**Figure C.3**
Specifying complex conditions in QBE. (a) The query Q0A. (b) The query Q0B with a condition box. (c) The query Q0B without a condition box.

# Handling AND conditions in a QBE Query

**Figure C.4**
Specifying EMPLOYEES who work on both projects. (a) Incorrect specification of an AND condition. (b) Correct specification.

(a)

**WORKS_ON**

| Essn | Pno | Hours |
|---|---|---|
| P._ES | 1 | |
| P._ES | 2 | |

(b)

**WORKS_ON**

| Essn | Pno | Hours |
|---|---|---|
| P._EX | 1 | |
| P._EY | 2 | |

**CONDITIONS**

| _EX = _EY |
|---|

# JOIN in QBE : Examples

- The join is simply accomplished by using the same example element (variable with underscore) in the columns being joined from different (or same as in C.5 (b)) relation.
- Note that the Result is set us as an independent table to show variables from multiple relations placed in the result.

# Performing Join with common example elements and use of a RESULT relation

**Figure C.5**

Illustrating JOIN and result relations in QBE. (a) The query Q1. (b) The query Q8.

**(a)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| _FN | | _LN | | | _Addr | | | | _DX |

**DEPARTMENT**

| Dname | Dnumber | Mgrssn | Mgr_start_date |
|---|---|---|---|
| Research | _DX | | |

| **RESULT** | | | |
|---|---|---|---|
| P. | _FN | _LN | _Addr |

**(b)** **EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| _E1 | | _E2 | | | | | | _Xssn | |
| _S1 | | _S2 | _Xssn | | | | | | |

| **RESULT** | | | | |
|---|---|---|---|---|
| P. | _E1 | _E2 | _S1 | _S2 |

# AGGREGATION in QBE

- Aggregation is accomplished by using .CNT for count,.MAX, .MIN, .AVG for the corresponding aggregation functions
- Grouping is accomplished by .G operator.
- Condition Box may use conditions on groups (similar to HAVING clause in SQL – see Section 8.5.8)

# AGGREGATION in QBE : Examples

**(a) EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | P.CNT. | | |

**(b) EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | P.CNT.ALL | | |

**(c) EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| | | | P.CNT.ALL | | | | P.AVG.ALL | | P.G. |

**(d) PROJECT**

| Pname | Pnumber | Plocation | Dnum |
|---|---|---|---|
| P. | _PX | | |

**WORKS_ON**

| Essn | Pno | Hours |
|---|---|---|
| P.CNT.EX | G._PX | |

**CONDITIONS**

| CNT._EX > 2 |
|---|

**Figure C.6**
Functions and grouping in QBE. (a) The query Q23. (b) The query Q23A. (c) The query Q24. (d) The query Q26.

# NEGATION in QBE : Example

**Figure C.7**
Illustrating negation by the query Q6.

**EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| P. | | P. | _SX | | | | | | |

**DEPENDENT**

| Essn | Dependent_name | Sex | Bdate | Relationship |
|---|---|---|---|---|
| _SX | | | | |

# UPDATING in QBE : Examples

(a)

**EMPLOYEE**

| | Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|---|
| I. | Richard | K | Marini | 653298653 | 30-Dec-52 | 98 Oak Forest, Katy, TX | M | 37000 | 987654321 | 4 |

(b)

**EMPLOYEE**

| | Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|---|
| D. | | | | 653298653 | | | | | | |

(c)

**EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|
| John | | Smith | | | | | U._S*1.1 | | U.4 |

**Figure C.8**
Modifying the database in QBE. (a) Insertion. (b) Deletion. (c) Update in QBE.

# Chapter Summary

- Relational Algebra
  - Unary Relational Operations
  - Relational Algebra Operations From Set Theory
  - Binary Relational Operations
  - Additional Relational Operations
  - Examples of Queries in Relational Algebra
- Relational Calculus
  - Tuple Relational Calculus
  - Domain Relational Calculus
- Overview of the QBE language (appendix C)