

**Sistemi Informativi Geografici**  
anno accademico 2002 – 2003  
docente: Dr. Alberto Belussi

STANDARD ODMG 3.0 (Object Database Management Group)

Porzione della sintassi del linguaggio OQL (Object Query Language) da considerare per il corso di Sistemi Informativi Geografici.

**Sintassi**

Convenzioni: le parentesi quadre sono metasimboli che indicano opzionalità; i nomi tutti maiuscoli sono simboli non terminali, il simbolo “|” rappresenta un’alternativa.

Ogni interrogazione in OQL è definita attraverso l’elemento sintattico QUERY, che rappresenta l’assioma della grammatica.

QUERY → select [distinct] TARGET\_LIST from VARIABLE\_LIST [ where COND ]  
[ group by ATTRIBUTI\_PARTIZIONE [ having CONDH]]  
[ ordered by LISTA\_ORD ]

TARGET\_LIST → ATTRIBUTO | STRUTTURA | OP\_AGR ( VAR . PERCORSO )

ATTRIBUTO → VAR | VAR . PERCORSO | PERCORSO

Nota: VAR è una variabile definita nella VARIABLE\_LIST.

PERCORSO → NOME | NOME . PERCORSO

STRUTTURA → Struct ( LISTA\_CAMPI )

LISTA\_CAMPI → CAMPO | CAMPO , LISTA\_CAMPI

CAMPO → NOME : ATTRIBUTO | NOME : ( QUERY ) |  
NOME : OP\_AGR ( QUERY ) | OP\_AGR ( VAR . PERCORSO )

OP\_AGR → max | min | sum | avg | count

VARIABLE\_LIST → VARIABLE | VARIABLE , VARIABLE\_LIST

VARIABLE → VAR in NOME\_TIPO | VAR in ( QUERY )

NOME\_TIPO → NOME

VAR → LETTERA

LETTERA → a | ... | z

COND → COND\_ATOMICA | COND OP\_LOG COND | ( COND ) | not COND

OP\_LOG → = | != | < | > | <= | >=

COND\_ATOMICA → ESPR OP\_CONF ESPR | ESPR OP\_CONF CONST |  
VAR in VAR . PERCORSO | VAR in ( QUERY ) |  
CONST in VAR . PERCORSO | CONST in ( QUERY )

ESPR → ATTRIBUTO | OP\_AGR ( QUERY )

LISTA\_ORD → ORD , LISTA\_ORD | ORD

ORD → VAR . PERCORSO

ATTRIBUTI\_PARTIZIONE → ATTR\_PART ATTRIBUTI\_PARTIZIONE | ATTR\_PART

ATTR\_PART → NOME : VAR . PERCORSO |  
NOME : VAR . PERCORSO OP\_CONF CONST |  
NOME : VAR . PERCORSO OP\_CONF VAR . PERCORSO

CONDH → CONDH\_ATOMICA | CONDH OP\_LOG CONDH | not CONDH | ( CONDH )

CONDH\_ATOMICA → OP\_AGR ( QUERY ) OP\_CONF CONST |  
OP\_AGR ( QUERY ) OP\_CONF OP\_AGR ( QUERY )

CONST → "costante dei tipi di base"

NOME → "stringa di caratteri che inizia con una lettera"

## Interpretazione delle interrogazioni OQL

Ogni interrogazione OQL produce una collezione di oggetti di tipo T (bag<T> oppure set<T> se viene specificata la parola chiave `distinct`). Il tipo T viene completamente definito dalla clausola `TARGET_LIST`, dove, per la porzione del linguaggio OQL sopra definita, si possono produrre solo i tipi: `TIPO_BASE` oppure `struct(...)` con struttura arbitraria.

```
TIPO_BASE → String | Integer | Float | Char | Boolean |  
           Date | NOME
```

Le variabili definite nella `VARIABLE_LIST` assumono come valori gli oggetti che appartengono all'estensione del `NOME_TIPO` (o della `QUERY`) associate alla variabile (ad esempio, `x` in `Studenti` associa la variabile `x` all'estensione della classe `Studenti`) e che soddisfano la condizione eventualmente specificata nella clausola `where`.

Nel caso in cui sia presente una clausola `group by` con gli attributi di partizione:

$$A_1: e_1, \dots, A_n: e_n,$$

il risultato dell'interrogazione prima della clausola `TARGET_LIST` che definisce il tipo degli oggetti risultato, è del tipo seguente:

```
set<struct(A1: type_of(e1), ..., An: type_of(en),  
         partition: bag<type_of(oggetti aggregati)>>>
```

dove `type_of(oggetti aggregati)` dipende dalla clausola `VARIABLE_LIST`. Se `VARIABLE_LIST` contiene le seguenti assegnazioni di variabili:

$$x_1: c_1, \dots, x_n: c_n,$$

allora `type_of(oggetti aggregati)` risulta essere:

```
struct(x1: type_of(c1), ..., xn: type_of(cn))
```