Extending the ODMG Object Model with Time

Elisa Bertino Elena Ferrari
Giovanna Guerrini Isabella Merlo

July 1998

CONTENTS

MOTIVATIONS

Temporal databases: databases to efficiently manage the entire history of data over time

APPLICATIONS:

- medical and scientific domain
- business and financial context
- support of historical data

two *data models* to be extended:

- relational
- object oriented

MOTIVATIONS

Most of research effort in *temporal relational databases*

A temporal extension of the relational standard has been proposed: *TSQL 2*

IDEA:

to propose an extension of the object-oriented standard (ODMG): $T_$ODMG

ODMG

Basic modelling concepts:

- object
- literal

Rich type system:

- object types (classes): atomic and collection
- literal types: atomic, collection and structured

ODMG

two aspects in the type definition:

- interface specification
- implementation specification

Objects are characterized by:

- a *state*: a set of properties (attributes and relationships)
- a *behavior*: a set of operations (signatures)

Additional type information: *extents*, *keys*, *supertype*

T_ODMG types

ODMG types:

- object types:

  atomic: Person, Employee, Department

  collection: O_Set<Long>, O_Set<Person>

- literal types:

  atomic: Long, Short

  collection: Set<Long>, Array<Employee>

  structured: Date, Interval, Time, Timestamp

  struct: struct Complex{Float re; Float im;}

We extend the ODMG type system by adding **temporal types**

T_ODMG types

T_ODMG temporal types (ODMG$\mathcal{TT}$):

$$\text{ODMG}\mathcal{TT} = \{temporal(\texttt{t}) \mid \texttt{t} \in \text{ODMG}\mathcal{T}\}.$$

We have revisited the previous definitions according to this new set of types

EXAMPLES:

$temporal(\texttt{Short})$

$\texttt{Set} <temporal(\texttt{Short}) >$

$temporal(\texttt{List} < \texttt{Employee} >)$

Temporal dimension

ODMG provides structured literals to manage the time dimension:

`Date`, `Interval`, `Time`, `Timestamp`

We adopt such types to manage time in order to be "closer" to the ODMG data model

We introduce a new structured literal type: `TimeInterval`

Temporal dimension

```
interface TimeStamp {
   typedef Unsigned short ushort;
   Date date();
   Time time();
   ushort year();
   ushort month();
   ushort day();
   ushort hour();
   ushort minute();
   float second();
   TimeStamp plus(in Interval an_interval);
   TimeStamp minus(in Interval an_interval);
   Boolean is_equal(in TimeStamp t);
   Boolean is_greater(in TimeStamp t);
   ...};
```

`TimeStamp` interface

```
interface TimeInterval {
   TimeStamp start();
   TimeStamp end();

   TimeInterval equal(in TimeInterval I);
   Set<TimeInterval> union(in Set<TimeInterval> SI,in TimeInterval I);
   TimeInterval intersect(in TimeInterval I);
   ...};
```

`TimeInterval` interface

$T_$ODMG values

For each type `t` we have defined the set of values of type `t` at time t

$$[\![\, \texttt{t} \,]\!]_t$$

EXAMPLES:

$[\![\, \texttt{Person} \,]\!]_t = \{i_1, i_2, \ldots\}$

$[\![\, \texttt{Boolean} \,]\!]_t = \{true, false\}$

Particularly relevant:

$[\![\, \texttt{TimeStamp} \,]\!]_t = \mathcal{TIME}$

$[\![\, \texttt{TimeInterval} \,]\!]_t = \mathcal{TIME} \times \mathcal{TIME}$

$[\![\, temporal(\texttt{t}) \,]\!]_t =$

> $\{f \mid f : \mathcal{TIME} \to_p \bigcup_{t' \in \mathcal{TIME}} [\![\, \texttt{t} \,]\!]_{t'}$ such that $\forall t'$ if $f(t')$ is defined then $f(t') \in [\![\, \texttt{t} \,]\!]_{t'}\}$

Classes: interfaces

Class **interface**:

$$(c, c\_type, lifespan, attr, rel, meth)$$

$c \in \mathcal{CI}$ is the class identifier

$c\_type \in \mathcal{OTT} \cup \mathcal{CI}$ is the class type

$lifespan \in (\mathcal{TIME} \times \mathcal{TIME})$

$attr$ set of class attributes

rel set of class relationships

$meth$ set of pair (m_name, m_sign):

$m\_name \in \mathcal{MN}$ is the method name
m_sign is the signature of the method

Classes: interfaces

Class **attributes**:

$attr$ a set of 3-tuple $(a\_name, a\_nature, a\_type)$:

- $a\_name \in \mathcal{AN}$ is the attribute name
- $a\_nature \in \{\texttt{temporal,static,immutable}\}$
- $a\_type \in$ ODMG$\mathcal{ST}$ is the attribute domain type

The set rel is analogous

Classes: characteristics

Class **characteristics**:

$$(super, extent, keys)$$

$super$ set of direct superclasses of c

$extent = (e\_name, e\_set)$:

e_name is the extent name
e_set value of type $temporal(\texttt{Set<}c\texttt{>})$

$keys$ set of pairs $(k\_nature, prop)$:

$k\_nature \in \{\texttt{absolute, relative}\}$
$prop \in \mathcal{AN} \cup \mathcal{RN}$

Class $C = (int, char, impl)$:

int is the class interface

$char$ are the class characteristics

$impl$ is the class implementation

Classes: example

Class `Person` interface:

$c = c\_type = \texttt{Person}$

$lifespan = [start, now]$

$attr = \{(\texttt{ssn}, \texttt{immutable}, \texttt{String}),$
$\quad (\texttt{name}, \texttt{immutable}, \texttt{String}),$
$\quad (\texttt{address}, \texttt{static}, \texttt{String})\}$

$rel = \{(\texttt{spouse}, \texttt{temporal}, \texttt{Person}, (\texttt{spouse}, \texttt{Person})),$
$\quad (\texttt{children}, \texttt{temporal}, \texttt{Set<Person>}, r\_inv_c),$
$\quad (\texttt{parents}, \texttt{temporal}, \texttt{Set<Person>}, r\_inv_p)\}$

$meth = \{(\texttt{marriage}, \texttt{p} : \texttt{Person} \rightarrow \texttt{Boolean}),$
$\quad (\texttt{move}, \texttt{newaddress} : \texttt{String} \rightarrow)\}$

Class `Person` characteristics:

$super = \texttt{Object}$

$extent = (\texttt{person}, \{\langle [start, now], \{\texttt{i}_1, \ldots, \texttt{i}_4\}\rangle\}$

$keys = \{\langle \texttt{absolute}, \texttt{ssn}\rangle, \langle \texttt{relative}, \texttt{spouse}\rangle\}$

Objects

$o = (i, lifespan, v, r, class\text{-}history)$

$i \in \mathcal{OI}$ is the oid of o

$lifespan \in (\mathcal{TIME} \times \mathcal{TIME})$

$v \in \mathcal{V}$ and $v = \langle v_1^a\ a_1, \ldots, v_n^a\ a_n \rangle$

$r \in \mathcal{V}$ and $r = \langle v_1^r\ r_1, \ldots, v_n^r\ r_n \rangle$,

$class\text{-}history = \{\langle I_1, c_1 \rangle, \ldots, \langle I_n, c_n \rangle\}$

Objects: example

$i = \mathtt{i}_1$

$lifespan = I = \langle t_1, t_2 \rangle$ and:

$t_1 = \langle\langle 1965, 3, 21\rangle, \langle 00, 00, 00\rangle\rangle$ $t_2 = now$

$v = \langle$ $\{\langle I,$ `JS65I23`$\rangle\}$ **`ssn`**,

$\{\langle I,$ `JohnSmith`$\rangle\}$ **`name`**,

`"Fifth Avenue 275 NY" address` $\rangle$

$r = \langle$ $\{\langle\langle t_3, t_2\rangle,$ $\mathtt{i}_2\rangle\}$ `spouse`$\}$,

$\{\langle\langle t_4, t_5\rangle,$ $\{\mathtt{i}_3\}\rangle$

$\langle\langle t_6, t_2\rangle,$ $\{\mathtt{i}_3, \mathtt{i}_4\}\rangle\}$ `children`$\rangle$,

where:

$t_3 = \langle\langle 1990, 5, 14\rangle, \langle 00, 00, 00\rangle\rangle$

$t_4 = \langle\langle 1993, 12, 31\rangle, \langle 00, 00, 00\rangle\rangle$

$t_5 = \langle\langle 1995, 1, 17\rangle, \langle 23, 59, 59\rangle\rangle$

$t_6 = \langle\langle 1995, 1, 18\rangle, \langle 00, 00, 00\rangle\rangle$

$class\text{-}history = \{\langle I, \mathtt{Person}\rangle\}$

Objects

The object lifespan must satisfy several *temporal constraints*

EXAMPLES:

- if object o belongs to the extent of class c at time $t \Rightarrow t$ must be part of the lifespan of o

- for each instant of existence of object o, o must be instance of a cass c

Consistency notions

Object migrations $\Rightarrow$ the most specific class can vary over time

We require that each object must be a consistent instance of all the classes to which it belongs to

two steps:

- the set of attributes characterizing the object for each instant t of its lifespan must be determined
- the correctness of their values must be checked

Consistency notions

two kinds of consistency:

- *Historical consistency*. The values of the temporal properties of the object at a given instant are legal values for the temporal properties of the class

- *Static consistency*. The values of the static attributes of the object are legal values for the static attributes of the class

Object consistency

An object o is consistent if:

- for each instant t of its lifespan o contains a value for each temporal property of the class c to which it belongs at time t

- the value is of the correct type

- the instant t in which the property is defined belongs to the lifespan of o

Future work

- comply with ODMG 2.0
- impact of temporal types over object-oriented type systems
- temporal query language: T_OQL
- multiple granularities
- selective mechanism to record past values