Definicja obiektowego modelu danych: struktura i zachowanie

Podziękowania

- Dla Grzegorza Enzo Dołęgowskiego za wpisanie moich notatek do komputera.

Relacyjna baza danych (przypomnienie)

Pojęcia pierwotne

A – zbiór nazw atrybutów
D – zbiór wartości atomowych (napisy, liczby, daty, wartości logiczne)
E – zbiór typów atomowych (integer, float, string, boolean, date)
K – zbiór nazw relacji (potem będzie to zbiór klas)

Schemat tabel (relacji)

kol : $\mathbf{K} \rightarrow P_{\text{fin}}(\mathbf{A})$ (nazwie relacji przyporządkowujemy skończone zbiory nazw kolumn)

typ : $\mathbf{K} \times \mathbf{A} \rightarrow \mathbf{E}$ (nazwie relacji i nazwie atrybutu przyporządkowujemy typ elementarny)

Schemat bazy relacyjnej

$SCH_{rel} = (\mathbf{K}, \text{kol}, \text{typ})$

Krotka relacji

Jeśli $R \in \mathbf{K}$, to krotka o schemacie R jest funkcją przypisującą wartości nazwom kolumn:

$$t : \text{kol}(R) \rightarrow \mathbf{D}$$

dla $A \in \text{kol}(R)$, $t(A) \in \mathbf{D}_{\text{typ}(R,A)}$

W – zbiór wszystkich możliwych krotek

Egzemplarz relacji

Jeśli $R \in \mathbf{K}$, to relacja o nazwie R jest skończonym zbiorem krotek o schemacie R.

Egzemplarz relacyjnej bazy danych

Funkcja val, która przypisuje nazwom relacji egzemplarze relacji:

$INST_{rel} = (\text{val})$
val : $\mathbf{K} \rightarrow P_{\text{fin}}(\mathbf{W})$

Poprawność relacyjnej bazy danych

$SCH_{rel} = (\mathbf{K}, \text{kol}, \text{typ})$
$INST_{rel} = (\text{val})$

Dla każdego $R \in \mathbf{K}$ i dla każdej krotki $t \in \text{val}(R)$, t ma schemat R.

Obiektowa baza danych

Pojęcia pierwotne

A – zbiór nazw atrybutów
O – zbiór identyfikatorów obiektów (OID)
D – wartości atomowe (napisy, liczby, wartości logiczne)
K – zbiór nazw klas i IsA częściowy porządek na **K**
E – zbiór typów atomowych (integer, float, string, boolean, date)

Wartości złożone (*W*)

W to najmniejszy zbiór o następujących właściwościach:

1. $\mathbf{D} \subseteq \mathbf{W}$ (skalar)
2. $\mathbf{O} \subseteq \mathbf{W}$ (referencja)
3. Jeśli $A_1, A_2, ..., A_n \in \mathbf{A}$ ($A_i \neq A_j$ dla $i \neq j$) oraz $w_1, w_2, ..., w_n \in \mathbf{W}$,
   to $[A_1{:}w_1, A_2{:}w_2, ..., A_n{:}w_n] \in \mathbf{W}$ (krotka)
4. Jeśli $w_1, w_2, ..., w_m \in \mathbf{W}$ ($w_i \neq w_j$ dla $i \neq j$), to $\{w_1, w_2, ..., w_m\} \in \mathbf{W}$ (zbiór)

Obiekt

Para (o, w) przy czym $o \in \mathbf{O}$, $w \in \mathbf{W}$.

Typy (T)

T to najmniejszy zbiór o następujących właściwościach:

1. $\mathbf{E} \subseteq \mathbf{T}$ (typ elementarny)
2. $\mathbf{K} \subseteq \mathbf{T}$ (typ referencyjny)
3. Jeśli $T \in \mathbf{T}$, to $\{T\} \in \mathbf{T}$ (typ zbiorowy)
4. Jeśli $A_1, A_2, ..., A_n \in \mathbf{A}$ ($A_i \neq A_j$ dla $i \neq j$) oraz $T_1, T_2, ..., T_n \in \mathbf{T}$, to
   $[A_1{:}T_1, A_2{:}T_2, ..., A_n{:}T_n] \in \mathbf{T}$ (typ krotkowy)

Hierarchia typów

$\leq$ częściowy porządek na **T** (podtyp $\leq$ nadtyp)

$\leq$ jest najmniejszym częściowym porządkiem o następujących właściwościach:

1. Jeśli K IsA L, to $K \leq L$ [Pracownik IsA Osoba, więc Pracownik $\leq$ Osoba]
2. Jeśli $T = [A_1{:}T_1, A_2{:}T_2, ..., A_n{:}T_n] \in \mathbf{T}$ i $U = [B_1{:}U_1, B_2{:}U_2, ..., B_m{:}U_m] \in \mathbf{T}$ oraz $n \leq m$ i
   dla każdego $i = 1, 2, ..., n$ istnieje j takie, że $A_i = B_j$ i $U_j \leq T_i$, to $U \leq T$.
3. Jeśli $T \leq U$, to $\{T\} \leq \{U\}$.

Schemat struktury

$SCH_{struct} = (\mathbf{K}, \text{IsA}, \text{typ})$

typ : $\mathbf{K} \rightarrow \mathbf{T}$ musi spełniać warunek: K IsA L $\Rightarrow$ typ(K) $\leq$ typ(L)

Rozszerzenie bazowe

Zbiór obiektów klasy

inst : $\mathbf{K} \rightarrow P_{\mathrm{fin}}(\mathbf{O})$

dla każdego K, L $\in \mathbf{K}$, K $\neq$ L zachodzi inst(K) $\cap$ inst(L) = $\varnothing$

Rozszerzenie

Zbiór obiektów klasy i jej podklas

inst* : $\mathbf{K} \rightarrow P_{\mathrm{fin}}(\mathbf{O})$

$$\mathrm{inst}^*(K) = \bigcup_{\substack{L \in \mathbf{K} \\ L \text{ IsA } K}} \mathrm{inst}(L)$$

Dziedzina (typu)

dom : $\mathbf{T} \rightarrow P(\mathbf{W})$

1. dom(integer) = zbiór liczb całkowitych
2. dom(K) = inst*(K) dla K $\in \mathbf{K}$
3. Jeśli T = $[A_1{:}T_1, A_2{:}T_2, ..., A_n{:}T_n] \in \mathbf{T}$, to
   dom(T) = $\{[A_1{:}w_1, A_2{:}w_2, ..., A_n{:}w_n]$: dla i = 1, 2, ..., n $w_i \in \mathrm{dom}(T_i)\}$
4. Jeśli T = {U}, to dom(T) = $P_{\mathrm{fin}}(\mathrm{dom}(U))$

Egzemplarz bazy danych (struktura)

Jeśli SCH_{struct} = ($\mathbf{K}$, IsA, typ) jest schematem struktury, to

$INST_{struct}$ = (inst, val)

jest egzemplarzem schematu SCH_{struct} o ile spełnione są następujące warunki:

(1) inst jest bazowym rozszerzeniem inst : $\mathbf{K} \rightarrow P_{\mathrm{fin}}(\mathbf{O})$
(2) val jest wartościowaniem val : $\mathbf{O} \rightarrow \mathbf{W}$
(3) o $\in$ inst(K), K $\in \mathbf{K} \Rightarrow$ val(o) $\in$ dom(typ(K))

klasy typy

$$\begin{tikzcd} \mathbf{K} \arrow[r, "typ"] \arrow[d, "inst"'] & \mathbf{T} \arrow[d, "dom"] \\ \mathbf{O} \arrow[r, "val"'] & \mathbf{W} \end{tikzcd}$$

obiekty wartości

Dziedziczenie struktury

Niech SCH_{struct} = (**K,** IsA, typ) będzie schematem struktury.

Jeśli typ(K) jest krotkowy, to typ*(K) zawiera wszystkie składowe typ(K) oraz wszystkie A:T takie, że A:T jest składową L taką, że K IsA L i nie istnieje M różne od K i L takie, że A jest składową M i K IsA M i M IsA L.

SCH_{struct} jest poprawny o ile:

1. przekształcenie typ* istnieje, oraz

2. K IsA L $\Rightarrow$ typ*(K) $\leq$ typ*(L)

Wartości domyślne atrybutów

$val_{st} : \mathbf{K} \rightarrow \mathbf{W}_{krotkowe}$

Musi być spełniać warunek — jeśli K $\in$ **K** i A:w jest składową val_{st}(K), to typ*(K) zawiera składową A. Niech T będzie typem A w typ*(K). Wówczas musi być prawdą, że w $\in$ dom(T).

Schemat struktury z wartościami domyślnymi

SCH_{struct} = (**K,** IsA, typ, val_{st})

Wartościowanie z wartościami domyślnymi val*: $\mathbf{O} \rightarrow \mathbf{W}$

val*(o) jest najuboższą wartością krotkową, która zawiera val(o) i spełnia poniższy warunek:

Jeśli

1. typ*(K) jest krotkowy i ma składową A:T,
2. istnieje klasa L taka, że A:w jest składową val_{st}(L) i K IsA L
3. nie istnieje klasa M, taka że val_{st}(M) ma składową A oraz K IsA M i M IsA L [nie istnieje klasa pośrednia między K i L, która definiowałaby wartość domyślną dla A],
4. val(o) nie ma składowej A,
5. o $\in$ inst(K),

to:

6. A:w jest składową val*(o).

Egzemplarz bazy danych z dziedziczeniem i wartościami domyślnymi

$INST_{struct}$ = (inst, val) jest egzemplarzem SCH_{struct} = (**K,** IsA, typ, val_{st}), o ile spelnione są następujące warunki:

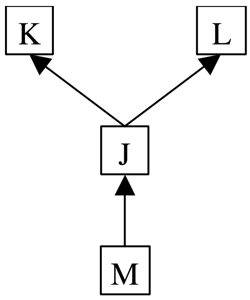
1. Przekształcenie typ* istnieje.
2. Przekształcenie val* istnieje.
3. o $\in$ inst(K), K $\in$ **K** $\Rightarrow$ val*(o) $\in$ dom(typ*(K))

$$\begin{tikzcd}
\text{klasy} & & \text{typy} \\
\mathbf{K} \arrow[rr, "\text{typ*}"] \arrow[d, "\text{inst}"'] & & \mathbf{T} \arrow[d, "\text{dom}"] \\
\mathbf{O} \arrow[rr, "\text{val*}"'] & & \mathbf{W} \\
\text{obiekty} & & \text{wartości}
\end{tikzcd}$$

Brak konfliktów dziedziczenia

SCH_{struct} = (**K,** IsA, typ, val_{st}) jest wolny od konfliktów dziedziczenia, wtw. dla każdej trójki K, L, M $\in$ **K** takich, że M IsA K, M IsA L zachodzi

1. Jeśli typ(K) i typ(L) zawiera składową A, ale typ(M) jej nie zawiera, to istnieje klasa J, taka że typ(J) zawiera A oraz J IsA K, J IsA L, M IsA J.
2. Dokładnie to samo odnosi się do val_{st}.



Model zachowania

M – zbiór nazw metod

Sygnatura

$M : K \times T_1 \times ... \times T_k \rightarrow T \qquad M \in \mathbf{M}; K \in \mathbf{K}; T_1, T_2, ..., T_k \in \mathbf{T}$

Schemat zachowania

SCH_{zach} = (**K**, IsA, **S**)

S to skończony zbiór sygnatur metod o następującej właściwości

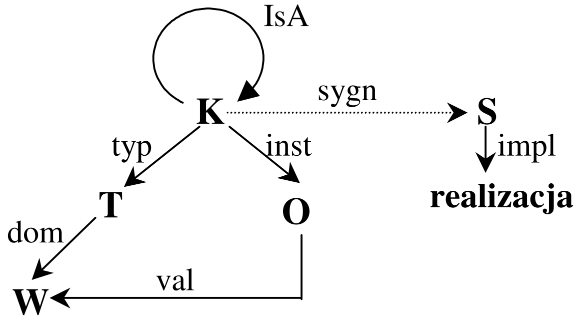
Jeśli $(M : K \times T_1 \times ... \times T_k \rightarrow T) \in \mathbf{S}$ i $M : K \times U_1 \times ... \times U_m \rightarrow U)$,
to $k = m$, $U \leq T$ i dla $i = 1, 2, ..., k$ $T_i \leq U_i$

Egzemplarz schematu zachowania

SCH_{zach} = (**K**, IsA, **S**)
$INST_{zach}$ = (inst, impl)
impl : **S** $\rightarrow$ Implementacje metod
$impl((M : K \times T_1 \times ... \times T_k \rightarrow T)) = I$
$I : dom(K) \times dom(T_1) \times ... \times dom(T_k) \rightarrow dom(T)$ (funkcja częściowa)

Pojedyncza dyspozycja = wybór metody tylko zależy tylko pierwszego argumentu.
Wielokrotna dyspozycja = wybór metody tylko zależy od wszystkich argumentów.

Diagram modelu formalnego



Schemat i egzemplarz obiektowej bazy danych

$SCH_{struct} = (\mathbf{K}, IsA, typ, val_{st})$	schemat struktury
$SCH_{zach} = (\mathbf{K}, IsA, \mathbf{S})$	schemat zachowania
$SCH = (\mathbf{K}, IsA, typ, val_{st}, \mathbf{S})$	schemat bazy obiektów
$INST_{struct} = (inst, val)$	egzemplarz schematu struktury
$INST_{zach} = (inst, impl)$	egzemplarz schematu zachowania
$INST = (inst, val, impl)$	egzemplarz bazy obiektów