

Synapsově-orientovaný (SO) přístup, vztahově orientované úložiště (CODS)

PA116 - 10

(c) Zdenko Staníček, Michal Oškera, srpen 2006

Obsah

- Opakování – principy a motivace
- Vztahově orientovaný přístup
- Teoretická východiska
- TI a NP hierarchie typů
- Dvojid, synapse, bambus a bambusiště
- Informační schopnost bambusu
- Souvislost metody DM HIT a CODS
- Srovnání OODS a CODS
- Univerzální CODS

Fritjof Capra: Tkáň života

- *„Obrovským šokem vědy dvacátého století bylo to, že systémům nemůžeme porozumět na základě jejich analýzy. Vlastnosti částí nejsou skutečnými vlastnostmi, ale mohou být pochopeny pouze v kontextu většího celku. . . .*
- *Systemové myšlení je kontextuální, což je opakem analytického myšlení.*
- *Analýza znamená pojímat něco odděleně v zájmu porozumění.*
- *Systemové myšlení znamená umístit to něco do kontextu většího celku (v zájmu porozumění).“*

Základní principy systémového myšlení (F. Capra)

- 1. **posun od částí k celku** - tj. akceptování předpokladu, že chování celku nelze poznat analýzou vlastností jeho částí
- 2. schopnost **zaměřit pozornost** na různé **úrovně systému**
- 3. poznání, že **části neexistují** - ve skutečnosti to, co v prvním bodě nazýváme částí, je pouze uspořádání v neoddělitelném předivu vztahů; proto **posun od částí k celku lze pokládat za posun od objektů ke vztahům**

Pragmatika

- Princip pragmatického vnímání objektů
 - Tytéž objekty našeho zájmu vykazují v různém čase či pod různými úhly pohledu různé vlastnosti a různé souvislosti směrem k jiným objektům
 - Tyto změny vlastností však nejsou objektivní, ale naopak subjektivní a jsou v souladu se snižující se či zvyšující se mírou našeho zájmu o tyto objekty, která je určována pragmatickou potřebou tyto objekty v různém čase nebo situacích různě vnímat

Pragmatika - příklady

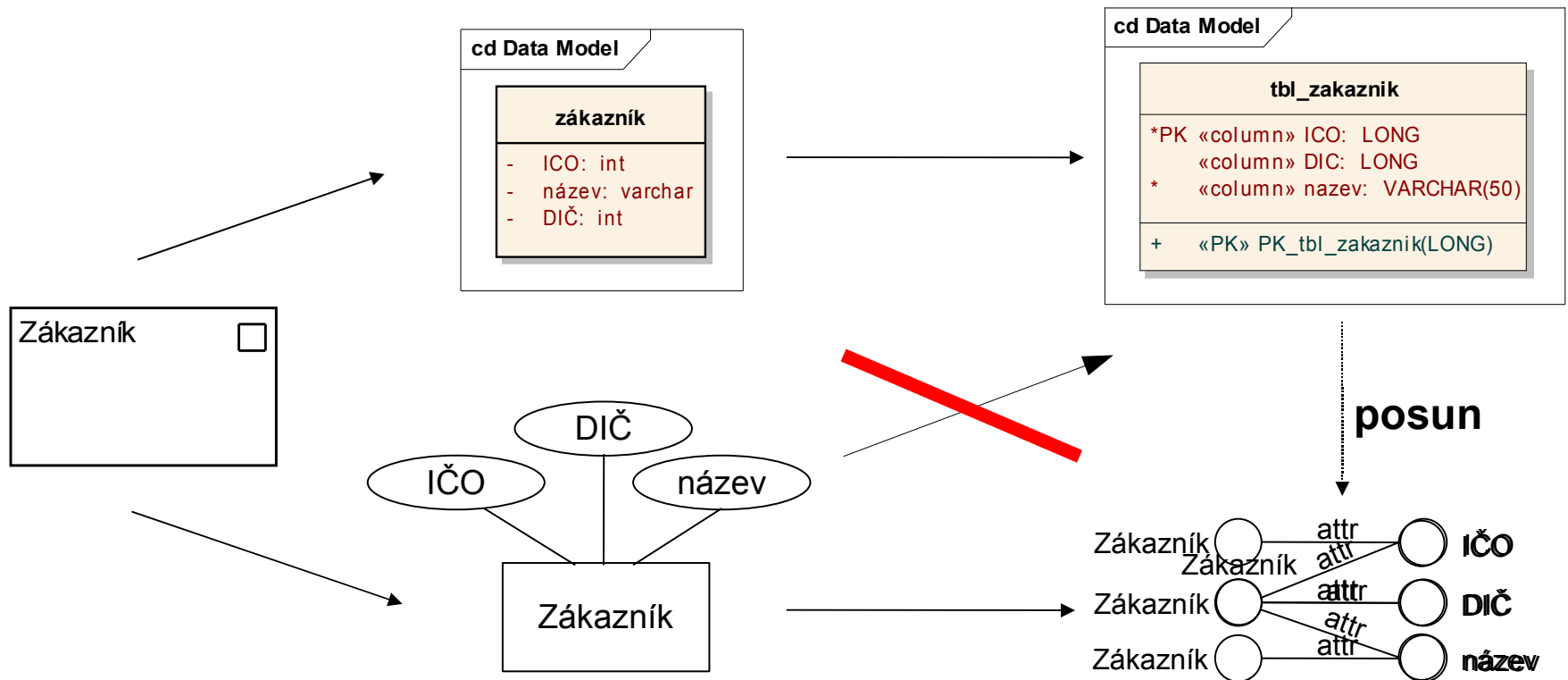
- Různé role
 - Objekt typu (#Zaměstnanec) je jinak vnímán
 - Personalistou
 - Šéfem výroby
 - ...
- Různé situace (různý čas)
 - Objekt typu (#Žena/#Muž) je jinak vnímán(a)
 - Při prvním setkání
 - Před svatbou
 - Po ní

Vztahově orientovaný přístup

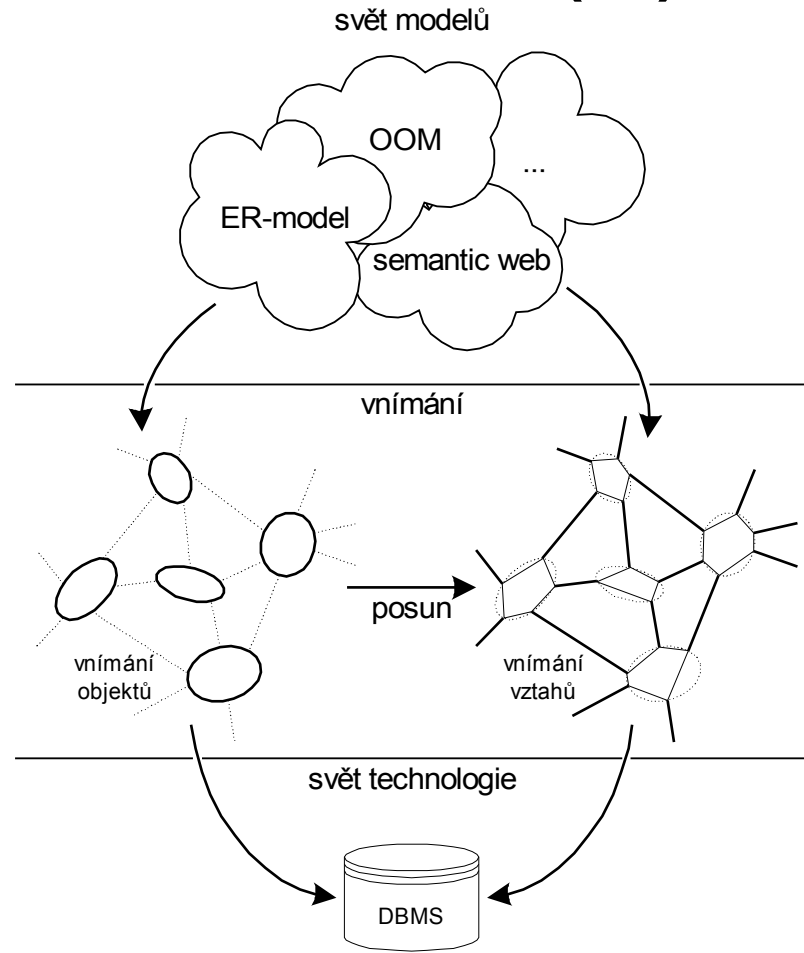
- Podstatou *vztahově orientovaného přístupu k ukládání dat* je ukládání výskytů vztahů a nikoliv ukládání předem určených celků v podobě na začátku fixně daných tabulek
- **Princip vztahového vnímání modelů**
*Při pohledu na model čehokoliv, který je reprezentovatelný grafem, a přemýšlení o něm, se zaměřujeme **primárně na vztahy (hrany grafu)** mezi jednotlivými částmi modelu (uzly grafu).*

Vztahově orientované vnímání modelů (1)

- posun od objektů ke vztahům



Vztahově orientované vnímání modelů (2)



Vztahově orientovaný přístup - očekávání

- Ukládání dat způsobem zohledňujícím princip **pragmatického vnímání objektů**
- Realizace **univerzálního datového úložiště** stávajících a budoucích silných modelovacích nástrojů (např. DMT)
- Základ **nového způsobu organizace dat**, který dá vzniknout novým možnostem dotazování a práce se znalostmi vůbec – tento způsob je alternativou nikoli náhradou

Teoretická východiska

- **TIL**
 - P. Tichý
 - Rozvětvená hierarchie typů – pouze insiprace
 - TI a NP hierarchie typů
- **HIT**
 - M. Duží, P. Materna, F. Krejčí, Z. Staníček
 - Místo na **BS** je pozornost soustředěna na **ATR**
 - Binarizační princip (opakování)

Typ a kategorie z hlediska CODS

- Typ je speciální případ kategorie
 - Uvážíme-li množinu kategorií K a množinu typů T , které zavádíme v rámci tvorby libovolného systému, pak platí: $T \subset K$
- Každý objekt je přímo právě jednoho typu / bydlí přímo právě v jednom typu
 - Říkáme, že objekt je instancí typu
- Nadtyp-podtyp (NP hierarchie)
- Zmiňování/užívání typu

Sorty HITu a typy v CODS

- Každé entitní sortě E z libovolného modelu přiřadíme typ T_E , který slouží pro pořádání její konečné podtřídy, jejíž obsah je výsledkem našeho aktuálního poznávání a potřeby
- Podobně každé deskriptivní sortě D přiřadíme typ T_D
- **Sorty z metody HIT nejsou totéž co typy v CODS!**

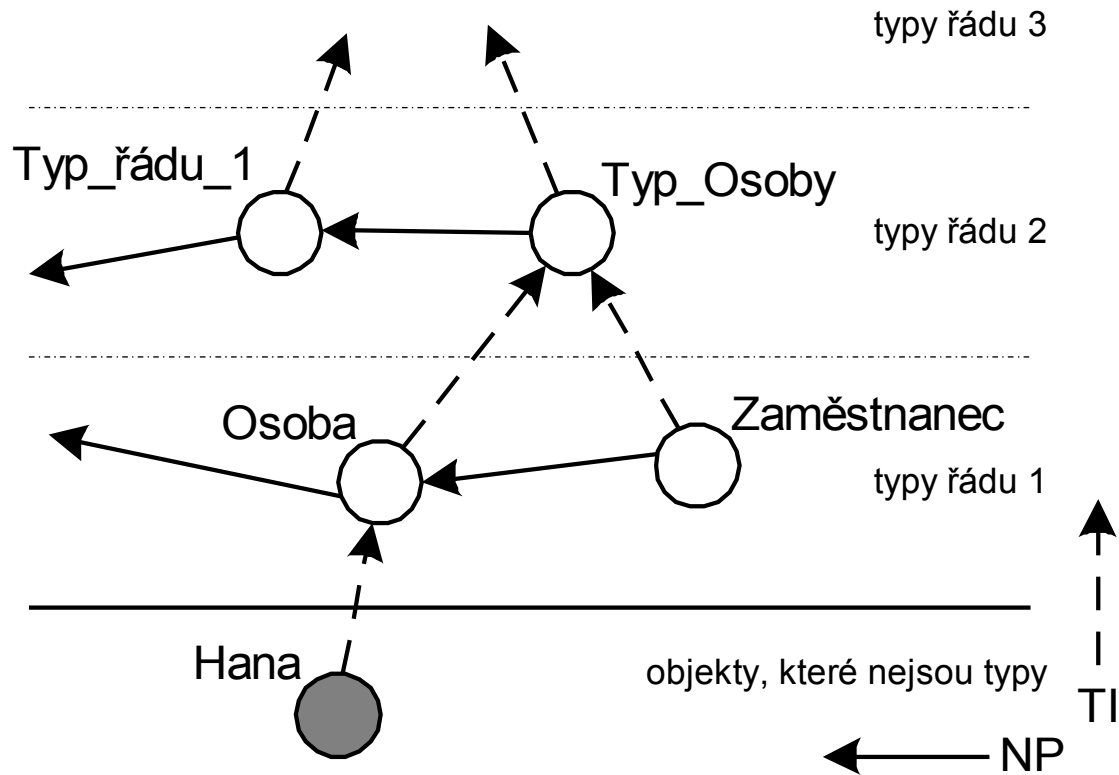
TI a NP hierarchie typů

- Dva základní druhy vztahů mezi typy, kterými lze uspořádat typy do dvou hierarchií
- TI hierarchie
 - Pořádá typy dle vztahu **typ-instance**
- NP hierarchie
 - Pořádá typy dle vztahu **nadtyp-podtyp**

TI hierarchie

- Objekty zájmu, které nejsou typy, jsou pořádány **typy 1. řádu**
 - Typy 1. řádu jsou listy v pomyslném stromě reprezentujícím TI-hierarchii
- Typy řádu $N+1$ pak pořádají **typy řádu N**
 - Typy řádu většího než 1 nazýváme **typy vyšších řádů** nebo také **meta-typy**
- **TI hierarchie není objektivně dána, nýbrž je výsledkem procesu modelování a pragmatického poznávání aktuálního světa!**
 - Stejně tak jako NP hierarchie

TI hierarchie - příklad



- Více příště

Princip identifikace

- Mějme model $M = (BS, ATR, RUL, OPE)$, nechť $E \in BS$ je entitní sorta, potom v ATR musí existovat, takový atribut A typu $((Wrlld, Time) \rightarrow (E \rightarrow D))$, pro který platí, že:
 - je totální funkcí,
 - D je deskriptivní sorta a že
 - atribut A' typu $((Wrlld, Time) \rightarrow (D \rightarrow E))$ je jeho přípustná singulární rotace
- Potom se A nazývá **identifikační atribut entitní sorty E** a D se nazývá **identifikací sorty E**
 - Identifikátor jakožto prvek identifikace D jednou přiřazený konkrétnímu objektu sorty E se nesmí žádnou aktualizací změnit

Rozšíření principu identifikace

- Pro libovolné dvě sorty E_1 a E_2 s identifikačními atributy po řadě A_1 a A_2 platí
 - Jestliže je D identifikací sorty E_1 a zároveň sorty E_2 musí být pro libovolné dva prvky $e_1 \in E_1$ a $e_2 \in E_2$ zajištěno, že budou identifikovány různými prvky identifikace D ,
 - tj. aplikace funkce, která je konstruována konstrukcí $\lambda w \lambda t [\neq [A_1 e_1]_{wt} [A_2 e_2]_{wt}]$, kde $\neq / (D, D) \rightarrow \text{Bool}$ testuje prvky sorty D na neshodu, musí vracet vždy hodnotu *true*

Identifikace v CODS

- V CODS identifikujeme právě ty objekty, které jsou pořádány některým z typů z námi zvolené TI hierarchie – tj. i typy samotné
- Objekty jsou identifikovány pomocí prvků typu **univerzální světové identifikace (UWID)**
- **Každá instance typu UWID identifikuje nejvýše jeden objekt neohledě na to, jakého je tento objekt typu!**

Identifikace UWIDy

- Požadované vlastnosti uwidů
 - Celosvětová jedinečnost
 - Nemají omezený rozsah
- Důsledky
 - Identifikace bez obav o přetečení počítačů
 - Spolehnutí, že jakékoliv dvě různé elementární znalosti v rámci jednoho SW systému jsou identifikovány různými identifikátory
 - Při interakci dvou SW systémů nedojde ke kolizi identifikátorů

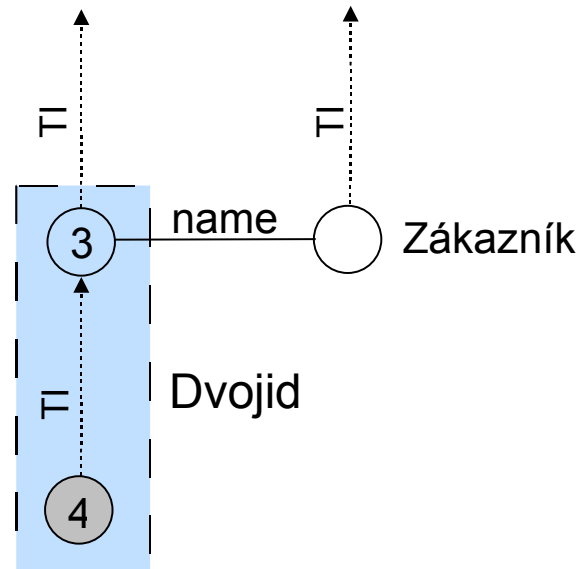
Pojetí objektu v CODS

- Objekty si lze představit jako izolované body v mentálním prostoru, které nemají žádnou vlastnost nutně
- Antiesencialismus – P. Materna
 - Logické skoby na vlastnosti, nahá individua

Dvojíd - motivace

- Označení

- Objekt, který má vlastnost býtí typem: ○
- Objekt, který nemá vlastnost býtí typem: ●



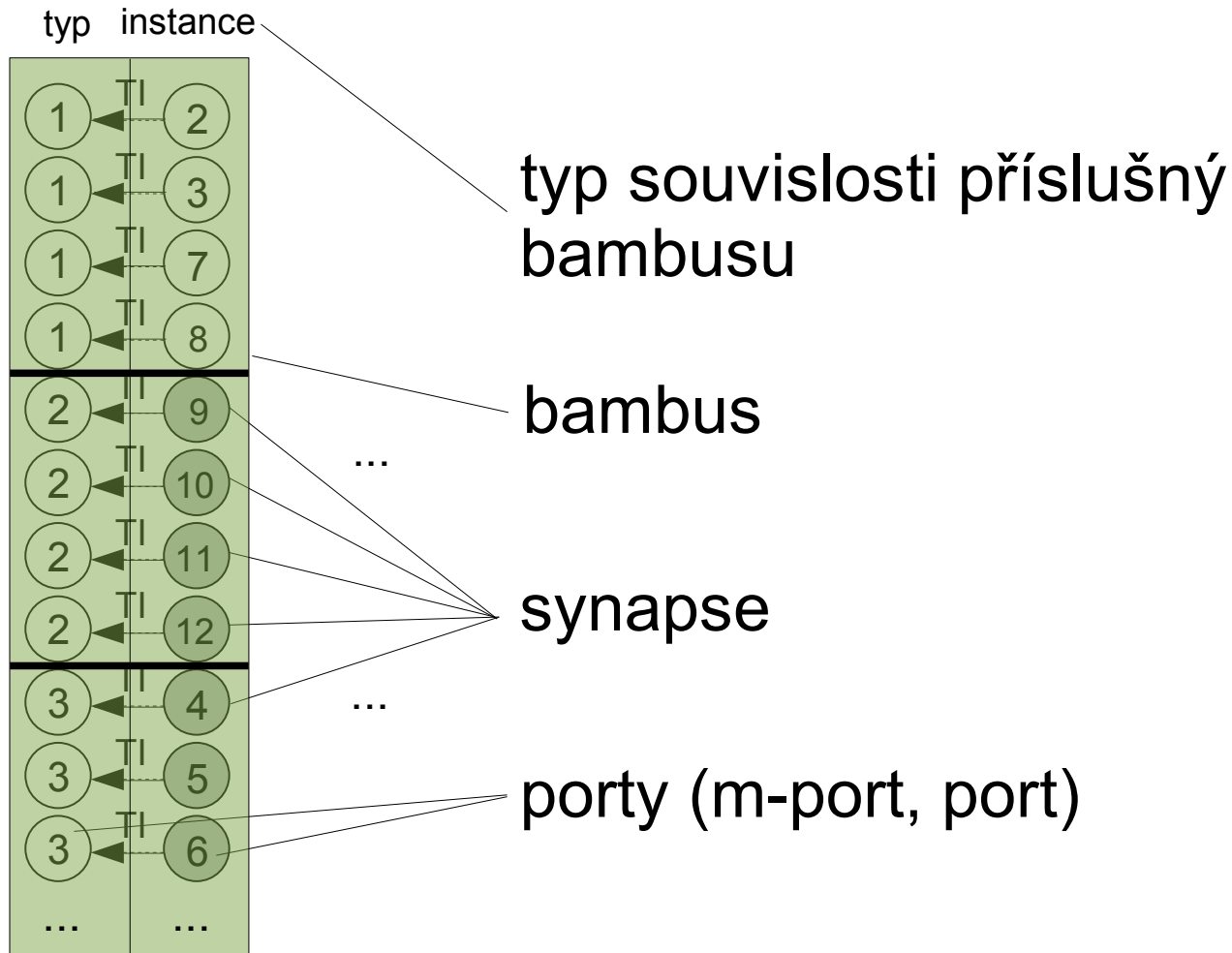
Dvojid - definice

- Necht' $tuwid \in UWID$, $iuwid \in UWID$ jsou uwidy
- Jestliže platí, že objekt identifikovaný uwidem $iuwid$ je instancí typu identifikovaného uwidem $tuwid$, pak uspořádanou dvojici **($tuwid$, $iuwid$)** nazýváme **dvojid**, anglicky **dvoid**.
- (První člen dvojice budeme nazývat m-port, druhý pak port)

Bambus a synapse (1)

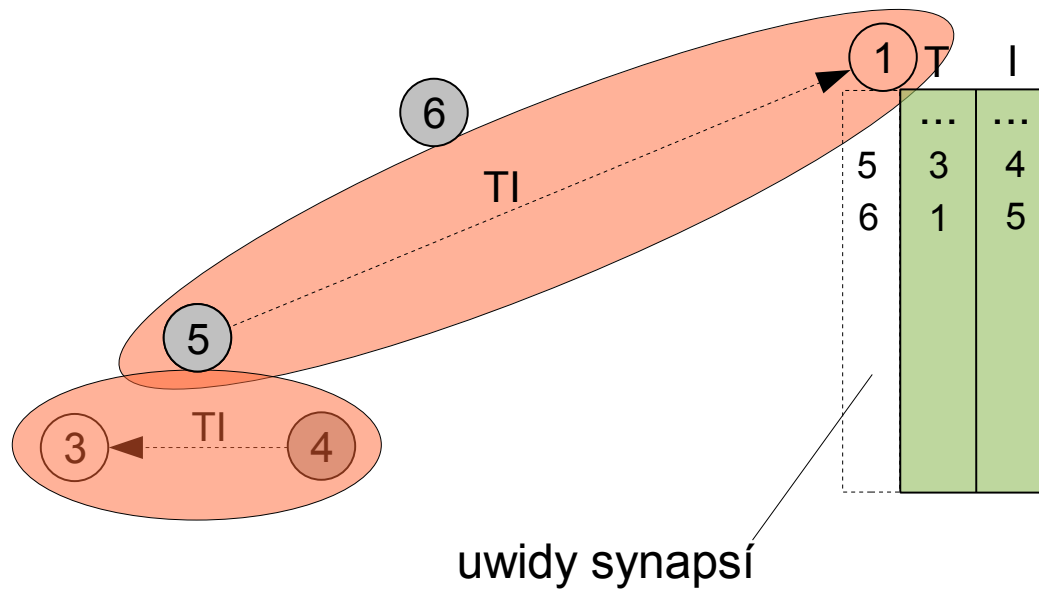
- Pomocí bambusů ukládáme konkrétní instance binárních souvislostí
- Synapse je výskyt daného typu souvislosti
- **Bambusy jsou kontejnery na synapse, ba co víc – jsou to typy na ukládání výskytů souvislostí!**
 - Formální HITovské definice – viz DP OSK, strana 35-42

Bambus a synapse (2)



Synapse jako objekt

- Synapse jakožto výskyt jistého druhu souvislosti se může legitimně stát objektem zájmu



- **Nuntá podmínka, aby CODS mohlo být úložištěm pro silný modelovací nástroj!**

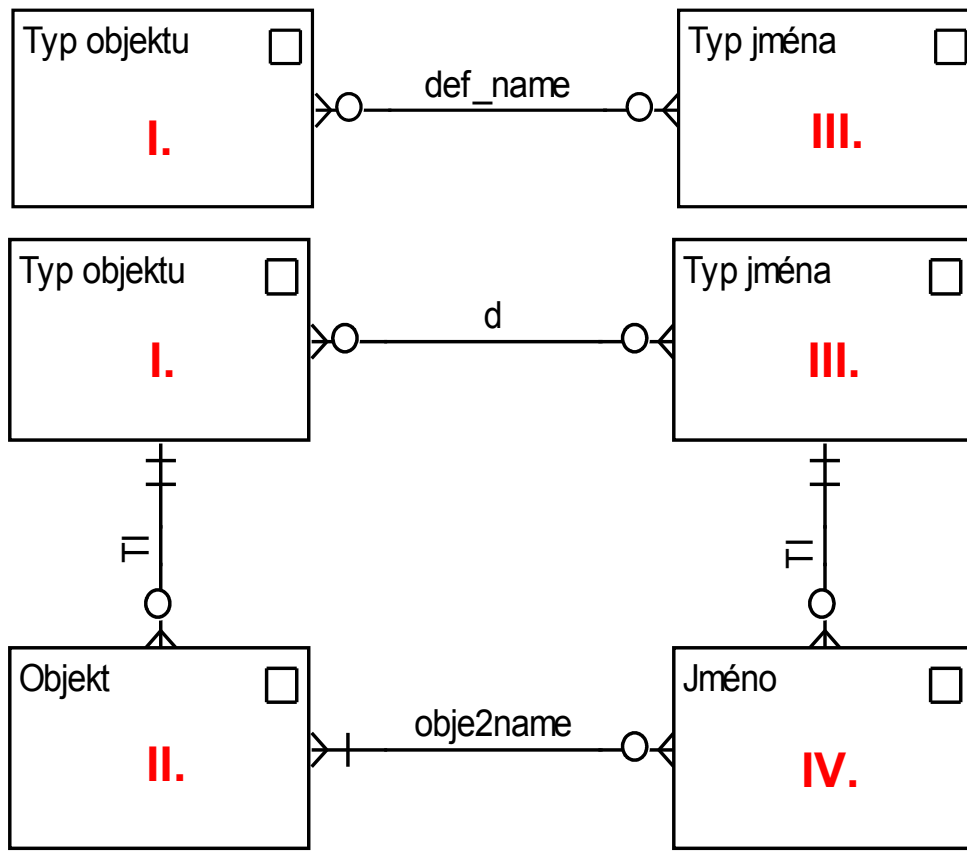
Typy bambusů

- Instanční bambus
 - Pořádá všechny výskyty vztahu **typ-instance**
- Tenký/definiční bambus S
 - Pořádá výskyty vztahu mezi dvěma typy $T1$ a $T2$
 - Tyto výskyty vyjadřují **možnost existence** vztahu **dané sémantiky S** mezi instancemi typů $T1$ a $T2$
- Tlustý bambus S
 - Pořádá **existující** výskyty vztahu **dané sémantiky S** mezi dvěma konkrétními instancemi typů $T1$ a $T2$

Vlastnosti bambusů

- Typ
 - Instanční, tlustý, tenký
- Název
- Druh
 - Deskriptivnost (deskriptivní, nedeskriptivní)
- Sémantika a kardinalita bambusu
 - Vycházejí ze sémantiky a kardinality binárního HIT-atributu, kterého výskyty bambus pořádá

Informační schopnost bambusu



def_name

I. III.

...	...
1	2
3	2
...	...

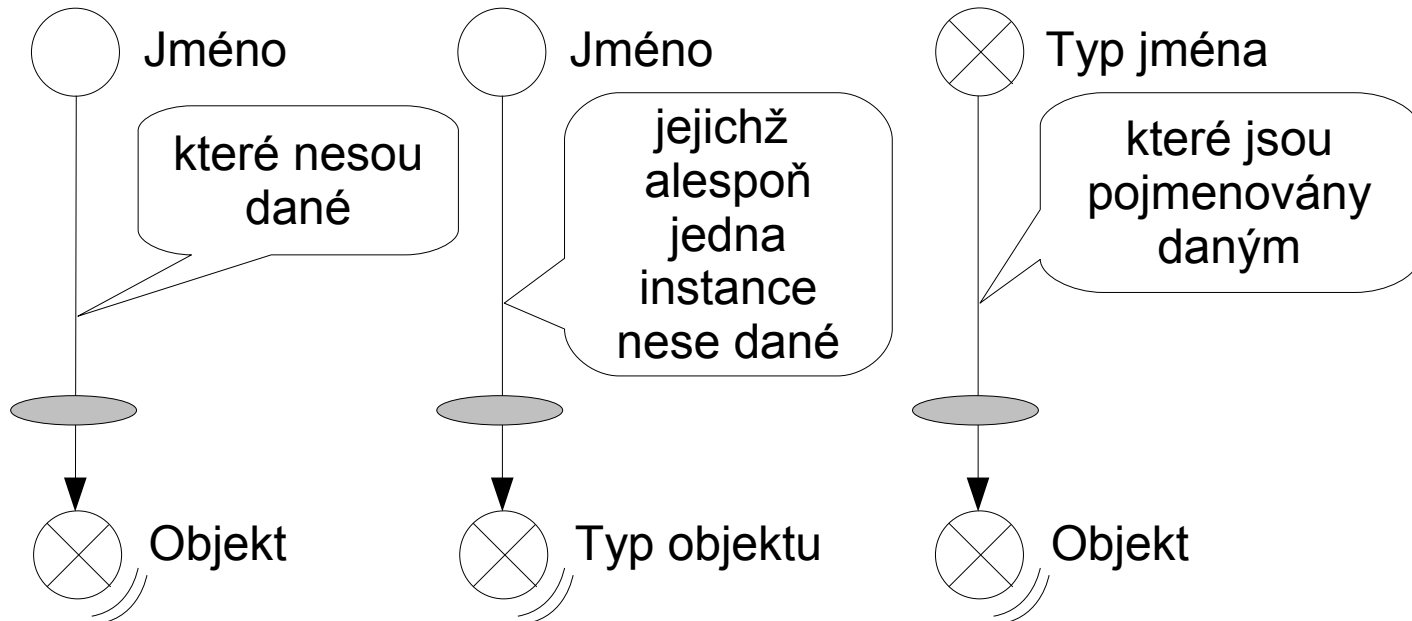
object2name

I. II. III. IV.

...
1	3	2	Zákazník
3	4	2	Firma1
3	5	2	Firma2
...

Informační schopnost bambusu

- Zajímavé HIT-atributy definovatelné nad bambusem obje2name



Vztahově orientované úložiště (CODS) aka Bambusiště

- Bambusiště B_{forest} je trojice skládající se z
 - Právě jednoho **instančního bambusu** (I)
 - Blíže neurčeného počtu **dvojic tenkých a tlustých bambusů** se specifikovanými vlastnostmi, zejména definovanou sémantikou (B_{spec})
 - Množiny tvrzení konzistence (IC)
- $B_{\text{forest}} = (I, B_{\text{spec}}, IC)$

Vztah metody HIT a CODS

- Výstupem datového modelování dle metody HIT je datový model reprezentovaný trojicí (BS, ATR, CC)
- Bambusiště je reprezentováno trojicí (I, B_{spec}, IC)
- \Rightarrow Potřeba algoritmu převodu datového modelu dle metody HIT na bambusiště, které umožní ukládat data oblasti, kterou datový model popisuje

Algoritmus převodu DM dle HIT na přísušné CODS (1)

- Vstup: datový model $M = (BS, ATR, CC)$
 - Výstup: bambusiště $B_M = (I_{BM}, B_{spec}, IC)$ pro model M
- (1) Užitím binarizačního principu vytvoř z množin BS , ATR a CC množiny $BS' = BS \cup \{ \text{konkat. typy} \}$, ATR' obsahující informačně ekvivalentní HIT-atributy složitosti 2 a CC' obsahující přeformulovaná tvrzení konzistence z CC
- (2) Nechť I je prázdný instanční bambus a nechť T je typ, který pořadí všechny typy odpovídajícím prvkům z BS' a je identifikován $uwid_T$, pak pro každý prvek $t \in BS'$ vygeneruj $uwid_t$ a vlož dvojici $(uwid_T, uwid_t)$ do I

Algoritmus převodu DM dle HIT na přísušné CODS (2)

(3) Necht' $c_{ATR'} = \text{card}(ATR')$, $k \in \mathbb{N}$, $k = 1$,

$ATR'_k = ATR'$ a $B_k = \emptyset$, pak je-li $c_{ATR'} > 0$,

pokračuj krokem (a), jinak krokem (4)

(a) Necht' $A \in ATR'_k$ je libovolný HIT-atribut a

necht' je jednoho z následujících typů:

- $((\text{Wrld}, \text{Time}) \rightarrow (T_1 \rightarrow T_2))$
- $((\text{Wrld}, \text{Time}) \rightarrow (T_1 \rightarrow (T_2 \rightarrow \text{Bool})))$

Algoritmus převodu DM dle HIT na přísušné CODS (3)

(b) Na základě sémantiky a kardinality HIT-
atributu A vytvoř řádně specifikované sobě
navzájem odpovídající bambusy, a to bambus
tenký b_{thin} a bambus tlustý b_{thick}

(c) Vlož synapsi ($uwid_{T1}$, $uwid_{T2}$) do bambusu b_{thin}

(d) $B_{k+1} = B_k \cup \{ b_{\text{thin}}, b_{\text{thick}} \}$, $ATR'_{k+1} = ATR'_k \setminus \{ A \}$

(e) $k = k + 1$, je-li $k \leq c_{ATR'}$, pokračuj krokem (a)

(4) Polož $i_{\text{BM}} = i$ a $B_{\text{spec}} = B_k$, $IC = CC'$

Srovnání OODS a CODS (1)

- CODS oproti OODS ukládá data
 - duálně
 - CODS: ukládá primárně výskyty souvislostí mezi objekty a sekundárně objekty samotné – ovšem zase jen pomocí souvislostí
 - OODS: ukládá primárně agregované shluky souvislostí (objekty) a sekundárně potom vztahy mezi nimi – OODS pomocí agregovaných shluků konzervuje aktuální stav poznání světa a neumožňuje tuto konzervaci efektivně „rekonzervovat“

Srovnání OODS a CODS (2)

- CODS oproti OODS umožňuje
 - koncentrovaně ukládat hodnoty deskripcí
 - CODS: od jedné konkrétní hodnoty deskripce (časová známka, jméno, číselná hodnota) je možné lehce přejít ke všem objektům, kterých se tato hodnota deskripce nějakým způsobem týká – souvisí s nimi
 - OODS: tytéž hodnoty deskripcí (hodnoty vlastností objektů) jsou ukládány dislokovaně - příslušně ke konkrétnímu objektu; je proto např. netriviální zjistit, co všechno souvisí s daným datumem

Srovnání OODS a CODS (3)

- CODS oproti OODS umožňuje
 - snadno do sebe uložit svou strukturu
 - CODS: bambusy jsou kontejnery (typy) na pořádání synapsí, které v případě potřeby můžeme jejich zavedením jakožto instancí příslušného bambusu do bambusu instančního zmiňovat; obdobně můžeme zmiňovat i bambusy samotné
 - OODS: prakticky není možné
 - **Nuntá podmínka, aby CODS mohlo být úložištěm pro silný modelovací nástroj!**

Srovnání OODS a CODS (4)

- CODS oproti OODS dodržuje
 - princip uniformity a nezávislosti datových struktur
 - CODS: bambusy jakožto kontejnery na data jsou svou strukturou nezávislé na struktuře dat, které ukládají
 - OODS: struktury, ve kterých jsou data uložena kopírují strukturu samotných dat
 - **Nuntá podmínka, aby CODS mohlo být úložištěm pro silný modelovací nástroj!**

Princip uniformity a nezávislosti datových struktur

- Datové **struktury** používané pro ukládání dat oblasti zájmu vymezené logickým datovým modelem musí být na tomto logickém modelu svou strukturou **nezávislé**
- **Podoba** jednotlivých instancí datových struktur musí být **uniformní**

Univerzální CODS

- CODS jako úložiště silného modelovacího nástroje
- Dodržení principů
 - Pragmatické vnímání objektů
 - Princip univerzálního modelování a princip práce s neznámým
 - Princip jednoty modelujícího a modelovaného světa
 - Princip univerzality modelovacího nástroje

Implementační aspekty CODS

- Implementace univerzálního CODS
 - Projekt eTrium UIR, generace 5
 - Databázový stroj Caché
 - Viz <http://www.intersystems.com/cache/index.html>
 - Datová struktura globál
 - Modifikace B*-stromů
 - Viz DP OSK, strana 48-52