

Modely pre adaptívne a kolaboratívne vzdelávanie

Marián Šimko, Michal Barla, Pavel Michlík

▼ OBSAH

- ▶ Lisp
 - ▶ 1 Úvod - paradigmy programovania
 - ▶ 1.1 Procedurálne programovanie
 - ▶ 1.2 Objektovo-orientované programovanie
 - ▶ 1.3 Deklaratívne programovanie
 - ▶ 1.4 Aplikatívne programovanie
 - ▶ 1.5 Programovanie ohraničeniami
 - ▶ 1.6 Vizualne programovanie
 - ▶ 1.7 Paradigmy v súčasnosti
 - ▶ 2 Výrazy
 - ▶ 2.1 Výrazy a príkazy
 - ▶ 2.2 Vlastnosti čistých výrazov
 - ▶ 2.2.1 Funkcionálne programovanie
 - ▶ 2.3 Definícia funkcie
 - ▶ 2.3.1 Typy hodnôt vo funkciách
 - ▶ 2.3.2 Čiastočná funkcia
 - ▶ 2.4 Funkcionálny program
 - ▶ 3 Základné prvky jazyka Lisp
 - ▶ 3.1 Lisp
 - ▶ 3.2 Základné typy údajov
 - ▶ 3.3 Abstraktný typ údajov lisp-zoznam
 - ▶ 3.3.1 Funkcia FIRST

3.4.1 Apostrof a funkcia QUOTE

V predchádzajúcich častiach sme v príkladoch niekedy používali pri zápise zoznamu

zoznam

Popis:

Lisp-zoznam. Abstraktný dátový typ, základná dátová štruktúra jazyka Lisp. Definovaný ako postupnosť s-výrazov, čiže prvkom môže byť aj zoznam.

Príklady:

```
(1 2 3 4 5)
(A B 4 5 6 B G)
(1 2 (3 4) 5)
```

znak apostrof. Teraz je vhodný čas na vysvetlenie, prečo a kedy sa tento znak používa. Uvažujme napr. výraz (FIRST

FIRST

Syntax:

```
(FIRST list)
```

Popis:

prvý prvok zoznamu list

Príklady:

```
* (FIRST '(7 2 14))
7
* (FIRST '((hong kong) (ping pong) bang))
(hong kong)
(REST (A B C))
```

vyhodnotením ktorého chceme získať druhý prvok zoznamu (A B C). Inými slovami,

▶ funkcia

funkcia

Popis:

Obsah

- Lisp
 - 1 Úvod - paradigmy programovania
 - 1.1 Procedurálne programovanie
 - 1.2 Objektovo-orientované programovanie
 - 1.3 Deklaratívne programovanie
 - 1.4 Aplikatívne programovanie
 - 1.5 Programovanie ohraňovaním
 - 1.6 Vizualne programovanie
 - 1.7 Paradigmy v súčasnosti
 - 2 Výrazy
 - 2.1 Výrazy a príkazy
 - 2.2 Vlastnosti čistých výrazov
 - 2.2.1 Funkcionálne programovanie
 - 2.3 Definícia funkcie
 - 2.3.1 Typy hodnôt vo funkciách
 - 2.3.2 Čiastočná funkcia
 - 2.4 Funkcionálny program
 - 3 Základné prvky jazyka Lisp
 - 3.1 Lisp
 - 3.2 Základné typy údajov
 - 3.3 Abstraktný typ údajov lisp-zoznam
 - 3.3.1 Funkcia FIRST

3.4.1 Apostrof a funkcia QUOTE

V predchádzajúcich častiach sme v príkladoch niekedy používali pri zápise zoznamu

zoznam

Popis:

Lisp-zoznam. Abstraktný dátový typ, základná dátová štruktúra jazyka Lisp. Definovaný ako postupnosť s-výrazov, čiže prvkom môže byť aj zoznam.

Príklady:

```
(1 2 3 4 5)
(A B 4 5 6 B G)
(1 2 (3 4) 5)
```

znak apostrof. Teraz je vhodný čas na vysvetlenie, prečo a kedy sa tento znak používa. Uvažujme napr. výraz (FIRST

FIRST

Syntax:

```
(FIRST list)
```

Popis:

prvý prvok zoznamu list

Príklady:

```
* (FIRST '(7 2 14))
7
* (FIRST '((hong kong) (ping pong) bang))
(hong kong)
(REST (A B C))
```

výhodnotením ktorého chceme získať druhý prvok zoznamu (A B C). Inými slovami,

▸ funkcia

funkcia

Popis:

▼ **OBSAH**

- ▶ Lisp
 - ▶ 1 Úvod - paradigmy programovania
 - ▶ 1.1 Procedurálne programovanie
 - ▶ 1.2 Objektovo-orientované programovanie
 - ▶ 1.3 Deklaratívne programovanie
 - ▶ 1.4 Aplikatívne programovanie
 - ▶ 1.5 Programovanie ohraničeniami
 - ▶ 1.6 Vizualné programovanie
 - ▶ 1.7 Paradigmy v súčasnosti
 - ▶ 2 Výrazy
 - ▶ 2.1 Výrazy a príkazy
 - ▶ 2.2 Vlastnosti čistých výrazov
 - ▶ 2.2.1 Funkcionálne programovanie
 - ▶ 2.3 Definícia funkcie
 - ▶ 2.3.1 Typy hodnôt vo funkciách
 - ▶ 2.3.2 Čiastočná funkcia
 - ▶ 2.4 Funkcionálny program
 - ▶ 3 Základné prvky jazyka Lisp
 - ▶ 3.1 Lisp
 - ▶ 3.2 Základné typy údajov
 - ▶ 3.3 Abstraktný typ údajov lisp-zoznam
 - ▶ 3.3.1 Funkcia FIRST

3.3.1 Apostrof a funkcia QUOTE

V predchádzajúcich častiach sme v príkladoch niekedy používali pri zápise zoznamu

zoznam

Popis:

lisp-zoznam. Abstraktný dátový typ, základná dátová štruktúra jazyka Lisp. Definovaný ako postupnosť s-
výrazov, čiže prvkom môže byť aj zoznam.

Príklady:

(1 2 3 4 5)

(A B 4 5 6 B G)

(1 2 (3 4) 5)

znak apostrof. Teraz je vhodný čas na vysvetlenie, prečo a kedy sa tento znak používa. Uvažujme napr. výraz

(FIRST

FIRST

Syntax:

(FIRST list)

Popis:

prvý prvok zoznamu list

Príklady:

(FIRST '(7 2 14))

?

(FIRST '((hong kong) (ping pong) bang))

(hong kong)

(REST (A B C))

výhodnotením ktorého chceme získať druhý prvok zoznamu (A B C). Inými slovami,

funkcia

funkcia

Popis:

▼ OBSAH

- ▶ Lisp
 - ▶ 1 Úvod - paradigmy programovania
 - ▶ 1.1 Procedurálne programovanie
 - ▶ 1.2 Objektovo-orientované programovanie
 - ▶ 1.3 Deklaratívne programovanie
 - ▶ 1.4 Aplikatívne programovanie
 - ▶ 1.5 Programovanie ohraničeniami
 - ▶ 1.6 Vizualne programovanie
 - ▶ 1.7 Paradigmy v súčasnosti
 - ▶ 2 Výrazy
 - ▶ 2.1 Výrazy a príkazy
 - ▶ 2.2 Vlastnosti čistých výrazov
 - ▶ 2.2.1 Funkcionálne programovanie
 - ▶ 2.3 Definícia funkcie
 - ▶ 2.3.1 Typy hodnôt vo funkciách
 - ▶ 2.3.2 Čiastočná funkcia
 - ▶ 2.4 Funkcionálny program
 - ▶ 3 Základné prvky jazyka Lisp
 - ▶ 3.1 Lisp
 - ▶ 3.2 Základné typy údajov
 - ▶ 3.3 Abstraktný typ údajov lisp-zoznam
 - ▶ 3.3.1 Funkcia FIRST

3.4.1 Apostrof a funkcia QUOTE

V predchádzajúcich častiach sme v príkladoch niekedy používali pri zápise zoznamu

zoznam

Popis:

Lisp-zoznam. Abstraktný dátový typ, základná dátová štruktúra jazyka Lisp. Definovaný ako postupnosť s-výrazov, čiže prvkom môže byť aj zoznam.

Príklady:

```
(1 2 3 4 5)
(A B 4 5 6 B G)
(1 2 (3 4) 5)
```

znak apostrof. Teraz je vhodný čas na vysvetlenie, prečo a kedy sa tento znak používa. Uvažujme napr. výraz (FIRST

FIRST

Syntax:

```
(FIRST list)
```

Popis:

prvý prvok zoznamu list

Príklady:

```
* (FIRST '(7 2 14))
7
* (FIRST '((hong kong) (ping pong) bang))
(hong kong)
(REST (A B C))
```

vyhodnotením ktorého chceme získať druhý prvok zoznamu (A B C). Inými slovami,

- ▶ funkcia

funkcia

Popis:

OBSAH

- Lisp
 - 1 Úvod - paradigmy
 - 1.1 Procedurálne
 - 1.2 Objektovo-ori
 - 1.3 Deklaratívne
 - 1.4 Aplikatívne p
 - 1.5 Programova
 - 1.6 Vizualne programovanie
 - 1.7 Paradigmy v súčasnosti
 - 2 Výrazy
 - 2.1 Výrazy a príkazy
 - 2.2 Vlastnosti čistých výrazov
 - 2.2.1 Funkcionálne programovanie
 - 2.3 Definícia funkcie
 - 2.3.1 Typy hodnôt vo funkciách
 - 2.3.2 Čiastočná funkcia
 - 2.4 Funkcionálny program
 - 3 Základné prvky jazyka Lisp
 - 3.1 Lisp
 - 3.2 Základné typy údajov
 - 3.3 Abstraktný typ údajov lisp-zoznam
 - 3.3.1 Funkcia FIRST

QUOTE

Syntax:

(QUOTE arg)

Popis:

Vráti nevyhodnotený argument arg. Skrátenejší zápis

QUOTE je '

Príklady:

```
* (QUOTE (A B C))  
(A B C)
```

v príkladoch niekedy používali pri zápise zoznamu

zoznam

typ, základná dátová štruktúra jazyka Lisp. Definovaný ako postupnosť s-zoznam.

```
(A B 4 5 6 B G)  
(1 2 (3 4) 5)
```

znak apostrof. Teraz je vhodný čas na vysvetlenie, prečo a kedy sa tento znak používa. Uvažujme napr. výraz (FIRST

FIRST

Syntax:

(FIRST list)

Popis:

prvý prvok zoznamu list

Príklady:

```
* (FIRST '(7 2 14))  
7  
* (FIRST '((hong kong) (ping pong) bang))  
(hong kong)  
(REST (A B C))
```

vyhodnotením ktorého chceme získať druhý prvok zoznamu (A B C). Inými slovami,

▶ funkcia

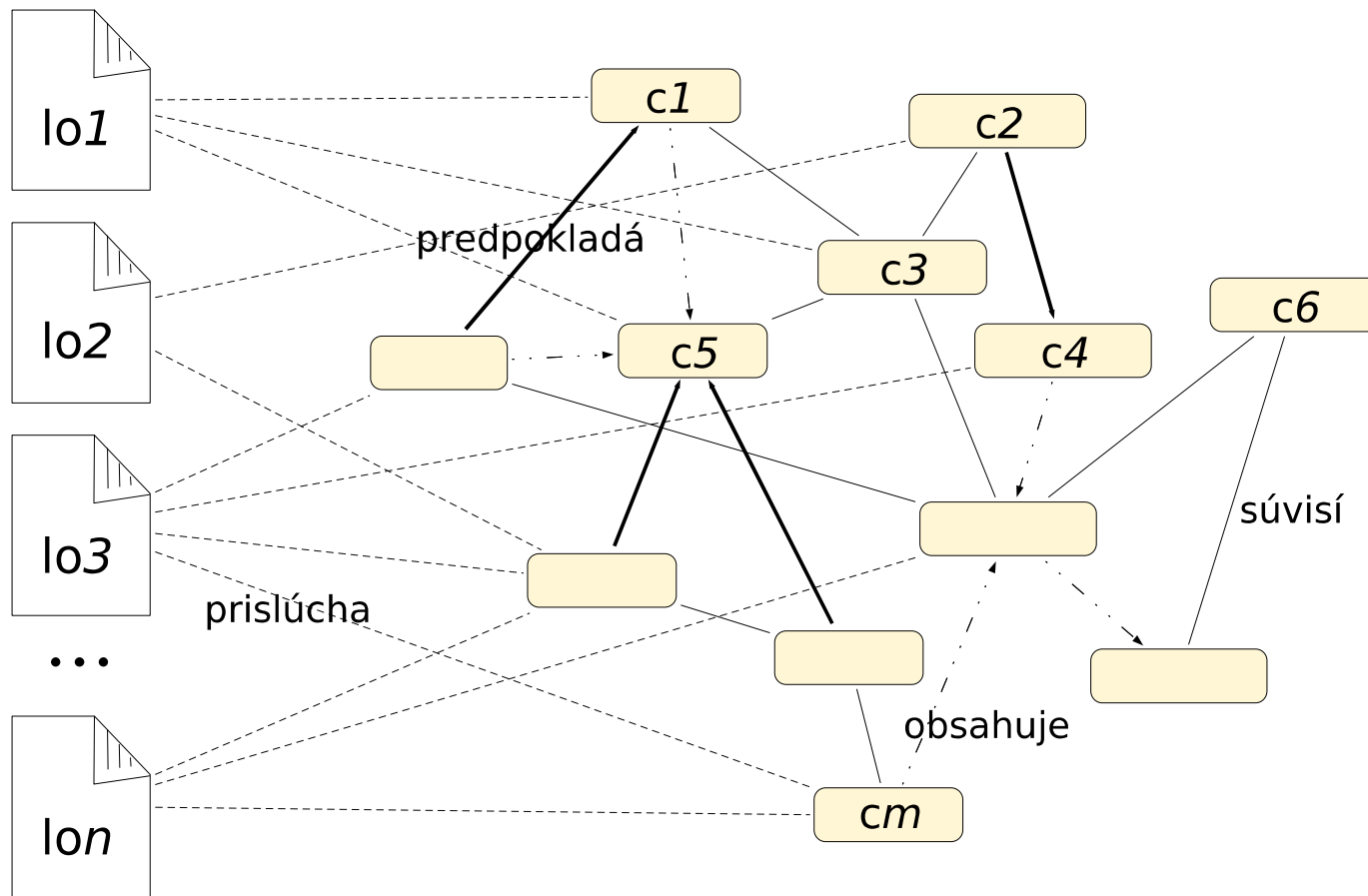
funkcia

Popis:

Prípady použitia

- štúdium materiálov
 - adaptívne – s prispôsobovaním
 - interaktívne – so skúšaním
 - kolaboratívne – s ostatnými používateľmi
 - pohodlne – s možnosťou anotovať

Domain modeling

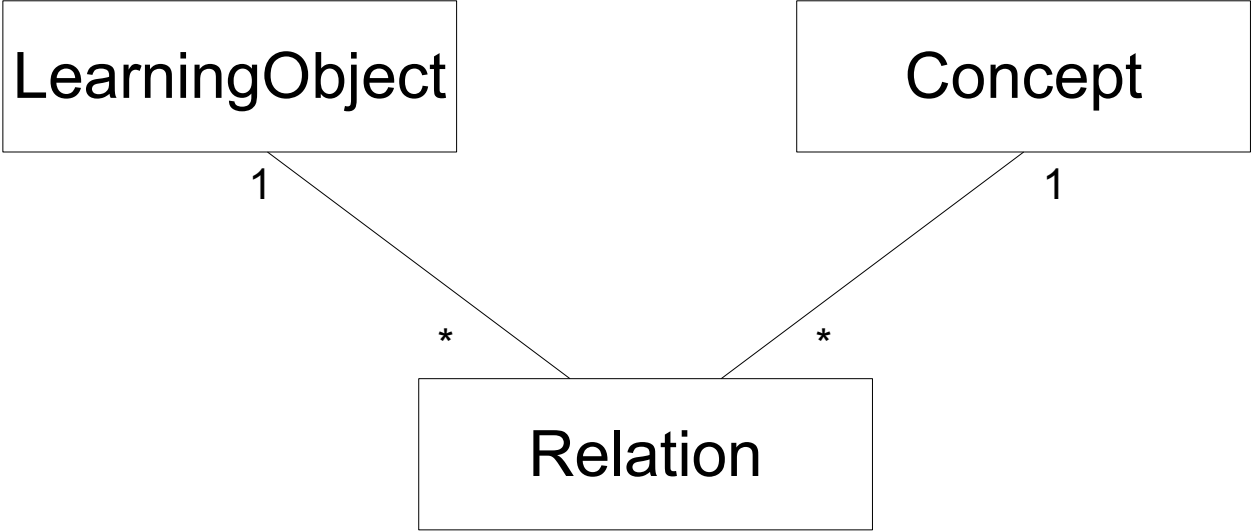


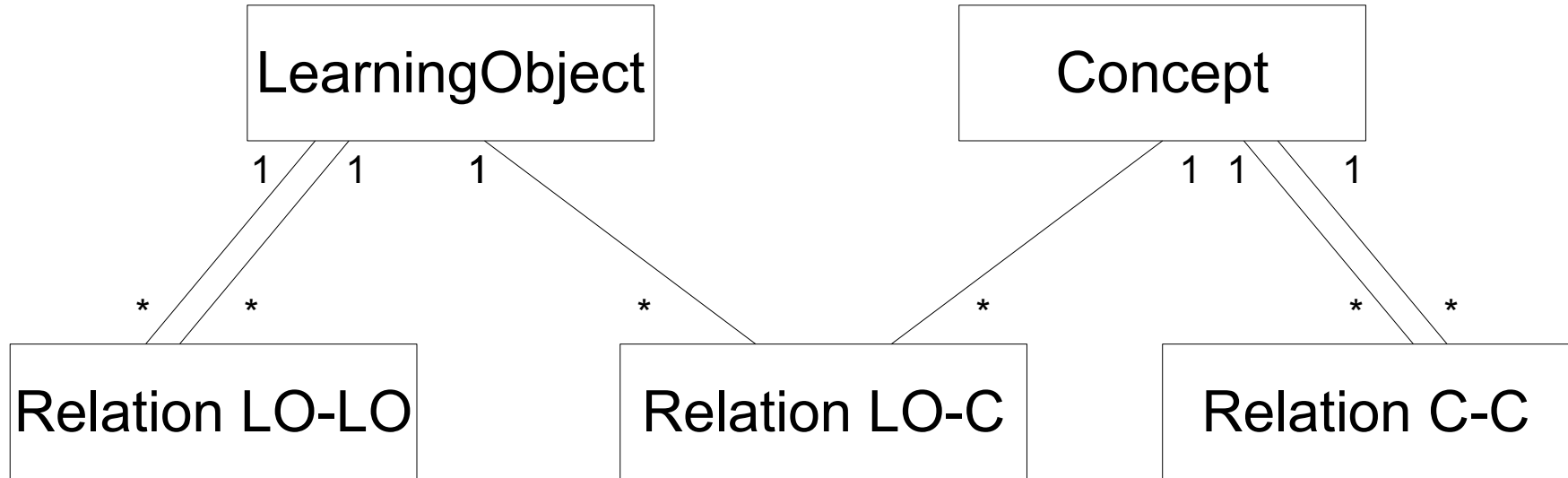
priestor výučbových objektov

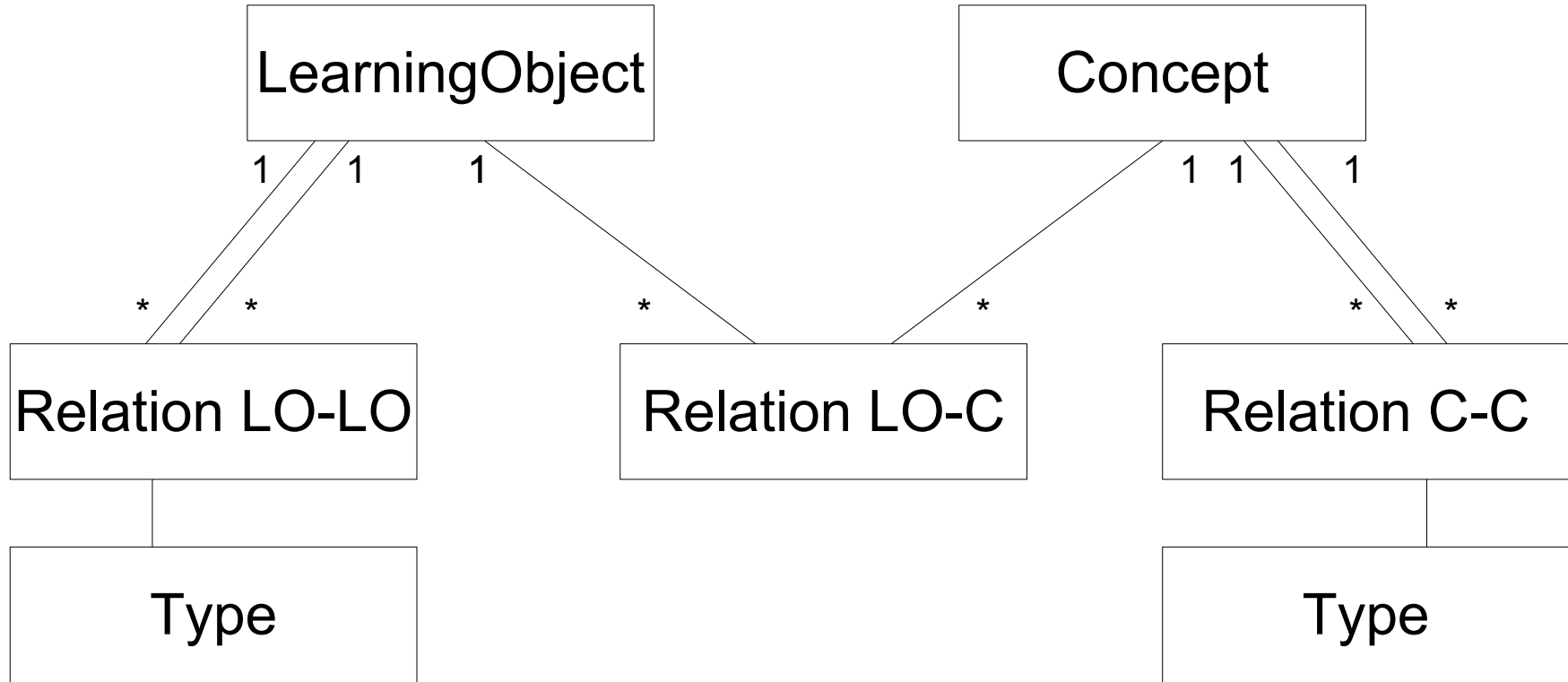
priestor konceptov

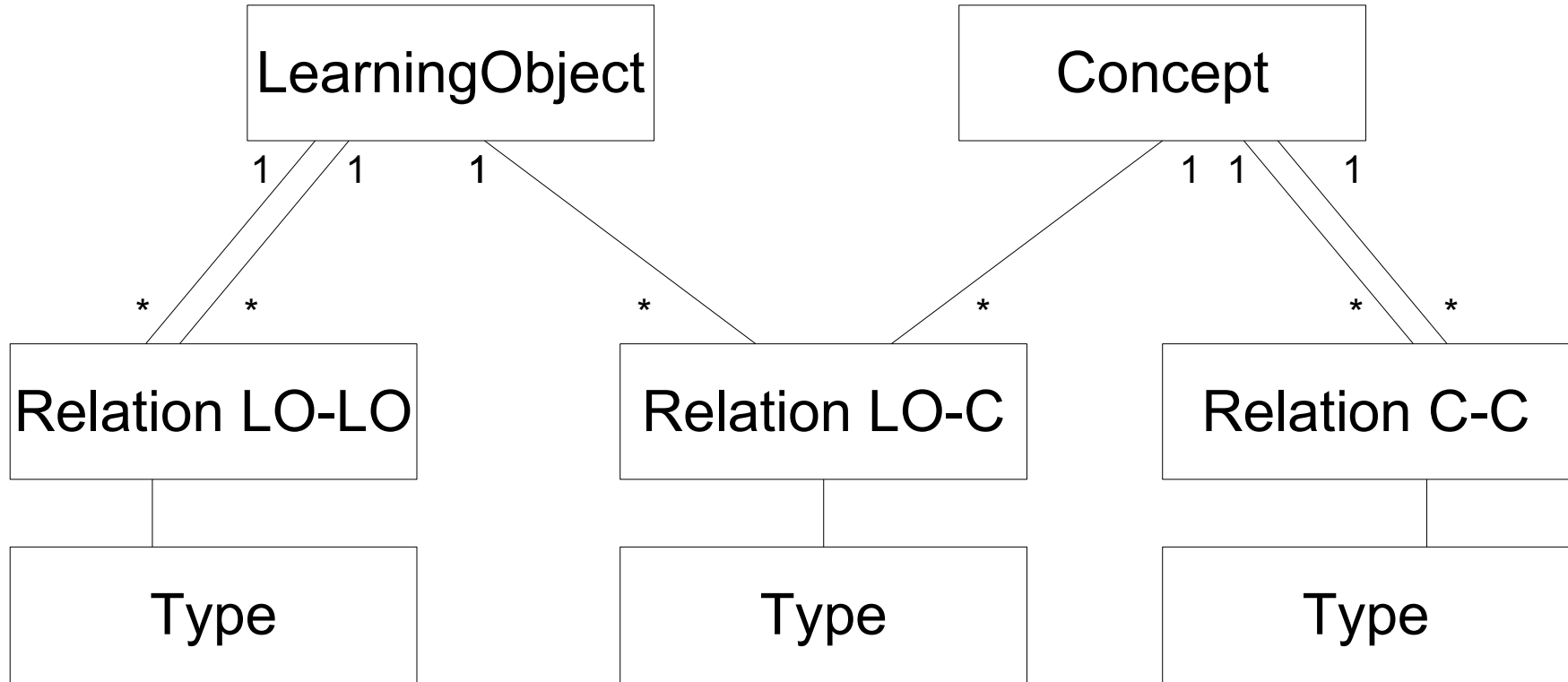
LearningObject

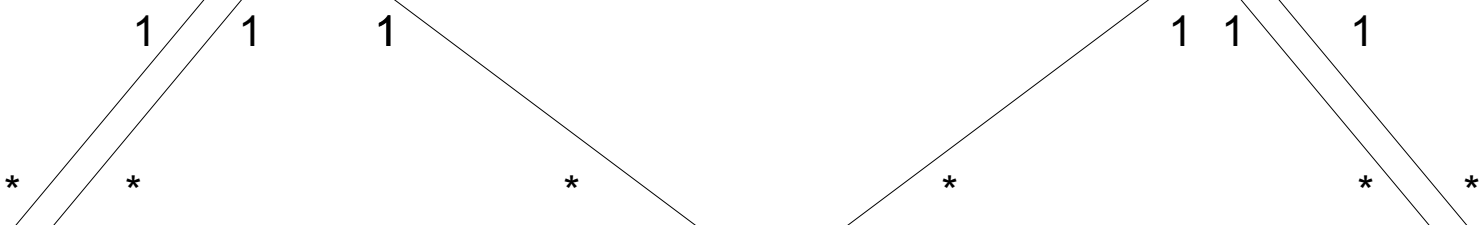
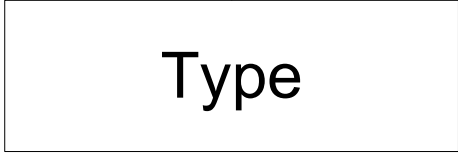
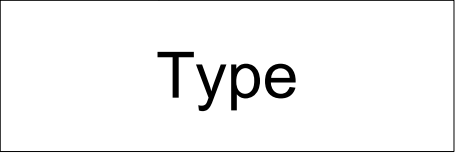
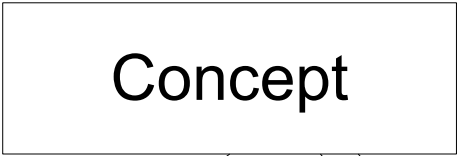
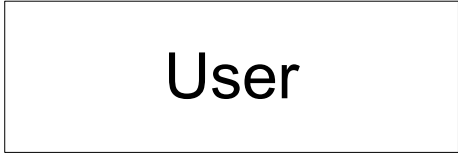
Concept

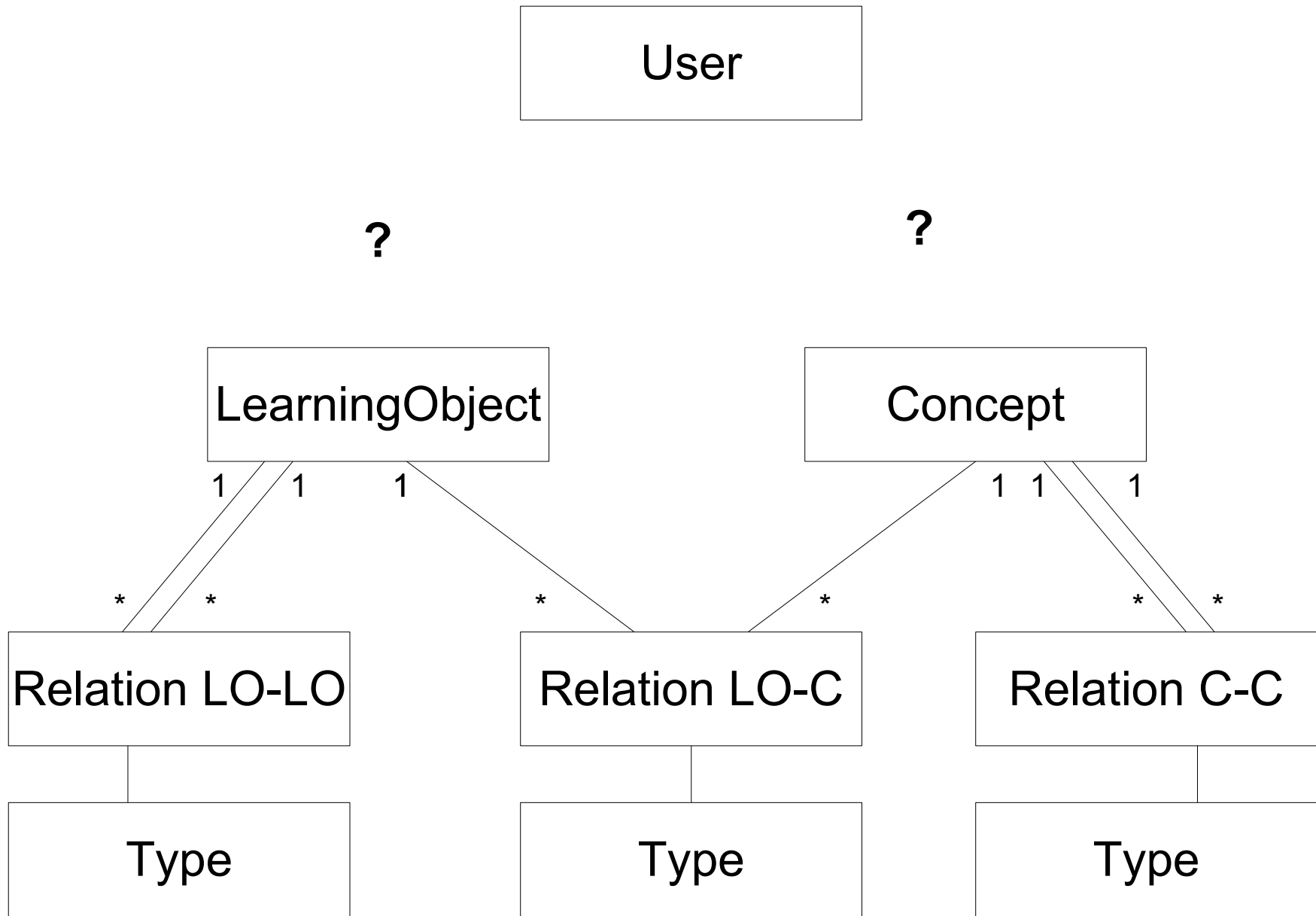












User

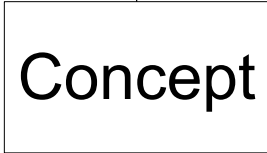
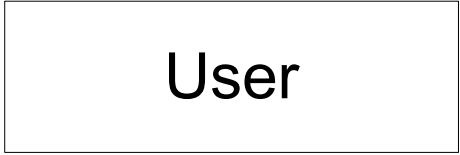
LearningObject

Concept

User

Content

Metadata

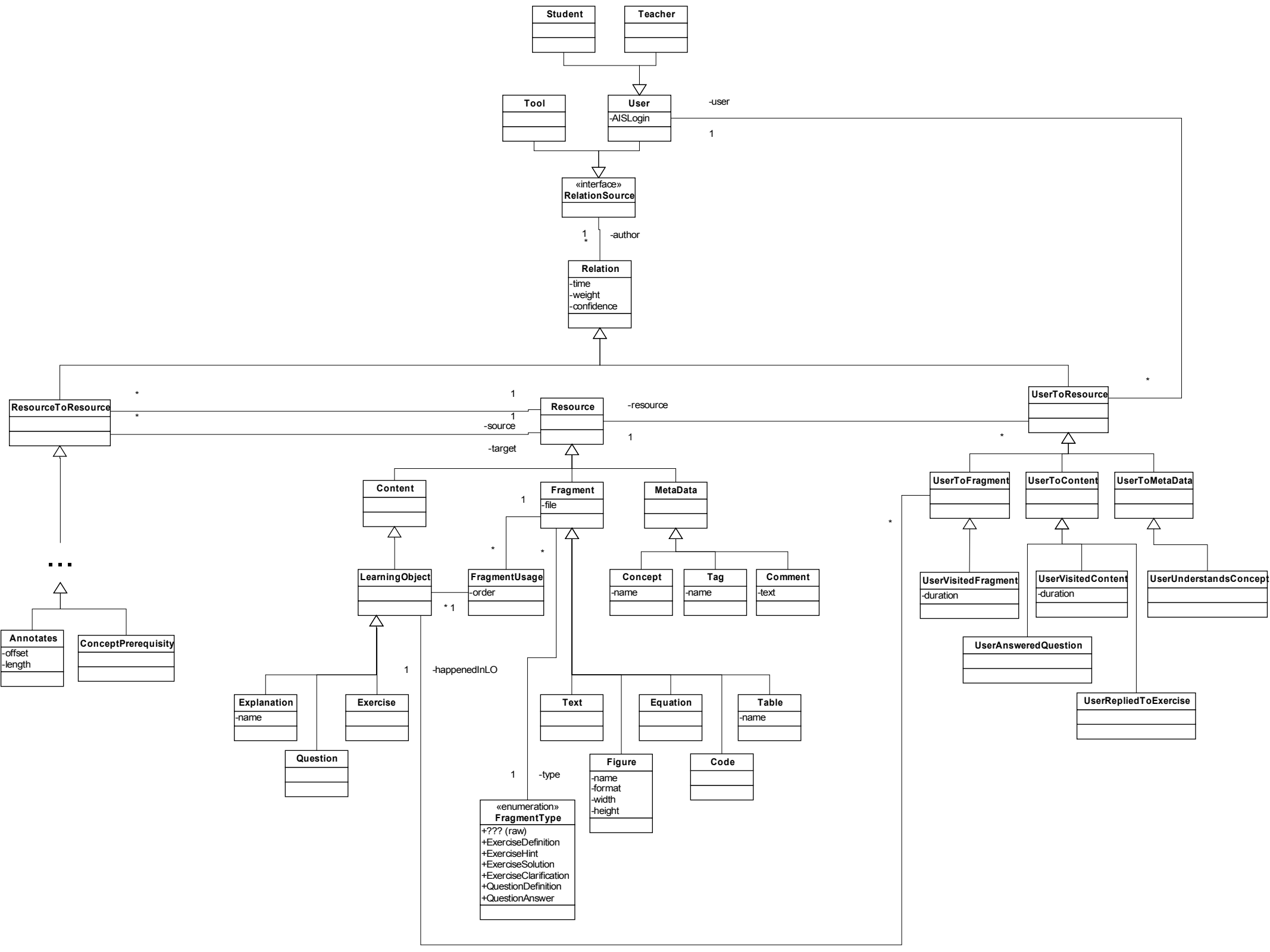


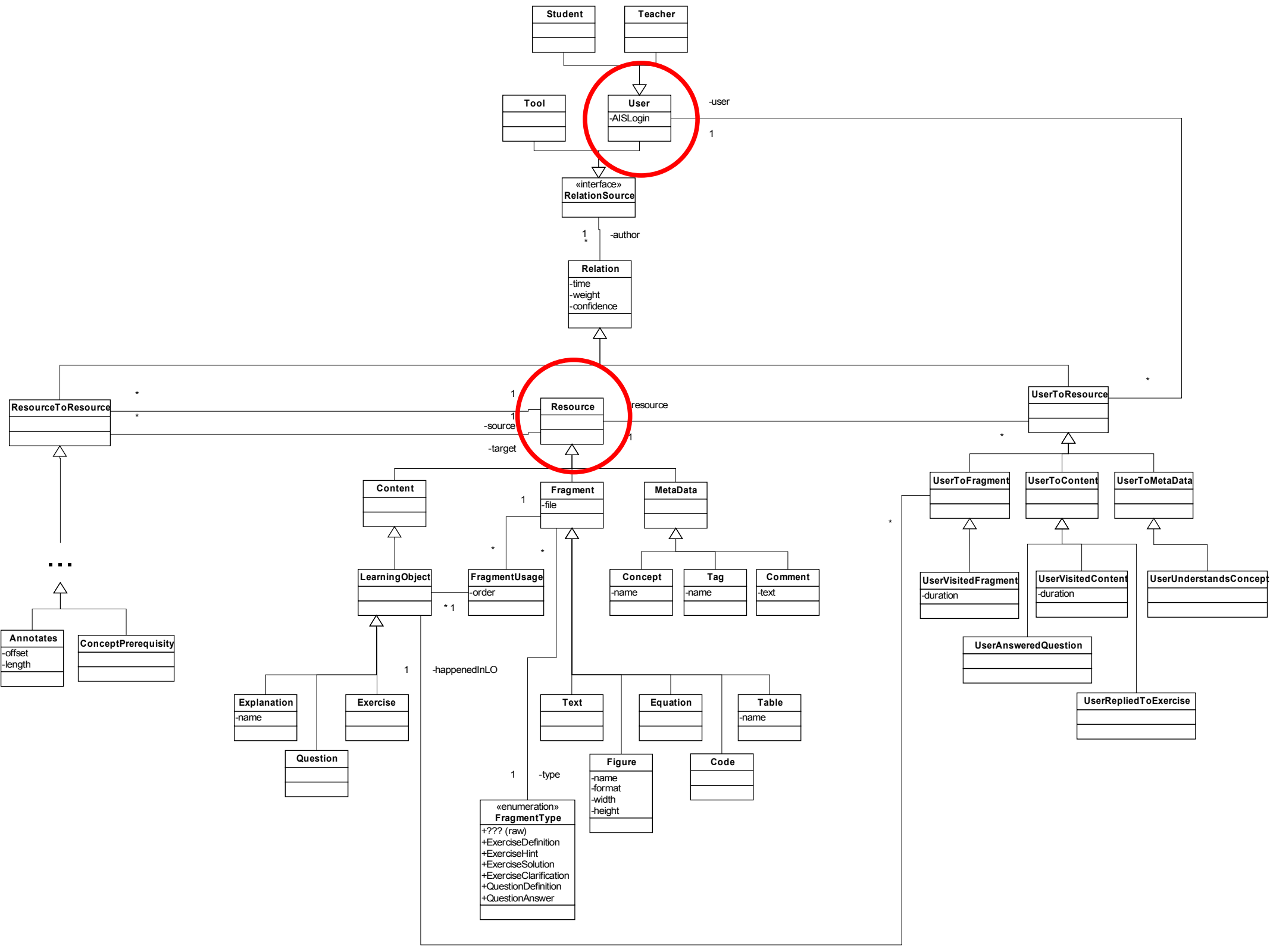
User

Content + Metadata

User

Resource



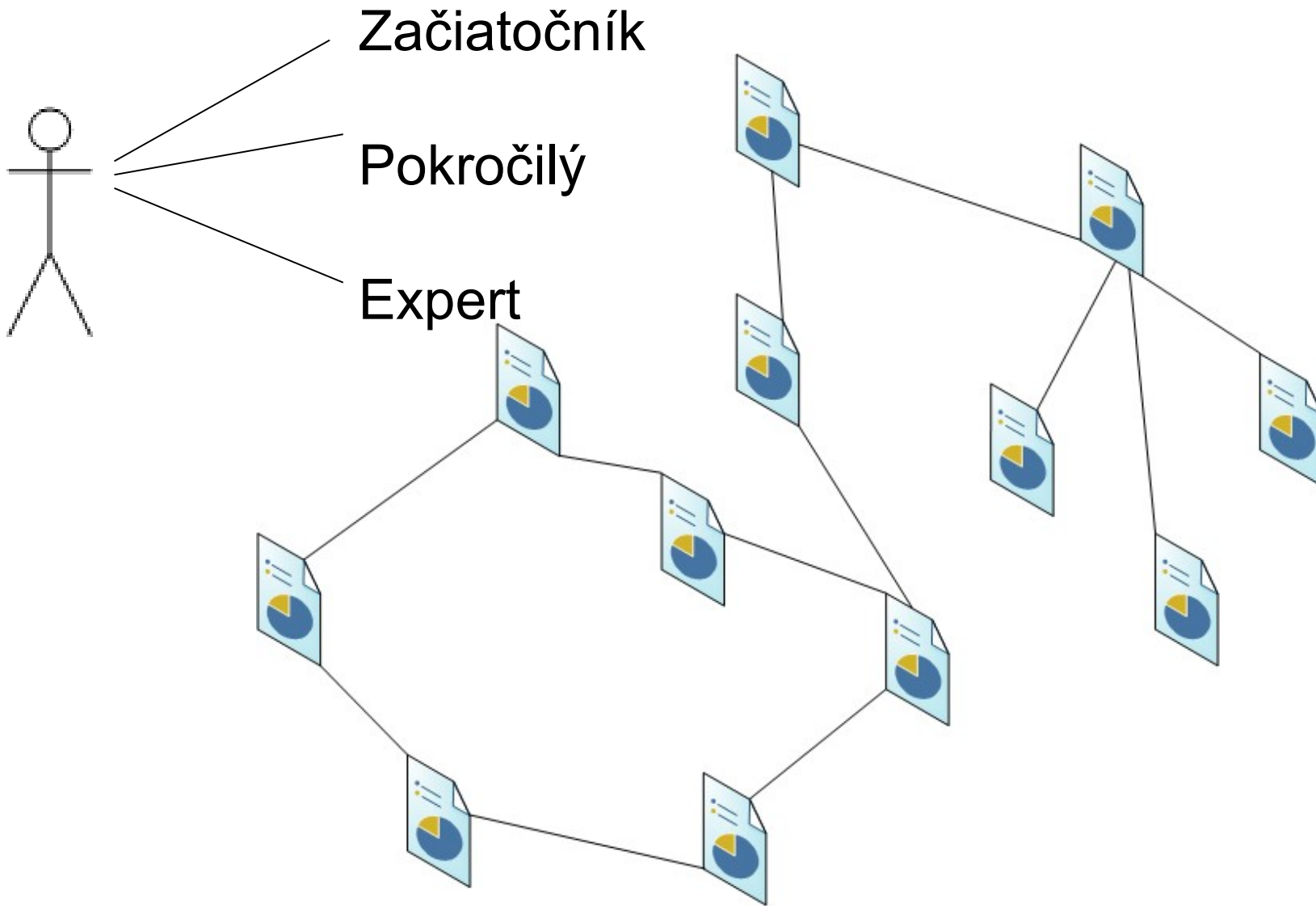


User modeling

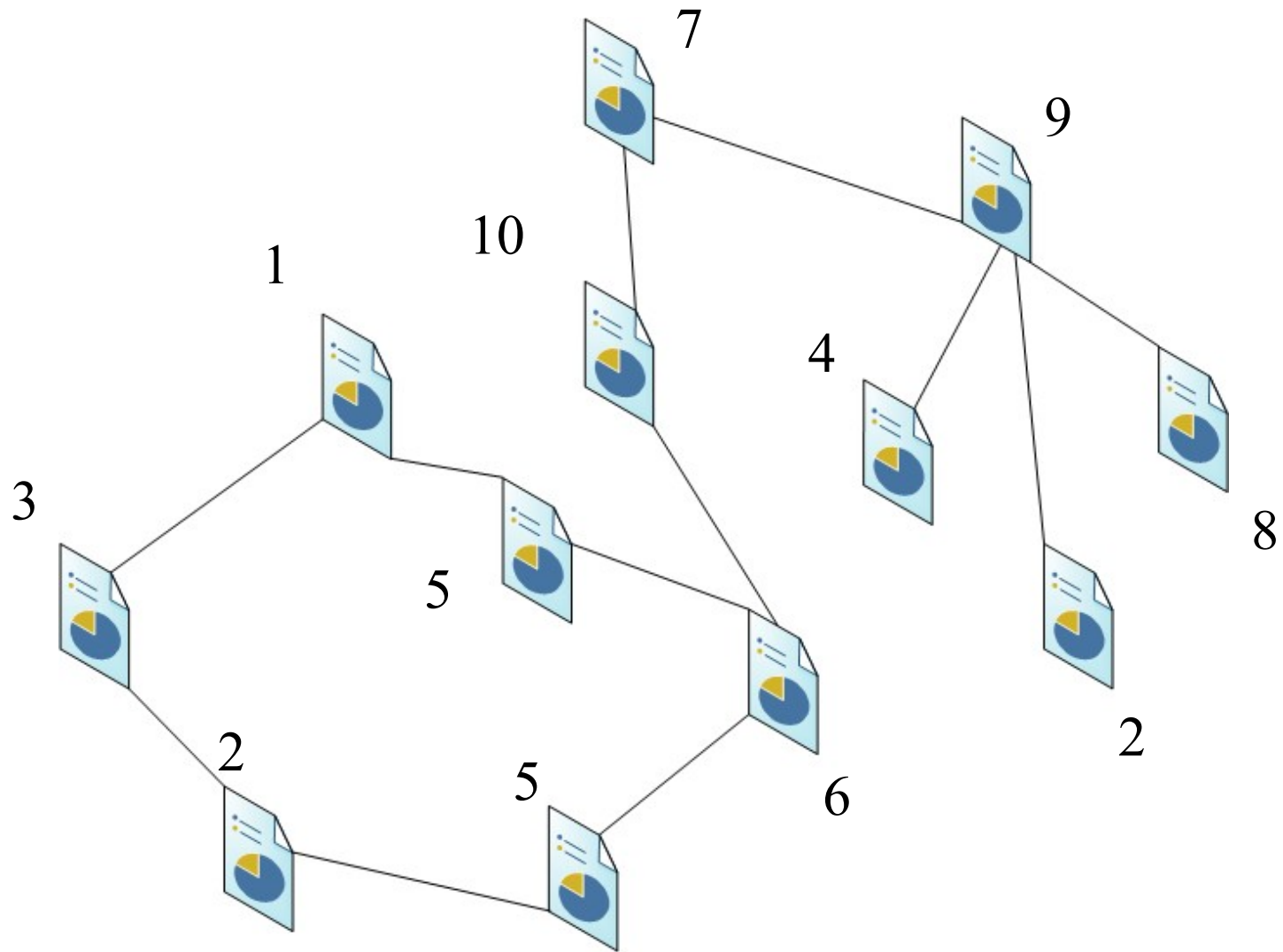
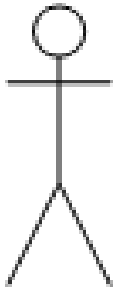
Čo modelovať/čomu sa prispôsobovať

- Preferencie
- Vedomosti
- Ciele
- Predchádzajúca navigácia
- Pozadie, profesia
schopnosti
- Platforma, šírka pásma
- ...

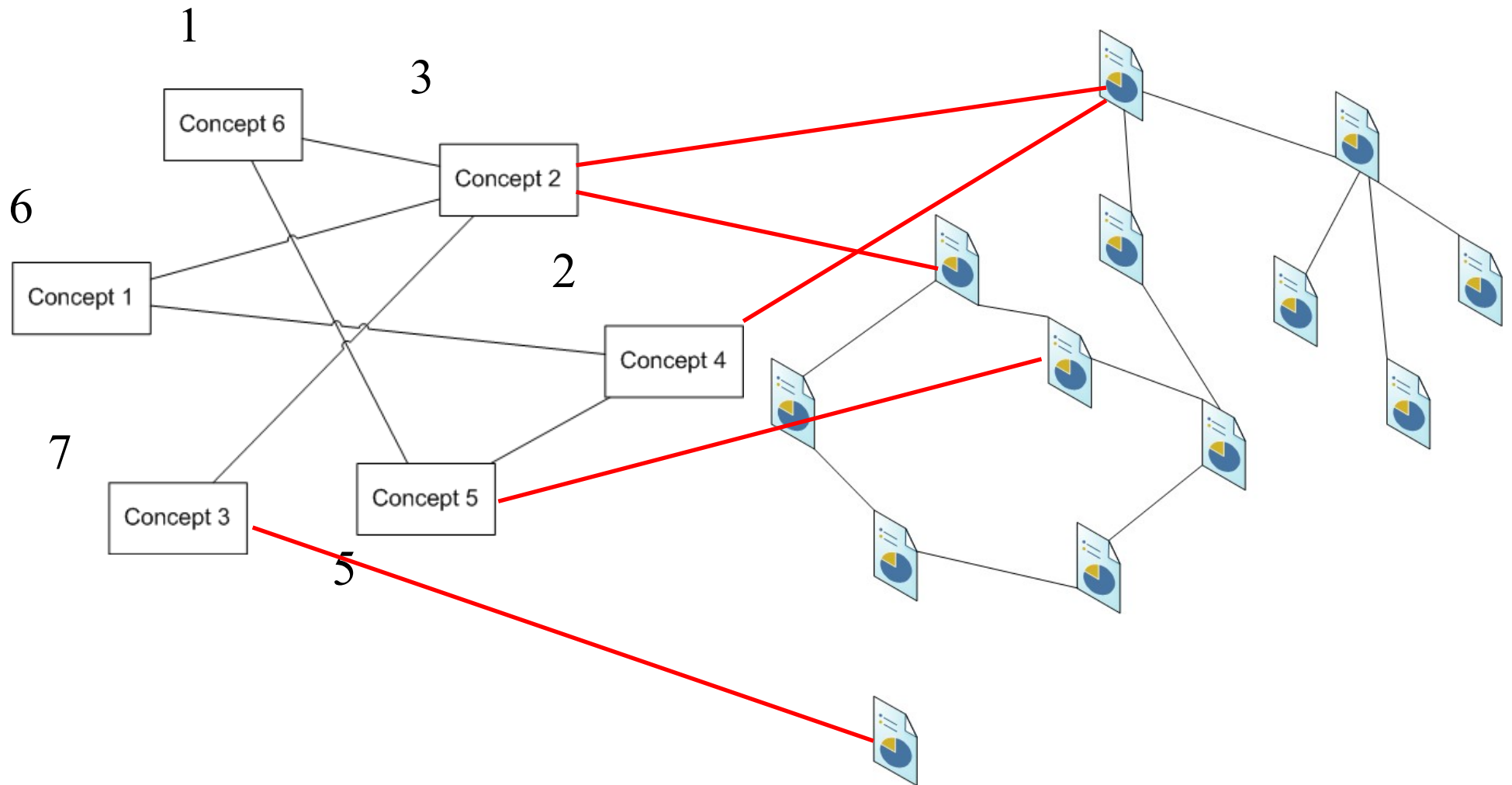
Stereotypy



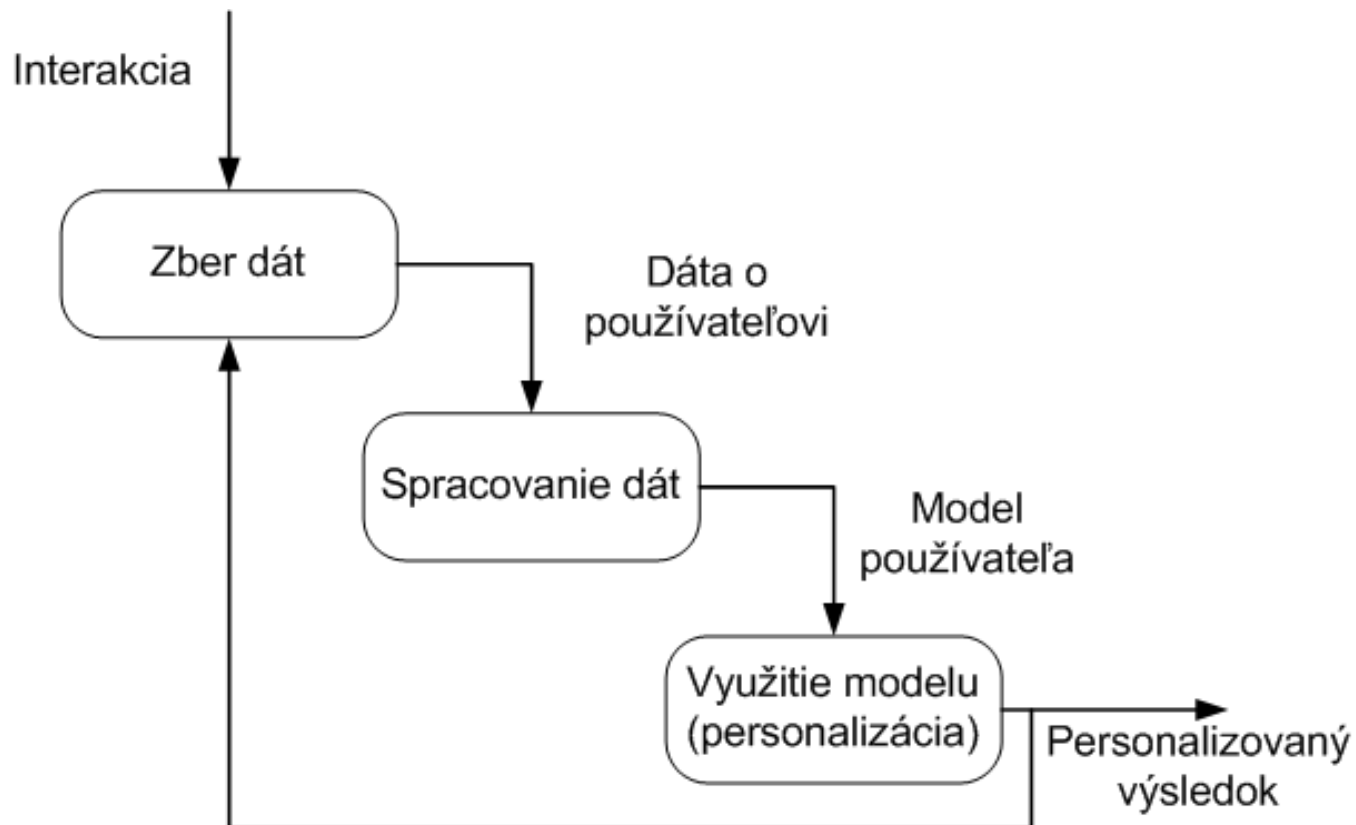
Prekryvný model



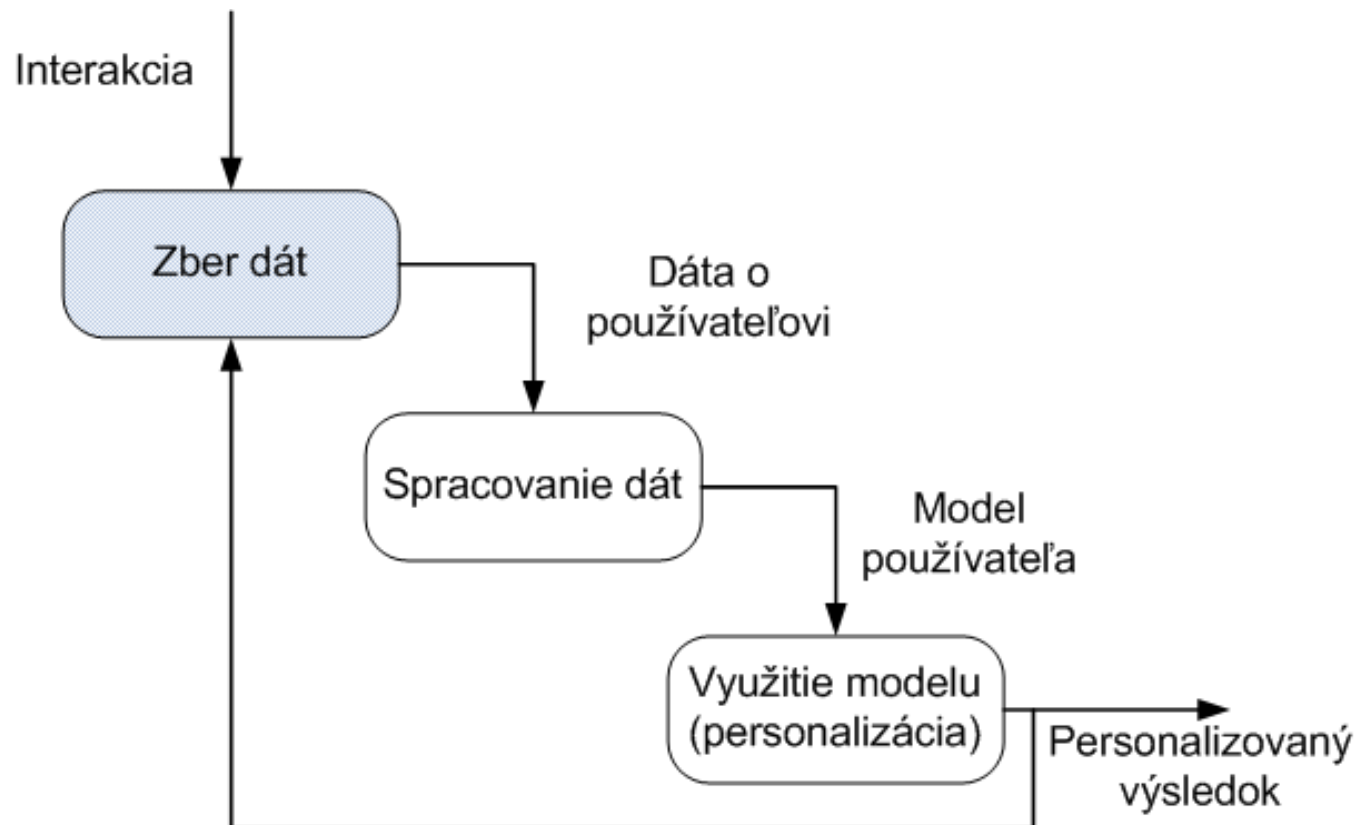
Otvorený priestor



Proces v adaptívnych systémoch



Zber dát



Zber dát – dotazník

- tzv. *self-described*
- dotazníky, vstupné testy...
 - toto používateľ nechce
 - stráca koncentráciu
 - je znechutený
- často používané
 - objektívne?

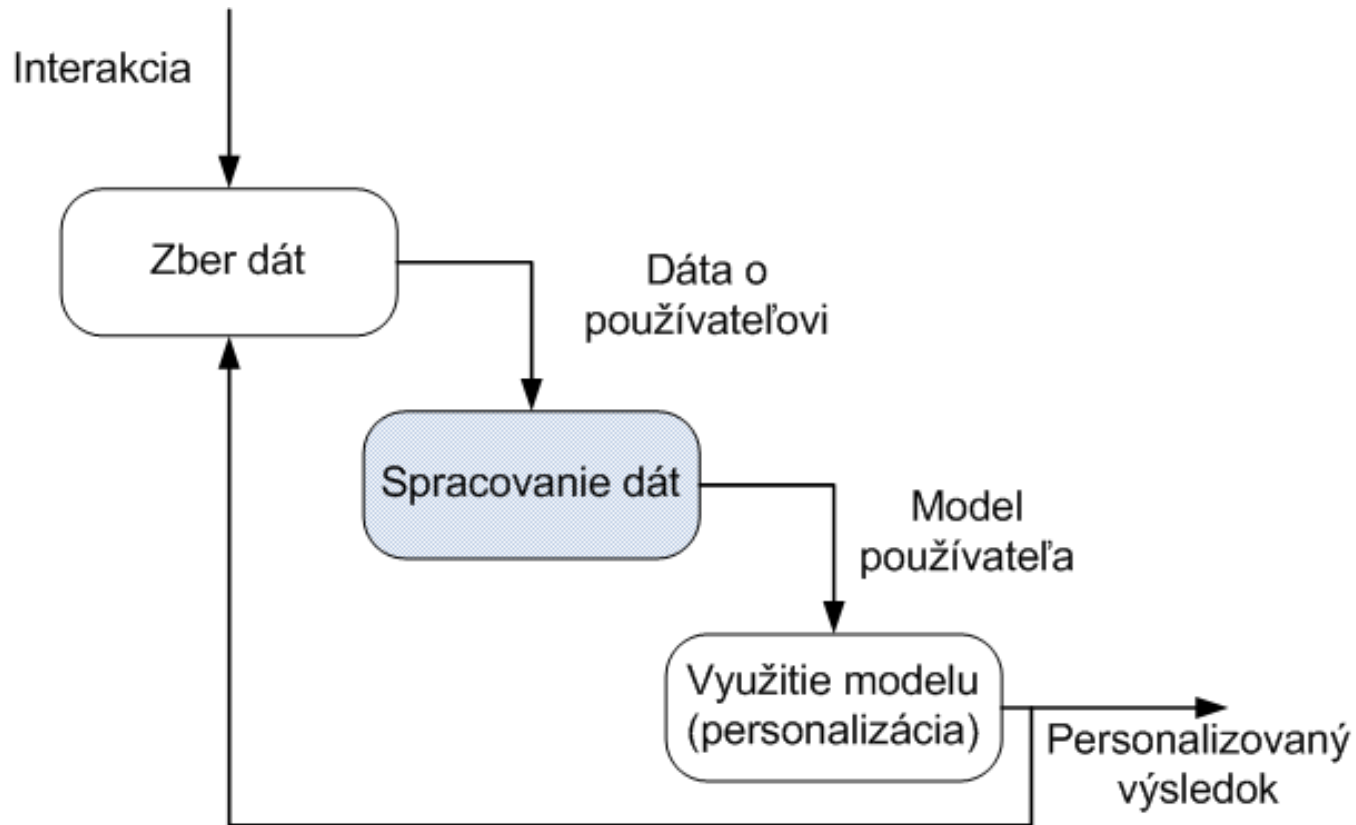
Zber dát – Spätná väzba

- more like this, not for me
- “už som to pochopil”
- explicitná vs. implicitná
 - Vidím, že hodnotím
 - Používam systém a pritom vlastne aj hodnotím
- čo s hodnotením?
 - Collaborative Filtering (systémy s odporúčaním)

Zber dát - logy

- Štandardné logy web servera
 - Úroveň HTTP protokolu
 - Identifikácia session?
 - Interpretácia?
- Vlastné logovacie riešenie (klient vs. server)
 - Standalone desktop aplikácia (keylogger)
 - Súkromie?
 - JavaScript, JavaApplet
 - Logovanie súčasťou aplikačnej logiky – UM servre

Spracovanie dát



Analýza dát

- Čo si môžeme dovoliť záleží od toho
 - aké zbierame dáta a
 - ako je reprezentovaný náš model
- niekedy je interpretácia jasná – formulár
- inokedy sa treba s dátami “pohrat”
 - dolovanie v dátach, strojové učenie, umelá intel.
 - štatistické modely (napr. Bayesovské siete)
 - pravidlové prístupy
 - ...

Reprezentácia modelu používateľa

- Čo od modelu očakávame
- Ako ho vytvárame
- Ako reprezentujeme doménu

- In-memory (binárne, textovo – vlastné XML)
- Vektor
- Relačná databáza
- RDF(S)/OWL

Zdieľanie modelu

- In-memory (binárne, textovo – vlastné XML)
 - Veľmi ťažko zdieľateľné
- Vektor
 - Väčšinou silne viazané na inštancie domény
- Relačná databáza, RDF(S)/OWL
 - Podľa toho, ako „šikovne“ je navrhnutá schéma

Reprezentácia modelu

- Explicitné typy charakteristík
 - Preferencia doménového atribútu
 - Znalosť (vedomosť) konceptu
 - Ohodnotenie nejakej doménovej položky (obsahu)
- Atribúty charakteristík
- Metadáta charakteristík
 - časová pečiatka, zdroj (vrstva, pohľad), relevancia, dôveryhodnosť, ...

Zusammenfassung

Čo zatiaľ uvažujeme

- Content resource
 - Learning Object
 - Comment
 - Blog
- Fragments
 - Definition, Hint, Solution, Clarification
 - Text, Equation, Code, Table, Figure

Čo zatiaľ uvažujeme

- Metadata resource
 - Concept
 - Tag
 - Comment
 - Discussion
- User
 - Student
 - Teacher

Čo zatiaľ uvažujeme

- Resource-to-Resource relations
 - is-a, part-of, prerequisite, ...
- Resource-to-User relations
 - user modeling
 - logging
 - authoring

Čo ďalej...

- „Proof of concept“
 - webové služby
- Voľba technológie
 - Ruby on Rails
 - Spring
 - SilverLight
- Pomoc vítaná...

Literatúra

- Brusilovsky, P., Kobsa A., Nejdl, W. (Eds.):
The Adaptive Web, Methods and Strategies of
Web Personalization, LNCS 4321, 2007
- T. Joachims et al: *Evaluating the Accuracy of
Implicit Feedback from Clicks and Query
Reformulations in Web Search*, ACM
Transactions on Information Systems (TOIS),
Vol. 25, No. 2 (April), 2007.