

Analytic Pattern (Analytické vzory)

Introduction (Úvod)

Juraj Petráš, Žilina, 2010

e-mail: j.petras@petrasoft.sk



Agenda

1. Motivácia
2. Definícia
3. Ukážky vzorov analýzy



1

Motivácia



- Prečo je dobré poznať analytické vzory?
 - Dokumentujú znalostí z jednotlivých problémových oblastí (finančníctvo, zdravotníctvo, atď.)
 - Ušetria čas
 - Zvyšujú kvalitu
 - Zlepšujú porozumenie medzi analytikmi a designermi

História



- Inšpirovaný prácou Christophera Alexandera, rakúskeho Američana, ktorý sa 1977-79 zaslúžil o zavedenie Návrhových vzorov a jazyka v oblasti architektúry (nie softvérovej ale tej, čo navrhuje budovy)
- Kent Beck and Ward Cunningham začali experimentovať s aplikáciou týchto myšlienok na tvorbu softvéru v roku 1987
- Návrhové vzory získali popularitu po uvedení knihy „Elements of Reusable Object-Oriented Software“ v 1994 skupinou "Gang of Four" ([Gamma, Erich](#); [Richard Helm](#), [Ralph Johnson](#), and [John Vlissides](#)) na konferencii OOPSLA v Portlande
- V tom istom roku bola aj prvá konferencia Pattern Languages of Programming Conference a založené Portland Pattern Repository
- Martin Fowler: Analytic Pattern: Reusable object models, 1998



Definícia

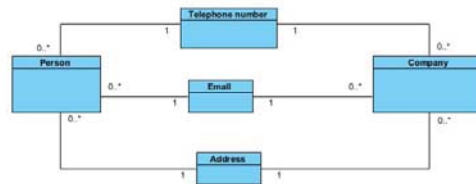
- Čo je to analytický vzor?
- Mark Fowler hovorí o 2 kategóriách vzorov:
 - Analytické vzory sú skupiny konceptov, ktoré reprezentujú a všeobecnú konštrukciu v modelovaní problémovej (business) domény
 - Podporné vzory popisujú ako aplikovať analytické vzory
- Analytický vzor je začiatok hľadania, nie konečný cieľ analýzy.
- Analytický vzor nie je hotový návrh, ktorý môže byť jednoducho transformovaný do kódu.
- Doménou vzorov analýzy sú typy/interfaces (nie triedy/classes) a relácie medzi nimi .

Accountability Sledovateľnosť/Zodpovednosť



- Analytický vzorec modelujúci mnohé problémové oblasti, kde osoba alebo organizácia je nejakou zodpovednosťou voči inej osobe alebo organizácii
- Party supertyp zovšeobecňujúci osobu alebo organizáciu
- Organizational Hierarchy
- Organizational Structure
- Party + Organizational Structure = Accountability
- Operating Scope: definuje sféru zodpovednosti
- Post: často-presúvajúce zodpovednosti môžeme zoskupiť a priradiť pozíciu miesta osoby alebo organizácii

Person & Company

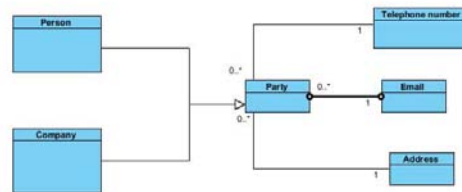


Analysis Patterns

7

Typický príklad: Adresár, ktorý každý má v svojom mobile

Party



Analysis Patterns

8

Party: meno pre koncept, ktorý každý pozná, ale na ktorom mene sa naša komunita nevie zhodnúť

Jednoduchá organizačná hierarchia

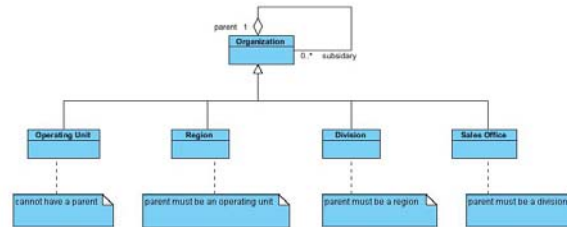


Analysis Patterns

9

- 4-úrovňová organizačná štruktúra
- 1 hierarchia
- Pevne zadefinované typy nadriadeným/podriadených organizačných jednotiek
- Málo flexibilná neumožňujúca organizačné zmeny

Organizačná hierarchia

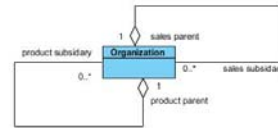


Analysis Patterns

10

Flexibilnejšia štruktúra umožňujúca zmeny: zmena constraints alebo pridaním nového sub-typu a

Dvojitá organizačná hierarchia

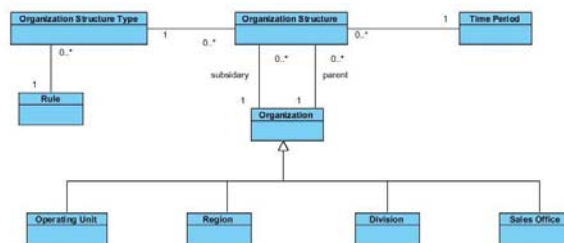


Analysis Patterns

11

1. Hierarchia: predaj
2. Hierarchia: produkt (napríklad PC, Notebooky, Tlačiarne, Fotoaparáty..)
3. Servisný tím podlieha oddeleniu predaja a súčasne produktovému oddeleniu
4. Kde zdefinovať pravidlá o organizačnej štruktúre?

N-dimenzionálna organizačná hierarchia



Model navrhujte tak, že najčastejšie zmeny v modeli zapríčia najmenší počet zmien na typoch

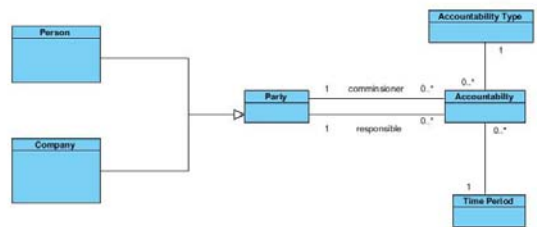
Analysis Patterns

12

Pravidlá o organizačnej štruktúre môžeme umiestniť do:

- Organization Structure Node alebo ešte lepšie
 - explicitným zadenovaním Organization Structure Type
-
- Model funguje dobre ak sa často mení štruktúra ale nie až tak dobre, keď sa menia sub-type organizácii. Každá zmena sub-typov, zapríčiňuje zmeny v pravidlách

Sledovateľnosť / Zodpovednosť



Kedykoľvek definujete vlastnosť alebo zodpovednosť pre typ, ktorý má super-tyt, zvážte, či vlastnosť/zodpovednosť nie je aplikovateľná na tento super-tyt

Analysis Patterns

13

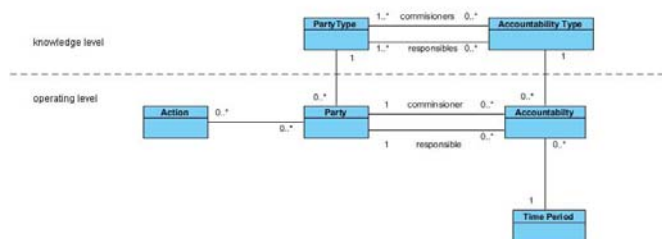
Sú vzťahy len medzi organizačnými zložkami? Nie Organization Structure → Accountability, čo otvára dvere k modelovaniu takých vzťahov ako je:

Zamestnanie: Commissioner: firma, Responsible: Zamestnanec

Riadenie: Commissioner: oddelenie, Responsible: manager

Informovaný súhlas: Commissioner: zdravotnícke zariadenie: Responsible pacient

Operačná vs. Vedomostná úroveň



Explicitne rozdeľte model na operačnú a vedomostnú úroveň

Analysis Patterns

14

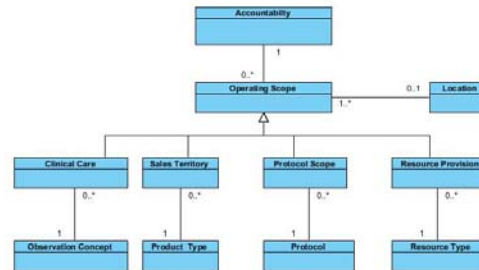
Vedomostná časť znižuje komplexnosť operačnej časti

Alternatívne mená pre toto členenie: model vs. meta-model, (problém meta-model sa používa na popis notácie na dokumentovanie modelov) alebo model/definition model

Podobnosť: class/object (trieda objekt)

Viac dôrazu na konfiguráciu, pretože objekty z vedomostného modelu sa musia vytvoriť pred spustením operačnej úrovne

Operačná pôsobnosť



Analysis Patterns

15

Accountability definuje vzťah medzi 2 osobami (FO alebo PO), ktorému často je potrebné priradiť ďalšie detaily. Tento predmet ich vzťahu vyjadruje operačná pôsobnosť napríklad:

Chirurg je zodpovedný za 20 operácií slepého čreva v Žilinskej nemocnici v novembri 2010.

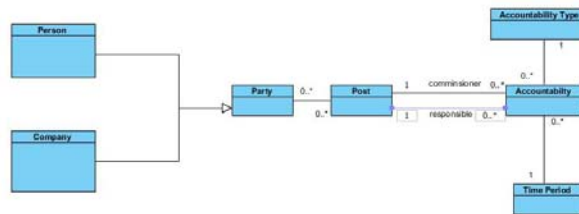
Accountability: Zamestnanie: responsible: Chirurg commissioner: Nemocnica Operational Scope:

Protokol o prevedení 20 operácií slepého čreva v novembri 2010 Location: Žilina

Servisný technik pokrýva trenčiansky a žilinský kraj pre opravu ABC kávovarov.

Accountability: Contract responsible: servisný technik, commissioner: ACM spoločnosť, Operational Scope1: Service Territory trenčiansky kraj Operational Scope2: Service Territory žilinský kraj

Pozícia



Analysis Patterns

16

Ak osoba opúšťa svoju pozíciu, náhradník automaticky preberá všetky zodpovednosti svojho predchodcu.

Zodpovednosti v pozícii sú často definované dopredu pred jej obsadení a osoby sa často menia v pozíciách

Záver



- Analytický vzor je začiatok hľadania, nie konečný cieľ analýzy.
- Analytické model nie sú správne alebo nesprávne, ale oni sú len viac užitočné alebo menej užitočné.
- Doménou vzorov analýzy sú typy/interfaces (nie triedy/classes) a relácie medzi nimi .
- Kedykoľvek definujete vlastnosť alebo zodpovednosť pre typ, ktorý má super-typ, zväzťe, či vlastnosť/zodpovednosť nie je aplikovateľná na tento super-typ
- Model navrhujte tak, že najčastejšie zmeny v modeli zapríčinia najmenší počet zmien na typoch
- Explicitne rozdeľte model na operačnú a vedomostnú úroveň
- Modelujte: „Čo“ a nie „Ako“