

PROJEKTOVÁNÍ RŮZNORODÝCH ÚLOH

Pavel Drbal

Vysoká škola ekonomická, nám. W. Churchilla 4, 13000 Praha 3

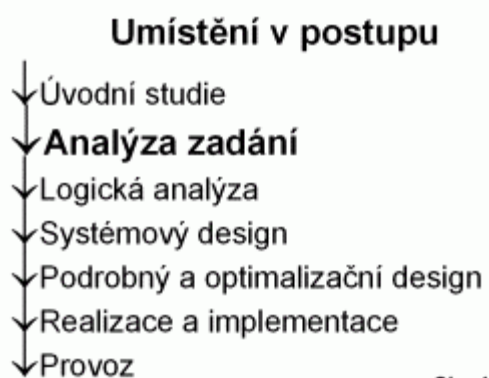
Anotace

Článek se zabývá prostředky objektivě orientovaných metodik pro zvládnutí různorodosti řešených úloh.

Klíčová slova: objektivý, analýza, zadání, model, různorodost

Úvod

Celkové schéma objektivě orientované projekce je celkem známé, v tomto článku bych se chtěl soustředit na různorodost úloh a na prostředky, pomocí kterých se s tím vypořádáme.



Obr. 1

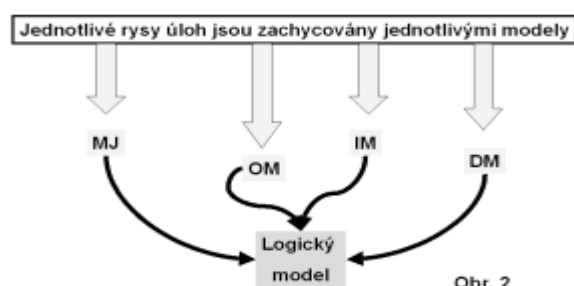
Před projektanta přicházejí nejrůznější úlohy, od úloh umělé inteligence, přes marketing podniků až po úlohy řízení v reálném čase (například letadel). Právě jedním z úkolů **analýzy zadání** je převést různorodost skutečných zadání do vcelku formalizovaného jazyka logických modelů. Analýza zadání má vyřešit:

- Porozumět zadání, odstranit rozpory
- Vytvořit společnou řeč mezi zadavateli, projektanty a uživateli
- Připravit údaje pro další etapy projekce

Skoro se mi chce říci, že analýza zadání plní funkci trychtýře (nálevky), která tuto různorodost standardizuje. Nejrůznější formulace zadavatelů je třeba vyjádřit objektivým jazykem, nebo jazykem modelu jednání a podobně.

Do nedávna převládal názor, že úloha se přímo modeluje v objektivém modelu, ukázalo se ovšem, že to vyhovuje jen některým typům úloha že velká třída úloh s funkčním zadáním, se zpracovává obtížně. V současné době se preferuje použití modelu jednání (Use Case), který se někdy kombinuje s objektivým modelem. Omezíme-li se pouze na tyto dva modely, znesnadníme si svou práci.

Schéma analýzy zadání



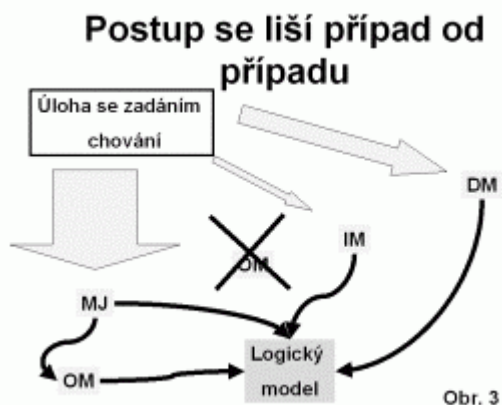
Obr. 2

Pro popis zadání lze použít různé modely strukturující okolí vytvářeného systému a popisující jeho chování k tomuto okolí (vytvářený systém je černá skříňka). Mohou to být modely: **model jednání (MJ)**, **kontextový objektivý model (KOM)**, **kontextový model datových toků (KDM)**, **pojmový model (POM)**, **dynamický model vnějších projevů (DM)** a další.

Jejich charakteristiky jsou uvedeny níže.

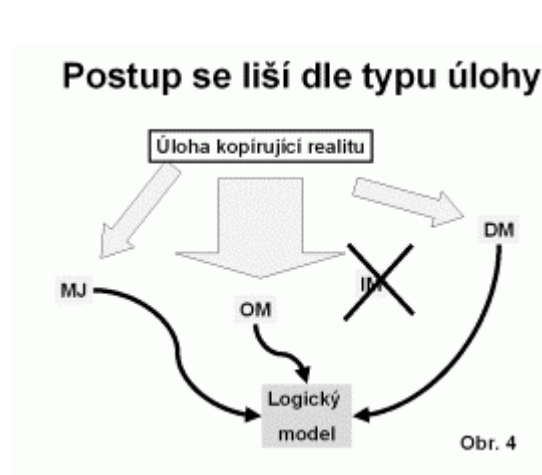
Každý model se hodí pro něco jiného. Přesněji řečeno, jednotlivé modely jsou určeny pro zachycení určitých aspektů reálných úloh.

Tak například model jednání pomocí aktorů zachycuje strukturu okolí vytvářeného systému a pomocí typů jednání zachycuje vnějškové chování budoucího systému. Máme relativně velké množství modelů, aby bylo možno zachytit nejrůznější aspekty úloh - tj. aby se metodikou pokryla co nejširší oblast úloh.



Reálné úlohy nebývají tak komplexní, aby vyvstala nutnost použít všechny modely. Obvykle v úloze vystupuje nějaký rys, který způsobuje převážnou část složitosti - například úloha může mít složité chování ale algoritmicky je jednoduchá (např. zpracování dokumentů - workflow) - může to být ale i obráceně - úloha je složitá algoritmicky a má jednoduché chování (např. kódování nebo komprimace).

Používání těchto „doplňkových“ modelů na nás žádné zvláštní nároky, protože je stejně známe z jiných etap projekce. Jen využijeme svých znalostí v další oblasti.



Příklad

Při tvorbě (reorganizaci) podniku s „plochou“ strukturou hraje informační systém klíčovou roli - do té míry, že návrh struktury podniku a návrh informačního systému jsou současné a že vytvoření informačního systému předchází reorganizaci. V tomto případě jsou informace o struktuře reality nulové, protože tato realita se bude teprve vytvářet, masivní jsou však informace o požadovaných funkcích a o dynamickém chování. Objektový model - stěžejní složka logického modelu - se vytvoří až z modelu jednání.

Ukážeme si jiný příklad: úlohu se zadáním statických vztahů.

Cílem je vybavit informačním systémem dobře fungující firmu - nic na její organizaci nechceme měnit.

Z takového zadání je zřejmé, že hlavním zdrojem informací jsou statické vztahy a že objektový model bude vznikat přímo jako obraz reality.

Statické vztahy nebývají jediným zdrojem informací, vždy jsou požadovány i funkční vlastnosti a mnohdy i dynamické vlastnosti.

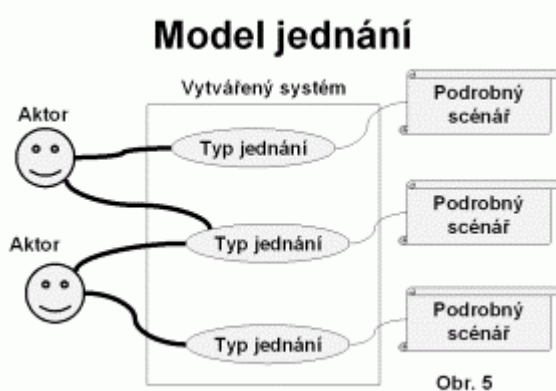
Modely pro analýzu zadání

Výčet těchto modelů:

- jednání
- pojmový objektový
- kontextový objektový
- kontextový datový
- dynamický
- informační

Stručně si je budeme charakterizovat.

Model jednání



Model jednání je celkem známý model, zdůrazníme zde jen jeho základní vlastnosti.

- Vytvářený systém je pokládán za „černou skříňku“, to znamená, že popisujeme jen vnější chování, nic nepředpokládáme o jeho vnitřku. Model jednání slouží jako pomůcka pro přesnou a úplnou formulaci zadání – a v zadání také nepředjímáme řešení systému.
- Model člení okolí systému (tedy to, co bude existovat za provozu) na aktory, prvky bezprostředně komunikující s vytvářeným systémem. Kladu důraz na

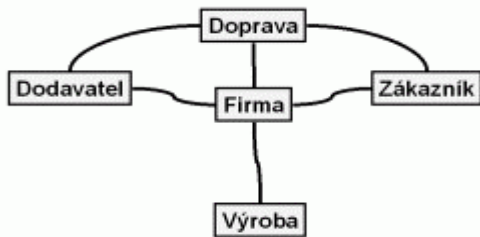
slovo „bezprostředně“ – jedná-li se o programový systém, pak aktorem je ten, kdo ťuká do klávesnice.

- Aktor je role (funkční role), ne člověk. Jeden člověk může postupně zastávat roli několika aktorů.
- Typ jednání je několikaslovní charakteristika komunikace mezi systémem a aktorem. Pohled na aktory a typy jednání slouží k celkovému pohledu na funkčnost a úplnost systému, se zanedbáním podrobností. Jeden typ jednání označuje logicky sevřenou část jedné seance, tedy seance se může skládat z několika typů jednání.
- Nedělejte členění na typy jednání příliš podrobné, svazovalo by vás to v další činnosti. Orientační heslo: „Jeden aktor maximálně pět typů jednání“.
- Každý typ jednání musí mít definovanou výchozí pozici a dosažitelný cíl. To je již věcí scénáře, který je podrobně rozepsán.
- Scénář je posloupnost dvojic „požadavek-reakce“, tato posloupnost musí úplně popisovat všechny možné případy, všechny odbočky, větvení, cykly – prostě všechny možné situace včetně chybových nebo neúspěšných ukončení.

Důležitost scénářů vyplývá z toho, že to je vlastně funkční zadání úlohy, jaké jsou scénáře, takové bude i řešení. Navíc podle scénářů se dělají testy, kterými se prověřuje funkčnost řešení.

Pojmový objektový model

Pojmový objektový model

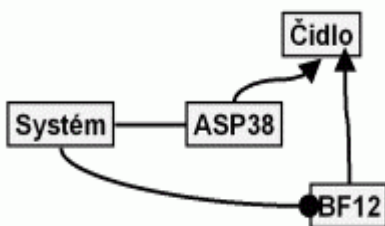


Obr. 6

- Používá se obvyklý grafický jazyk objektového modelu
 - Jména tříd jsou nahrazeny pojmy souvisejícími s úlohou
 - Asociace a agregace vyjadřují vztahy mezi těmito pojmy
 - Atributy a metody se běžně nepoužívají
- Snažíme se vyjádřit vztahy mezi pojmy, které jsou svázány s řešenou úlohou. Nevšímáme si příliš toho, jestli se pojem vztahuje k řešené úloze jako celku nebo k její části.

V podstatě žádný objektový model neděláme, pouze vytváříme seznam důležitých pojmů. Jestliže však tyto pojmy a vztahy mezi nimi zachytíme grafickou řečí objektového modelu, je vzniklý graf velmi inspirativní a pomůže nám uvědomit si důležitost pojmů a vztahů, která by nás běžně nenapadly.

Kontextový objektový model



Obr. 7

Kontextový objektový model

- Vytvářený systém je černou skřínkou, strukturalizuje se jeho okolí
- „Nejdříve porozumím okolí, pak mohu formulovat požadavky a reakce“

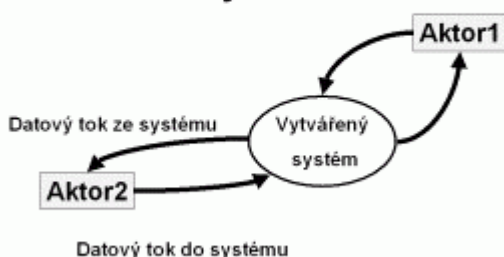
Podobnost s modelem jednání není náhodná, cíl je stále tentýž – zachytit strukturu okolí a určit chování vytvářeného systému.

Kontextový objektový model se soustředí na strukturalizaci okolí a pomíjí chování systému. Je vhodný v případech, kdy je okolí velmi složité a vyznat se v něm není jednoduché – poprvé jsem se s ním setkal u řídicího systému letadel.

Kontextový objektový model je inspirativní v tom smyslu, že nabízí zobecnění. Místo abychom popisovali chování systému ke konkrétním vyráběným čidlům, můžeme uvažovat nad možností, že popíšeme chování systému k obecnému typu čidel.

Kontextový objektový model umožňuje získat přehled o všech možných aktorech a o jejich možnostech v komunikaci, což umožní úplnost modelu jednání. Kontextový objektový model se obvykle používá před modelem jednání.

Kontextový model datových toků



Obr. 8

Kontextový model datových toků

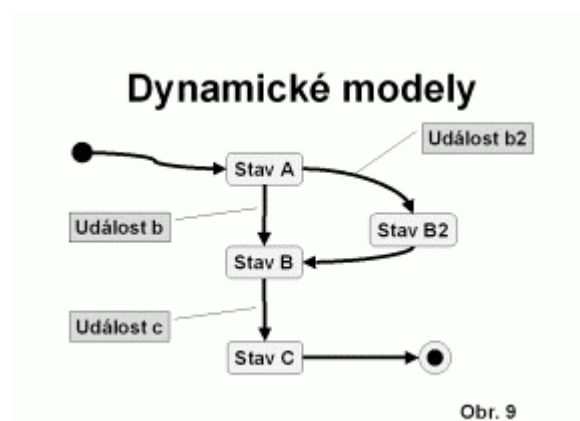
Model datových toků je skoro stejný jako model jednání, jen používá jiný grafický jazyk. To slovíčko „skoro“ je velmi důležité.

Základní přístup je stejný: vytvářený systém se považuje za černou skřínkou a strukturalizuje se okolí vytvářeného systému.

Liší se tím, že model datových toků se zabývá jen takovou komunikací, která je realizována datovými toky, tj. masivnějšími přenosy informací. V tomto modelu zanedbáváme řídicí informace.

Tvorba modelu datových toků není práce navíc, stejně bychom ji dělali v rámci funkčního modelu, jeho přínos je očividný, umožňuje nám zase jiný pohled na aktory a jejich způsob komunikace se systémem. V modelu jednání máme totiž sklon považovat za aktory pouze lidi (je nám to sugerováno například jazykem UML, který zobrazuje aktory jako lidské postavičky) a aktoři – technická zařízení mohou být opomenuti.

Dynamický model



U některých úloh je jejich složitost v jejich chování, ne v interakci nebo ve zpracování dat. U takových úloh bývá centrální prvek, jehož dynamika určuje chování celého systému. Například u běžné firmy takovým centrálním prvkem bývá „obchodní případ“. Jiný příklad: jestliže vytváříme systém pro tok dokumentů (workflow), pak logickou součástí zadání je vnější popis dynamiky dokumentů.

K popisu dynamiky vytvářeného systému se používají různé modely, které jsou navíc ještě modifikovány podle zvyklostí oborů. Mají

společné to, že popisují chování systému jako celku vzhledem k určitému (důležitému) prvku.

Nikdy se nepopisuje úplné chování vytvářeného systému – to ani na úrovni analýzy zadání a upřesňování zadání ani nejde, vždy jde pouze o jeden prvek (nebo několik málo prvků), které však jsou pro chování celého systému určující.

Dynamický model slouží k upřesnění modelu jednání. Jestliže je centrální prvek v jednom stavu, vyžaduje jiné typy jednání (a často i jiné aktory) než když je v druhém stavu. Přechod ze stavu do stavu vyžaduje určitý impuls, který musí zcela určitě být součástí nějakého typu jednání, a mít aktora, který je za tento přechod odpovědný.

Informační model

U úloh, které jsou objektově zpracovávány a mají velkou datovou složitost, se používají informační modely. Jsou to určité obměny datového modelu, jejich role je však poněkud jiná. Jejich role není určit datové jednotky zpracování, mají pouze identifikovat závislosti mezi informacemi (bez ohledu na to jestli uvažovaná jednotka později bude mít charakter samostatného objektu nebo pouze atributu).

K jejich záznamu se někdy používá grafický jazyk objektového modelu (samozřejmě jednotlivé prvky mají jiný význam), jindy je pro jednotlivé metodiky vytvořen speciální grafický jazyk.

Některé metodiky rozšiřují informační model o další vlastnosti (odpovědnost, typy použití, manipulace), takže získává charakter speciálního objektového modelu. Takovému modelu se

říká model business objektů a zastává centrální roli v analýze zadání, tj. nahrazuje model jednání.

Formulace zadání

Výše uvedené modely slouží následujícím účelům:

- Umožňují nám lépe porozumět zadání.
- Tvoří součást zadání (to se týká hlavně modelu jednání)
- Poskytují podklady pro logickou analýzu (hlavně dynamické a informační modely).

Ovšem vytvořené modely ještě netvoří zadání, zadání musí ještě obsahovat následující:

- Slovník termínů, na kterém se sjednotí zadavatelé, budoucí uživatelé a konstruktéři (projektanti) systému.
- Vlastní text zadání.
- Požadavky na rozhraní systému.

Slovník termínů

Během analýzy zadání se musí vytvořit společná řeč zadavatelů, uživatelů a konstruktérů. Lidé, kteří si nerozumějí, nemohou vytvořit společné dílo – již mnoho systémů skončilo tak, že sice krásně fungovaly, dělaly však něco jiného než chtěli zadavatelé a potřebovali uživatelé. Vytvářet společnou řeč je problém lidských kontaktů, který však nakonec musí vyústit do vytvoření společného slovníku termínů, který obsahuje potřebné termíny s krátkým vysvětlením (2 až 8 řádek), na kterém se shodli účastníci všech tří skupin. Tento slovník se během projekce doplňuje o další prvky (například jména komponent nebo tříd). Slovníku termínů se někdy říká model termínů.

Text zadání

Výchozí zadání se obvykle poskytuje ve formě souvislého textu (někdy docela mnohomluvného). Je účelné toto souvislé zadání rozčlenit na krátké textové úseky, každý úsek obsahuje jedno tvrzení o prvcích vytvářeného systému. Prvky a pojmy takového zadání jsou zařazeny do slovníku termínů s potřebnou jednoznačnou definicí.

Toto rozbití zadání na jednotlivé požadavky je velmi vhodné zvláště z hlediska kontroly práce – vždy mohou požadovat odpověď na otázku, který objekt je odpovědný za splnění určitého požadavku.

Takovému zadání rozčleněnému na jednotlivé podmínky nebo tvrzení se říká model požadavků, nebo také model pravidel, případně model věcných pravidel (business rules).

Někdy se model pravidel (tj. textová část zadání) rozděluje na obecnou část, která se používá převážně v logické analýze, a na model provozních vlastností, jehož pravidla se používají převážně v designu. Tento model provozních vlastností se například týká doby odezvy, omezení na výpočetní nebo paměťovou kapacita apod.

Rozhraní systému

Jako samostatná část (model) se vyčleňují požadavky na rozhraní celého systému. Rozhraní na aktory – technická zařízení je samozřejmou součástí systému. Rozhraní na lidské aktory je poněkud problematičtější, protože je dáno požadavky psychologie práce a zvyklostmi uživatelů. Obvykle vzniká v dialogu mezi konstruktéry a budoucími uživateli – musí být kompromisem mezi přepychem a možným.

Závěr

Je vidět, že objektový přístup poskytuje dostatek prostředků pro zvládnutí nejroztodivnějších úloh.