

# Objektově orientovaný vývoj aplikací s použitím Select Perspective a SELECT Enterprise

Ing. Michael Hanke (LBMS ČR)

**SELECT Enterprise je objektový nástroj CASE anglické firmy SELECT Software Tools, který je určen pro řízený rychlý vývoj aplikací. SELECT Enterprise je podpořen prakticky zaměřenou metodikou Select Perspective.**

## 1. Metodika Select Perspective

Metodika Select Perspective je objektově orientovaná metodika, kterou vytvořil Stuart Frost, výkonný ředitel firmy SELECT Software Tools. Vychází ze zkombinování technik OMT (James Rumbaugh) a OOSE (Ivar Jacobson).

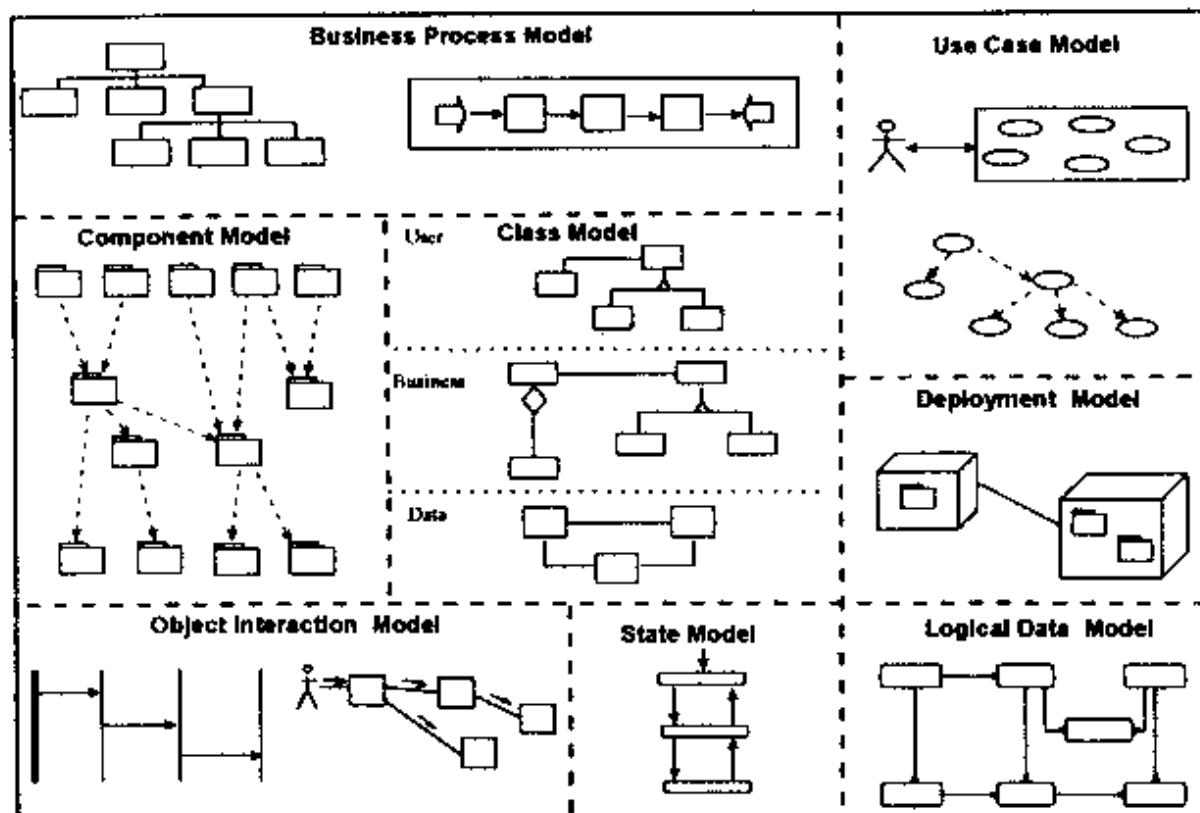
Perspective se od většiny ostatních metodik liší především mírou pragmatičnosti, kterou její tvůrci považují za největší příčinu jejího úspěchu na trhu. Stejně jako se vyvíjely softwarové nástroje pro modelování, rozvíjela se postupně také obsahová náplň metodiky Perspective do současné praktické podoby, získané na základě mnoha zkušeností z realizovaných projektů. Pragmatičnost SELECT Perspective se následně odráží na vzniku nové koncipované generace objektových nástrojů CASE.

Perspective je metodika vývoje celopodnikových informačních systémů s využitím komponent. Celopodnikové informační systémy jsou systémy pokrývající velkou část komplexních podnikových procesů (tzv. "business" procesy). Nejdůležitější vlastností takovýchto systémů je potřeba jejich rychlého vytvoření a dodání v co nejkratším termínu a s maximálním využitím opakovaného použití komponent. Perspective je tvořena souhrnem z praxe nejosvědčenějších modelovacích *technik*, které jsou aplikovány v různých fázích popisovaného *životního cyklu* projektu a zasazeny do celkové logické *architektury*.

### 1.1 Modelovací techniky

Perspective obsahuje objektově orientované techniky pro modelování různých aspektů vyvíjené aplikace. Všechny techniky jsou založeny na standardní notaci UML (specifikaci z roku 1997), kterou za účelem dosažení maximální úrovně praktické použitelnosti v některých případech zjednodušují a nebo naopak doplňují. Stávající verze metodiky pokrývá tvorbu následujících typů modelů (viz obr. 1): model podnikových procesů (Business Process Model), tzv. Use Case model, model tříd (Class Model), model interakce mezi objekty (Object Interaction Model) zahrnující dva typy diagramů tzv. Object Sequence Diagram a Object Collaboration

Diagram, dále model stavů (State Model), model komponent (Component Model), tzv. Deployment Model a konečně logický datový model (Logical Data Model), jehož přítomnost je důsledkem značného rozšíření relačních (a jiných neobjektových) databází. Většina technik je využitelných jak v projektech vývoje specifických řešení tak v projektech vývoje znovupoužitelných komponent.



Obr. 1: Modely používané v Select Perspective

## 1.2 Architektura

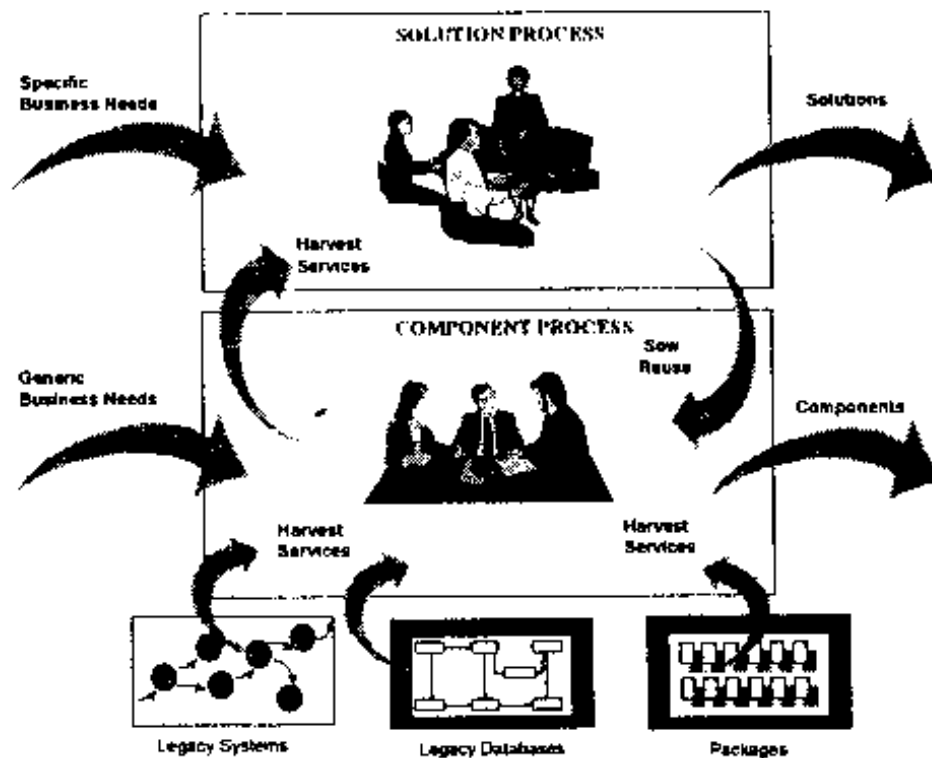
Vývoj aplikací založených na komponentách vyžaduje vytvoření architektury, která umožní efektivní vytváření a využití komponent při vývoji celopodnikových aplikací. Komponenta je spustitelná část kódu, která zapouzdřuje svoji implementaci formou černé schránky (tzv. black box) k níž se přistupuje prostřednictvím nabízených služeb. Ke službám komponent lze přitom přistupovat pouze přes zveřejněné konzistentní standardizované rozhraní. Jednotlivé komponenty lze připojovat prostřednictvím komunikačního rozhraní k jiným komponentám a vytvářet tak jejich větší celky. Perspective tuto myšlenku opakovaného využívání služeb podepírá svou celkovou koncepcí pro návrh znovupoužitelných komponent, vyjádřenou prostřednictvím tzv. architektury založené na službách (service based architecture). Každá vrstva architektury nabízí rozdílné typy služeb:

- uživatelské služby (User services) - využívají aplikační služby k plnění požadavků uživatelů, přispívajících k realizaci podnikových procesů
- aplikační služby (Business services) - konvertují data obdržená od datových a uživatelských služeb na informace
- datové služby (Data services) - nabízejí přístup k datům způsobem, který je nezávislý na fyzické formě uložení dat

### 1.3 Životní cyklus

Perspective se snaží v maximální míře integrovat modelovací techniky popsané výše do jasně definovaného životního cyklu (procesu) vývoje aplikace. Za účelem dosažení efektivního řízeného vývoje aplikací (controlled RAD) rozeznává Perspective dva typy procesů (typových životních cyklů):

- proces pro vývoj řešení - určen pro projekty vytvářející specifická řešení (jde především o vývoj uživatelských služeb se snahou maximálního opakovaného využití služeb existujících komponent, viz obr. 2)
- proces pro vývoj komponent - určen pro projekty vyvíjející komponenty (jde především o vývoj obecně použitelných aplikačních a/nebo datových služeb)



Obr. 2: Dva typy procesů v metodice Select Perspective

### 1.4 Řízení komponent

Významné vzájemné souvislosti mezi procesy pro vývoj specifického řešení a vývoj komponent popisované v kapitole 1.3 vyžadují nutnost existence mechanismů pro efektivní řízení komponent. Toto řízení je vhodné zabezpečit prostřednictvím nástrojů pro řízení komponent. Tyto nástroje umožňují uživatelům registrovat, vyhledávat a instalovat libovolné komponenty ukládané v repository. Dobrý nástroj pro řízení komponent umožní organizacím zpřístupnit své komponenty širokému okruhu potenciálních zájemců s využitím sítě Internet a naopak - umožní registrovat zájem uživatelů o komponenty. Repository obsahuje definice rozhraní jednotlivých komponent a služeb, které jsou prostřednictvím rozhraní nabízeny. Taktéž by měla umožňovat přístup k modelům komponent, vytvářených během procesu modelování. Popisovanou funkčnost pokrývá v nabídce společnosti SELECT Software Tools nástroj SELECT Component Manager.

## 2. Objektový CASE SELECT Enterprise

SELECT Enterprise je objektový prostředek CASE, jehož cílem je zefektivnit práci jak analytikům a návrhářům, tak programátorům objektově založených aplikací. SELECT Enterprise již od verze 5.1 (pozn: v současné době se připravuje uvedení na trh již šesté verze) podporuje tvorbu komponent a vytváření aplikací jejich opakovaným používáním. Významnou vlastností je též kompatibilita se standardem UML, jehož záměrem je sjednocení používaných technik pro modelování.

### 2.1 Popis procesů v organizaci

Pro statický popis procesů v organizaci slouží *diagram hierarchie procesů* (Process Hierarchy Diagram). Rozpad procesů se provádí na úroveň skupin procesů, resp. až do úrovně elementárních procesů v organizaci. Tyto elementární procesy jsou pak modelovány dále.

Dynamika procesů je pak zachycena na *diagramu procesních vláken* (Process Thread Diagram). Tento diagram umožňuje znázornit spouštění procesů vnějšími a vnitřními událostmi. Slouží především pro znázornění work-flow.

Jako základ pro definování požadavků na systém slouží technika Use Cases, kdy jsou definovány jednotlivé úkony uživatelů, které potřebují v systému realizovat a zachycují se na *diagramu typových úloh* (Use Case Diagram).

### 2.2 Definování objektů systému

Pro popis fungování systému je nejdůležitější *diagram interakce objektů* (Object Sequence Diagram), který popisuje Use Case ve formě strukturovaného scénáře se znázorněním komunikace jednotlivých objektů. Tento diagram se skládá ze dvou základních částí. Na levé straně jsou jednotlivé kroky scénáře spolu v různé formě návaznosti (iterace, sekvence, větvení atd.) a v pravé části jsou znázorněny objekty systému a zprávy, kterými spolu komunikují (message passing).

Základním diagramem sloužícím pro návrh systému je *diagram tříd* (Class Diagram), někdy také nazývaný jako statický model objektů. Na tomto diagramu jsou znázorněny vztahy dědičnosti (generalizace resp. agregace), vazby mezi objekty, případně metody a atributy objektů.

Pro objekty, které mají komplikovaný životní cyklus je možné vytvořit i *stavový diagram* (State transition Diagram) a zachytit tak logiku přechodu mezi jednotlivými stavy objektu.

### 2.3 Definování uložení objektů

Definování uložení objektů (Storage Mapping) je použito v případě, pokud jsou data uložena v relační databázi. Při mapování definujeme jakým způsobem jsou z jednotlivých tříd vytvářeny tabulky v databázi, definujeme primární klíče a atributy, které budou v databázi obsaženy. Výsledkem mapování je *fyzický datový model*, který lze použít pro vytvoření struktury databáze.

## **2.4 Tvorba dokumentace**

Pro tvorbu dokumentace slouží komponenta Document Generator. V první fázi si uživatel vybere jaký dokument si přeje vytvářet, např. OMT Class Design. Nyní má možnost vybrat, které sekce dokumentu chce použít. Poté spustí generování dokumentu. Dokument je vytvářen v MS Word pomocí OLE automation. Když je dokument vytvořen, je možné ho aktualizovat, doplňovat do vyhrazených sekcí vlastní popisy, či sekce přidávat a ubírat. Do dokumentu je možné začlenit i sekce vypisující konkrétní zvolené objekty. V případě potřeby je možné samozřejmě vytvořit si vlastní šablony, na základě kterých jsou pak sestavy vytvářeny.

## **3. Závěr**

Objektově zaměřený CASE nástroj SELECT Enterprise v kombinaci s pragmaticky pojatým metodickým zázemím představovaném Select Perspective nabízejí efektivní prostředí pro rychlé avšak řízené vytváření podnikových aplikací.