

VÝVOJ SOFTWAREVÝCH APLIKACÍ POMOCÍ NÁSTROJE

ORACLE DESIGNER/2000

Horčíčka Lukáš, Novotný Petr

Projektování informačních technologií, DESIGN CIS, s.r.o. Žerotínova 56, 787 01 Šumperk, Česká Republika, design_cis@design.cz, www.vsps-su.cz/~design

ABSTRAKT

Rozvoj informačních technologií ruku v ruce se snižováním cen zpřístupňuje stále výkonnější pracovní stanice široké uživatelské veřejnosti. Tímto však extrémně narůstají požadavky uživatelů na stále kvalitnější softwarové produkty, a to kvalitnější zejména ve smyslu intuitivnosti, ovládání a uživatelského rozhraní, při zachování maximální funkčnosti a spolehlivosti aplikace. Ve své podstatě jde vždy o zefektivnění práce uživatele v systému, což přináší značné časové úspory, neboť uživatel není již tolik vystaven stresům z komplikované ovládané aplikace a může se plně soustředit na činnost vyplývající z jeho profese, čímž zvyšuje produktivitu práce, uspokojuje sebe a o spokojenosti zaměstnavatele není třeba hovořit. Tyto aspekty však sebou přináší vysoké nároky na společnosti zabývající se vývojem a projektováním softwarových aplikací, které jsou nuceny pod tíhou požadavků uživatelů své aplikace neustále zdokonalovat, a to samozřejmě s co nejnižšími náklady na vývoj a minimální realizační dobou. Jedinou rozumnou cestou se takto stávají projektové nástroje a metodiky, tzv. Case systémy umožňující uchovat funkční jádro aplikace v bance dat Case systému a zároveň z těchto údajů generovat prototypy aplikací do nových prostředí.

1. ORACLE DESIGNER/2000

Designer/2000 zjednodušuje a urychluje návrhy modelů tím, že využívá výkonné intuitivní metody modelování na úrovni standardů v dané oblasti. K zobrazení struktury a vzájemných vazeb mezi všemi objekty v systému slouží záznamy vazeb mezi entitami, funkční hierarchie, tok dat a techniky modelování matic nebo definice z modelu procesu. Banka dat Designeru/2000 umožňuje kontrolu sdílení a opakované používání aplikačních objektů v několika projektech. Intuitivní prostředí Windows začleňuje běžně užívané desktop aplikace do svých modelů pomocí OLE 2.0 (Object Linking and Embedding). Designer/2000 dále používá navigátor objektů v bance dat pro orientaci v celé struktuře banky dat i pro její zobrazení. Zvyšuje individuální i týmovou produktivitu opakovaným využíváním objektů formou načtení a vložení (drag-and-drop), snadnou modifikaci přiřazení vazeb a komplexním dokumentováním zpráv o struktuře a obsahu sdílené banky obchodních dat. Transparentnost a jednoduchá obsluha prostředků pro modelování systémů, kombinovaná s důmyslnými nástroji navigace v Designer/2000 a sdílenou bankou dat, zajišťují solidní základ pro návrh a implementaci aplikace.

2. BANKA DAT (REPOSITORY)

Designer/2000 obsahuje dokonalý soubor prostředků ke správě obsahu banky dat a ošetření přístupu k nim. Tyto prostředky jsou k dispozici ze všech modelovacích a návrhových modulů a průběžně kontrolují prostředí modelování a projektování. Umožňuje nadefinovat a rozložit aplikace do různých projektů a zajišťuje tak uživatelům práva přístupu k jednotlivým komponentům tak, aby se v součinnosti s uživateli uvedly aplikace na trh.

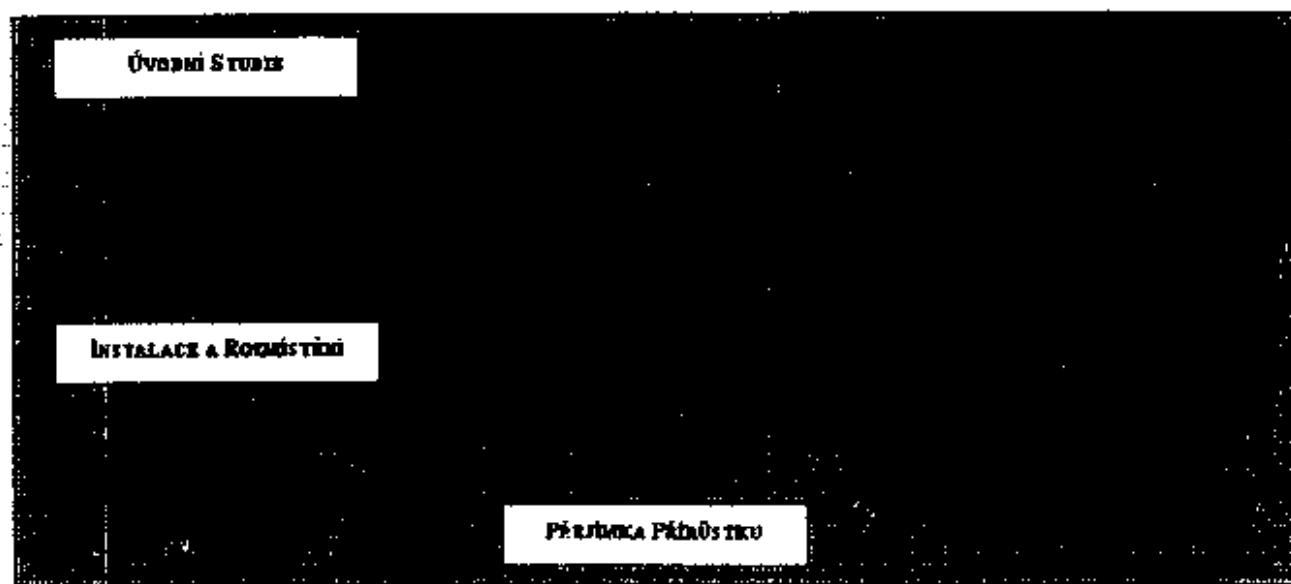
Podporuje opakované užití objektů v celém vývojovém prostředí tím, že sdílí definice v několika aplikačních systémech. Přesně mapuje a pomocí výkonných kontrolních prostředků řídí zpracování různých funkčních a vývojových systémů. V prostředích, využívajících distribuované banky dat Designer/2000, se distribuované projekty integrují pomocí komplexních operací výběru, přesunu a spojování dat. Prostředky správy ve všech komponentech Designer/2000 jsou koordinovány tak, aby rozsáhlý vývoj systému probíhal plynule i v komplexně distribuovaném prostředí.

Správa dat více projektů najednou a opakované užívání dat z jednotlivých projektů je užitečným pomocníkem při návrhu aplikace. Navigátor banky dat poskytuje intuitivní způsob správy objektů. Kontrola verze a unikátnost objektů aplikačních systémů jsou samozřejmostí. Systém banky dat, využívající procedur uložených na serveru, podporuje obousměrný přesun definic mezi Case nástroji.

3. ETAPY VÝVOJE SOFTWARE

Pomocí multimediálních technik lze animovat proces, ilustrovat modely časového průběhu, zachytit výsledky a obchodní příležitosti organizace. Techniky BPR umožňují organizaci důkladně pochopit její obchodní příležitosti a nabízejí jí metody jejich využití. Etapy vývoje software (obr. 1.) je možné rozčlenit do několika úrovní:

úvodní studie, analýza, prototypování, návrh a tvorba, převímka přírůstků, instalace a rozmístění.



Obr. 1. Etapy vývoje software

3.1. ÚVODNÍ STUDIE

Úvodní studii je možné také nazvat poznáváním. Jde vlastně o pochopení zadání nově vytvářeného systému. Z úvodní studie vzejdou ve známost základní entity systému a logický model aplikace. Již v této fázi je možné využívat nástroj Case. Oracle Designer/2000 nabízí pro toto použití dva moduly. Pro zmapování entit je to Entity Relationship Diagramer a pro zmapování logického modelu nabízí Process Modeller. Samozřejmě zatím není možné z těchto úvodních diagramů dělat závěry, ale je možné získat grafický pohled a zadání a pokud je toto zákazníkem odsouhlaseno alespoň z nějaké části, je možné říci, že jsou pochopeny jeho základní požadavky.

3.1.1. HLAVNÍ CÍLE

Hlavním cílem úvodní studie je zmapovat co možná největší množství entit nového systému a sestavit tak logický model aplikace. Je třeba pojmenovat kritická místa, kterými mohou být výstupy do jiných systémů, vstupy z jiných systémů, převody stávajících dat, požadavky na uživatele nové aplikace a samozřejmě nároky na výpočetní techniku. Nemělo by se také zapomenout na nastínění klíčových přínosů pro zákazníka.

3.1.2. POŽADAVKY NA SOFTWARE

Nově vytvářený software musí splňovat nároky uživatelů na současný trend vyvíjených aplikací, musí být plně implementovatelný do prostředí ve kterém je vyvíjen a ovládání aplikace by nemělo být příliš složité.

3.1.3. DEFINICE KRITICKÝCH OBLASTÍ

Kritickými místy každé aplikace jsou komunikace s externími aplikacemi. Importy dat se většinou řeší automatizovanou integrací nebo dávkami (tzn. generací nebo čtením dávek). Jsou samozřejmě případy, kdy dávky jsou jediným rozumným řešením (kompromisem). Co však dělat, když uživatel požaduje pracovat s daty z jiných aplikací. V tomto případě je samozřejmě neefektivní spravovat dvě databanky zákazníků (např. v projektované aplikaci a účetním softwaru).

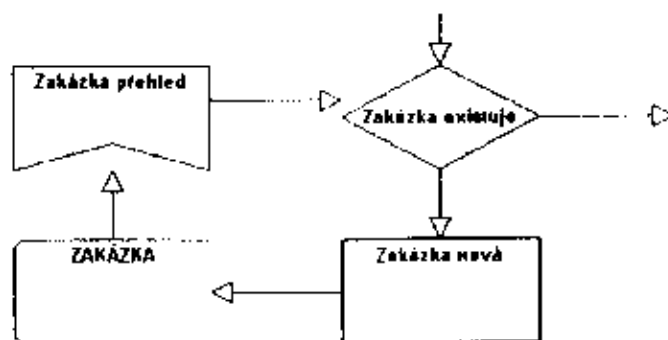
3.2. ANALÝZA

Nejdůležitější částí vývoje software je důkladná a kvalitní analýza. Na tomto kroku je založen úspěch konečného řešení. Dobře analyzovaný problém uložený v bance dat Case nástroje je nejcennějším, co společnost zabývající se vývojem software má k dispozici. Oracle Designer/2000 je jedním z nástrojů podporujících tento způsob vývoje a projektování.

3.2.1. DETAILNÍ ROZPRACOVÁNÍ POŽADAVKU UŽIVATELE

Detailní rozpracování požadavků uživatele je základem k úspěšnému a rychlému zhotovení aplikace. Veškeré informace jsou ukládány do banky dat, kde jsou následně pomocí diagramů zpracovány do uživatelsky příjemné a přehledné grafické podoby.

3.2.2. MODELOVÁNÍ PROCESU

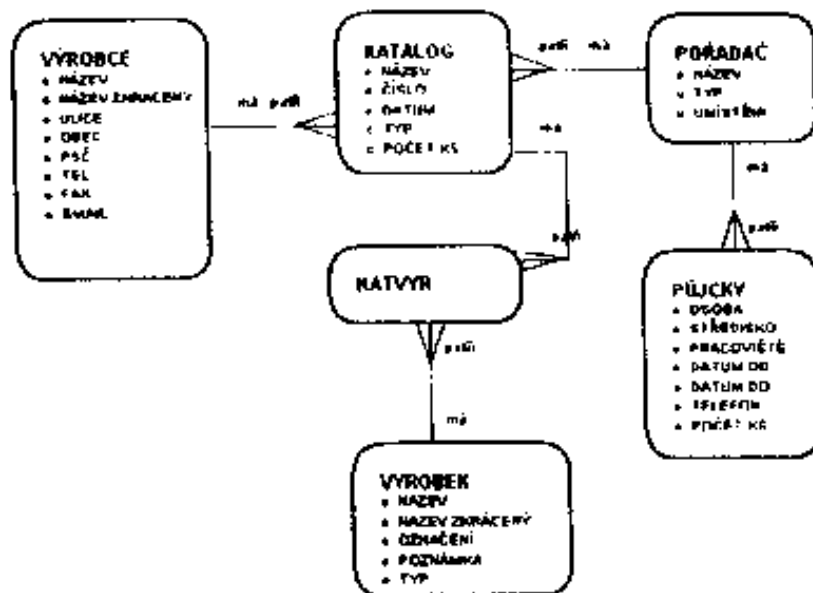


Obr 2. Modelování procesu

Modelování procesů se snaží popsat logickou návaznost všech částí aplikace. K modelování je možné použít animovaných ikon k označení jednotlivých kroků daných procesů, definovat vzájemné vazby a přiřazovat ke každému kroku parametry nákladů a času. Pomocí animací je možné sledovat celý proces od začátku až do konce v příslušném časovém měřítku a hned v zárodkách vývoje tak eliminovat případné pozdější nesrovnalosti v logice systému.

3.2.3. MODELOVÁNÍ VAZEB MEZI ENTITAMI

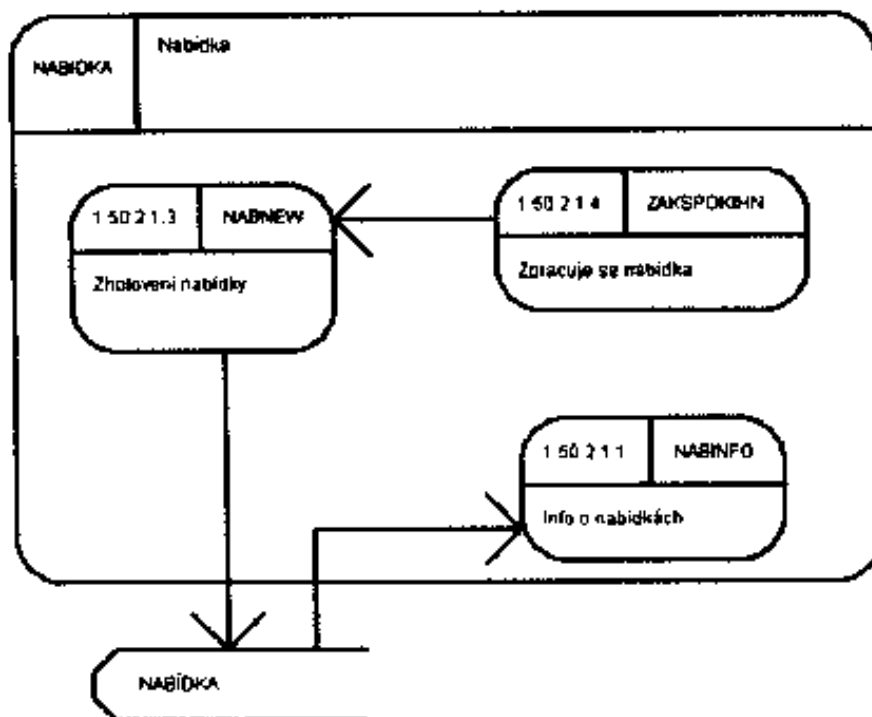
Podporuje tvorbu modelů vazeb mezi entitami a využívá k tomu entity, atributy, vazby, nadřazené a podřizené entity, dále vizuální zobrazení atributů a identifikátorů synonymem nebo diagramů, důmyslné prostředky automatického navrhování struktur, extensivní sestavy o kontrole kvality a další pomocné programy



Obr 3 Modelování vazeb mezi entitami

3.2.4. MODELOVÁNÍ DATOVÝCH TOKŮ

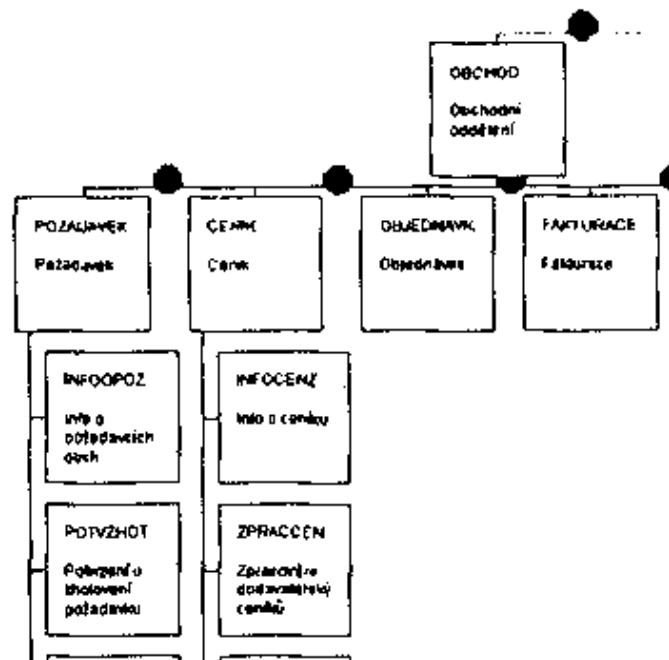
Oracle Designer/2000 obsahuje prostředky pro tvorbu a kontrolu modelů toku dat. Funkce, toky dat a externí entity pomáhají zobrazit datový tok informací v systému. Rychlý výběr a postup po stromu ve funkci hierarchii umožňuje rychlou a přehlednou orientaci.



Obr 4 Modelování datových toků

3.2.5. MODELOVÁNÍ HIERARCHIE FUNKCÍ

Podporuje tvorbu funkčních hierarchií. Ukončuje a prodlužuje větve hierarchie s cílem zvýšit přehlednost modelu. Horizontální, vertikální a kombinované zobrazení dává možnost jak přehlednou formou publikovat vazbu mezi funkcemi. Obsahuje bohaté prostředky pro tvorbu zpráv včetně možnosti analýzy funkce konkrétního bodu a zároveň obsahuje kontrolní body pro sledování kvality



Obr. 5 Modelování hierarchie funkcí

3.3. PROTOTYPOVÁNÍ

Prototypování je klíčovou fází při návrhu systému. Výsledkem je funkční prototyp aplikace, o kterém je možné diskutovat, názorně jej předvádět a přesněji tak uživateli přiblížit konečnou podobu systému.

3.3.1. GENEROVÁNÍ DATABÁZE

Kompletuje dodávky podnikových aplikací tím, že automaticky vytvoří všechny komponenty na straně serveru na základě definic uložených v bance dat. Návrhy tabulek jsou generovány v ANSI SQL DDL skripty jsou generovány z informací definovaných při navrhování systému, a to včetně základních informací o řádcích a sloupcích a validačních omezení cizích i primárních klíčů na straně serveru. Generují se také definice aplikační logiky na straně serveru (PL/SQL), včetně triggerů, uložených funkcí, procedur a parametrů paměti. Distribuci definic databázi je potom možné pro implementaci uložit na cílových serverech a zrychlit tak jejich instalaci. Designer/2000 zajišťuje zpětnou dekompozici existujících definic databáze a poskytuje projektantům rozsáhlé nástroje pro tvorbu diagramů.

3.3.2. GENEROVÁNÍ APLIKACE

Nástroj Oracle Designer/2000 disponuje komplexními prostředky na generování formulářů, zpráv a kódu, vycházejících z modelu aplikace. Generuje takto přenositelné, modulární a vysoce funkční aplikace, přičemž preference automaticky udávají styl, který zaručuje konzistenci a produktivitu návrhu. Vygenerované aplikace obsahují návrhy obrazovek, propracované zprávy, návod k práci s menu, přepínače, nabídkové seznamy, uživatelskou

a aplikační logiku a komplexní ošetření přístupu k databázi - to vše odvozené z tabulek, programových modulů a preferencí, uložených v bance dat. Těsná integrace s nástroji grafického programování pro návrh systémů je samozřejmostí. Zpětný rozklad existujících definic formulářů, zpráv a jejich uložení do banky dat umožňuje zachovat logiku aplikace a snadno ji přenášet na jiné platformy. Opakované generování aplikací chrání již vynaložené investice do vývoje software a umožňuje tak rychleji reagovat na požadavky uživatele. V bance dat lze definovat důmyslné operace sumarizace a seskupování a využívat je tak při zpracování zpráv. Vygenerovaný kód může obsahovat VBX kontroly s podporou OLE 2.0 Designer/2000 generuje aplikace, které lze zpracovávat na různých platformách, přičemž jejich vnější vzhled je stejný bez ohledu na danou platformu

3.4. VÝBĚR PŘÍRŮSTKŮ

Je rozhodující etapou při vývoji software. Pro rychlé zavedení aplikace je důležité vhodně vybírat přírůstky a generovat tak funkční části aplikace.

3.5. NÁVRH A TVORBA

Zde jsou zpracovávány odsouhlasené prototypy aplikací do jejich konečné podoby. Jedná se o závěrečný proces vývoje software, kdy je vytvářeno uživatelské rozhraní aplikace.

3.6. PŘEJÍMKA PŘÍRŮSTKU

Přejímkou přírůstku se rozumí předání smlouvené části aplikace do rutinního provozu.

3.7. INSTALACE A ROZMÍSTĚNÍ

Závěrečná fáze implementace software.

POUŽITÁ LITERATURA

- [1] Moderní metody řízení IS - Molnár Grada 1992
- [2] Návrh a provoz ASŘ - Rajhel, ČVUT Praha 1987
- [3] Modelování rozvoje výrobních systémů - Záruba, SNTL Praha 1992
- [4] Operativní řízení výroby - Tomek, SNTL Praha
- [5] ASŘ strojírenských podniků - Hromková, Molnár, SNTL Praha 1992
- [6] Moderní technologie řízení - Manuelianc, IŘ Praha 1990
- [7] Business Re-engineering with IT - Donovan, CTG, Inc. 1994
- [8] Metodologie Systems Engineering - LBMS Praha
- [9] Sborníky Systems Integration - 93, 94, 95, 96
- [10] WWW.ORACLE.COM
- [11] Firemní projektová dokumentace DESIGN CIS, s.r.o.