

Praktické zkušenosti s LANSOU na projektu Mzdy a personalistika

Jiří Havelka

1. Úvod – stručný popis modulu Mzdy a personalistika

Úvodem stručně popíši modul Mzdy a personalistika. Modul je realizován tak, aby byl provozovatelný jak samostatně tak jako součást systému pro řízení podniku – HARMONY, jehož jsme v ČSFR distributorem.

Základem modulu jsou dva databázové soubory

- soubor Pracovník (klíč rodné číslo)
- soubor Trvalých plateb a srážek (klíč rodné číslo a mzdová složka)

Soubor Pracovník obsahuje cca 150 položek, jejichž údržbu provádí (pokud v podniku v tomto členění existují) útvary ekonomiky práce, personální a mzdová účetna. Přístup k položkám je omezen hesly na zavádění údajů, jejich aktualizaci a prohlížení.

Soubor Trvalých plateb a srážek obsahuje pro každého pracovníka tolik vět, kolik plateb a srážek (pojištění, spoření, exekuce, celá výplata na sporožiro,...) pracovník měsíčně má. Pro každého pracovníka je v souboru alespoň jedna věta – výše měsíční zálohy.

Zpracování je rozděleno a procesy

- Pracovník
- Záloha
- Dobírka
- Měsíční zúčtování
- Roční zpracování
- Personalistika

Celé zpracování mzdové agendy je založeno na tzv. mzdové složce, která identifikuje část výpočtu mzdy pracovníka. Kromě trvalých plateb a srážek (viz výše) jsou to běžné údaje z tzv. „píchaček“. Shodné mzdové složky pro všechny pracovníky jsou modulem generovány automaticky (pracovní dny v měsíci, svátky,...). Ostatní (dovolená, nemoc, refundace, vojenské cvičení,...) zadává zodpovědný pracovník. Tyto údaje jsou zaváděny

do měsíčního souboru, ze kterého je vypočítána částka k výplatě. Pro statistické účely je měsíční soubor archivován.

Výpočtový program je jednoduchý, snadno lze jednotlivé mzdové složky, popř. celé části (např. výpočet daně ze mzdy, různé příplatky,...) z výpočtu mezd vyřadit.

2. Seznámení s LANSOU

V úvodu jsem stručně popsal část projektu Mzdy a personalistika proto, abych mohl na jeho realizaci stručně popsat práci a zkušenosti s LANSOU.

V listopadu 1991 jsme byli zástupcem firmy ASPECT „vyškoleni“. V uvozovkách proto, že úvodní školení sestává z krátkého úvodu do LANSY, zadání příkladů a – pod dozorem školitele si zkoušejte. Nejdříve v APPLICATION TEMPLATES, potom přímo v RDML. Toto školení je dvoudenní. Po úvodním školení následuje pauza na procvičení (1 – 2 měsíce) a školitel přijede znovu, aby zodpověděl dotazy a seznámil s novinkami LANSY. Po měsíci procvičování jsem začal realizovat projekt Mzdy a personalistika. Úmyslně neříkám programovat, protože LANSY poskytuje širokou škálu prostředků pro realizaci projektu.

3. Projekt

Modul nebyl projektován pro LANSU (je provozován na síti PC v několika podnicích). Mohu posoudit zatím pouze realizaci jednoho projektu, který není složitý ani z hlediska výpočtů ani z hlediska vztahů mezi databázovými soubory, procesy a funkcemi. Vzhledem k prostředkům, které LANSY nabízí, nepředpokládám ale zvláštní nároky na projektanta a potřebu školení.

K dokumentaci bude využito HELPů pro jednotlivé databázové položky, procesy a funkce.

Datové modelování verze LANSY, ve které jsem začal realizovat projekt, neobsahovala. V současné verzi je obsaženo, ale zkušenosti s ním nemám.

4. Programovací jazyk RDML

Programátor si velice snadno osvojí instrukce programovacího jazyka RDML. Každá instrukce, při použití PROMPT, nabízí množství parametrů, které ovlivňují její funkci.

Neexistují zde například klasické instrukce MOVE, COMPUTE, READ a WRITE. Místo těchto instrukcí používá programátor instrukci CHANGE, např.:

CHANGE(A) TO(1)

CHANGE(B) TO(1), což je identické se zápisem

CHANGE(A B) TO(1)

CHANGE(A) TO(A - 1) je příklad jednoduchého výpočtu

Namísto instrukce **READ** se používá **FETCH** a **SELECT** položek definovaných ve skupině položek (**GROUP_BY**) nebo listu (tabulce) **DEF_LIST** z určeného souboru. List může mít 9999 listů po 256B, listy je možno třídít, vybírat z nich všechny prvky nebo podle klíče.

Instrukci **WRITE** nahrazuje **INSERT** nebo **UPDATE**. Systém sám ošetří konec souboru, chybové stavy. Existenci věty v souboru testujeme instrukcí **CHECK_FOR**, která podle výsledku testu nastaví systémovou položku. Podle jejího obsahu rozhodne programátor o použití **INSERT** nebo **UPDATE**. Tento případ se často vyskytuje v modulu Mzdy a personalistika a je řešen zobrazením položek s upozorněním na možný přepis údajů. Pracovník který údaje zadává, rozhodne o dalším postupu.

Zde se dostáváme k instrukcím pro zobrazení položek z datové základny (**REQUEST**, **DISPLAY**, **POP_UP** a **BROWSE**), které i s použitím „okének“ přibližují práci na terminálu uživatelskému komfortu PC. Pro tisk jsou instrukce **PRINT** a **UPRINT**. Práce programátora s těmito instrukcemi je velmi jednoduchá, obsahují rovněž celou škálu parametrů. Systém **LANS**A sám, po definování výstupních položek ve skupině (**GROUP_BY**) nebo listu (**DEF_LIST**) rozvrhne položky na obrazovce. Programátor zadá atributy, např položku není možno přepsat, sloupcový nebo řádkový text, bez textu, atd. Po syntaktické kontrole programu může programátor položky na obrazovce (**SCREEN mode**) nebo tiskové sestavě (**REPORT mode**) přemísťovat, rušit, vkládat a měnit rušit nebo vkládat text.

Z programu je možno vyvolávat instrukce **AS/400**. Často bude využívána funkce spuštění dávkového zpracování při dlouhotrvajícím výpočtu nebo tisku sestavy.

Kromě instrukcí **RDML** je možno v programu vyvolat celou řadu funkcí, které jsou součástí dodávky **LANSY**.

5. Postup realizace projektu

5.1. Definice položek

Definujeme typ položky (**A** – Alphanumeric, **S** – Zoned Decimal, **P** – Packed), její rozsah, výstupní atributy, defaultní hodnotu, v případě požadavku na **PROMPT** pro tuto položku – jméno funkce která jej provádí.

Můžeme určit rozsah nabývaných hodnot a postup při jejich nesplnění. Tuto kontrolní funkci je možno vyvolat i z programu.

Při zavádění nové položky do slovníku, stejně jako při vytváření nového souboru, procesu, funkce nabídne systém možnost zkopírovat již existující prvek. Tato funkce s následnou modifikací bude zřejmě často využívána.

5.2. Definice souboru

Programátor je vedem systémem při určení položek ze slovníku dat sdružených v souboru, definici klíčů, definici logického souboru (např. u souboru Pracovník přístup podle příjmení pracovníka).

Můžeme definovat tzv. virtuální pole, což je v podstatě redefinice a platí jen pro soubor, kde jsou definována. Stejně tak je možno určit jakých hodnot smí položka nabývat a to právě v tomto souboru.

Soubory, stejně jako v OS AS/400 je nutno kompilovat.

5.3. Definice procesu a funkcí

Proces obsahuje množinu funkcí, které jsou již vlastními programy v RDML. Koncovému uživateli se zobrazí nejprve nabídka procesů, po volbě procesu nabídka jeho funkcí. Procesy je možno na sebe libovolně navazovat.

6. Zkušenosti s LANSOU

V krátkém příspěvku není možno popsat ba ani vyjmenovat všechny funkce a prostředky, které LANSA nabízí.

Dosud jsem programoval na sálových počítačích (Siemens, EC 1027, SM) v COBOLU se sítí terminálů a seznámil jsem se s RPG III a CL pro AS/400. Realizace projektu v LANSE je nesrovnatelná. Nemám ještě zvládnuty všechny možnosti, které LANSA poskytuje ale přesto mohu říci, že z hlediska produktivity práce, relativní nezávislosti na operačním systému, snadné zvládnutelnosti a výsledků pro koncového uživatele představuje LANSA mocný nástroj pro projektanta i programátora.

Problémy, které se vyskytly při realizaci projektu, nelze hodnotit jako závažné a v podstatě kromě jednoho jsem si se všemi poradil sám. K dispozici jsou poměrně kvalitní manuály s mnoha příklady. Kromě toho nám školitel nechal demonstrační model, včetně zdrojových programů a dat, ze kterého lze rovněž čerpat.

Jediný závažnější problém jsem měl při re-kompilaci souborů. Vyřešil jej zástupce fy ASPECT (nesprávné pořadí knihoven v seznamu knihoven OS AS 400).

Programy, které jsou vytvořeny šablonou (APPLICATION-TEMPLATES) jsou komentovány, je na programátorovi, aby programy v RDML opatřil poznámkami rovněž – údržba je potom snadná.

U všech definic (položek, souborů, procesů a funkcí) je povinný popis (description) a bez jeho vyplnění nedovolí definici uložit. Záleží pouze na zodpovědnosti programátora, jak je popis výstižný.

Na realizaci projektu jsem pracoval sám. Nově nastoupivšímu kolegovi, který se rychle seznámil s LANSOU, jsem krátce vysvětlil projekt a velice brzy sám prováděl úpravy a psal nové programy. To dostatečně dokumentuje snadnost zvládnutí tohoto prostředí.

Programátor potřebuje pouze základní znalosti operačního systému AS/400 (prohlížet a upravovat soubory, pracovat s výstupní frontou, obsluhovat tiskárnu,...). LANSY má vlastní editor, který je ale chudší než editor SEU AS/400. SEU lze ale z LANSY snadno vyvolat.

Přínosy pro programátora

- snadná zvládnutelnost
- bohaté podpůrné prostředky pro vytváření obrazovek a tisků
- potřeba znát podrobně OS AS/400 (i když to není nikomu na škodu)
- nesrovnatelné zvýšení produktivity práce v porovnání např.: s COBOLem, RPG III ap.
- možnost připojení objektů a souborů vytvořených mimo prostředí LANSY
- snadné zapojení modulu nově vytvořeného do již existujícího systému

Nevýhody LANSY

- nejvíce programátor počítá delší dobu trvání kompilace programu a souboru (nejprve probíhá generování zdroje do RPG III a potom jeho kompilace).

Nemám zkušenosti s jiným CASE prostředkem, ale po čtyřměsíční práci s LANSOU – mohu majitelům systémů AS/400 doporučit.

Autor:

Ing. Jiří Havelka

SOBA ČSFR

Dlouhá třída 39

110 00 Praha 1

tel.: (02) 231 51 95, 231 45 38, fax: (02) 231 09 82