

"CASE" nástroje MX Cobol OMS a zkušenosti s jejich využíváním

Vlastimil Čevela

Úvod

Předkládaný příspěvek se zabývá informacemi o využívání programových nástrojů, které sice tvoří ucelený systém CASE dle mezinárodně uznávaných metodik (proto ty uvozovky), ale na druhé straně mohou významně v duchu doslovného překladu "počítačově podpořit softwarové inženýrství".

V uvedených souvislostech mohou být zajímavé některé obecné vlastnosti jazyka Cobol, konkrétní možnosti objektově modulární stavebnice OMS a některé zkušenosti z práce na inovaci programového vybavení z Cobol EC1026 na MX Cobol/PC.

Nástroje, které jsou předmětem našeho zájmu, by měly podporovat řešení následující problematiky:

- A/ Popis prvků a vazeb v řešeném informačním systému formou nějakého počítačového modelu.
- B/ Model musí umožňovat kontrolu splnění požadavků zadavatele a formální konzistence systému.
- C/ Z modelu by mělo být možno (více či méně automatizovaně) odvodit potřebné programové vybavení.
- D/ Při vývoji i po dobu životnosti software by mělo být možno časově rozšířit jednotlivé stavy modelu i odvozeného programového vybavení.
- E/ Náročnost zvládnutí metodiky a vlastního modelování nesmí potlačit cíl, kterým je vytvoření a údržba prakticky využitelného programového vybavení.

Zjednodušeně lze tedy úkol "CASE" formulovat jako:

- Zachycení definic datových a funkčních prvků a jejich vazeb, t.j. předávání dat a předávání řízení.
- Záznamy informací o prvcích je třeba formalizovat tak, aby byla umožněna automatizovaná kontrola konzistence systému.
- Z hlediska dosažení konečného cíle pak mají význam především aktivní dokumentační prvky, t.j. takové, které slouží přímo jako zadání pro automatizovanou tvorbu programového vybavení.

1. Cobol jako "CASE" ?

Asi to bude znít paradoxně, ale některé vlastnosti jazyka Cobol (Common Business Oriented Language), uvedené v 1. definici z dubna 1960, se skutečně snaží řešit i problémy, patřící dnes do oblasti CASE.

V té souvislosti je zajímavá již charakteristika situace, formulovaná jako zadání koncepce pro Cobol na ustavující schůzce sdružení CODASYL (Conference On Data Systems Languages) ve dnech 28. a 29. května 1959, t.j. přesně před 36 lety.

"Je nutné vyvinout pro existující počítače i pro počítače vyráběné v budoucnosti systém, který umožňuje nahrazení jednoho počítače jiným, a to i nekompatibilním tak, aby náklady, spojené s přenosem programů byly minimální."

"Rychlý vývoj vědy a techniky a prudké zvyšování požadavků na řízení vedou k nutnosti častých úprav v programech. Je žádoucí, aby součástí programu byla tak úplná dokumentace, která by tyto změny umožňovala."

"Často je nutno produkovat velké množství programů v krátkých termínech, a to s pomocí relativně nezkušených programátorů."

Samozřejmě, že některé věci byly vývojem překonány, ale např. principy definice dat se tvůrcům Cobolu podařilo navrhnout s takovou předvídatostí, že nebyly dodnes překonány.

Popisy dat

- Všechna data jsou v Cobolu povinně explicitně popisována v samostatných sekcích.
- Datové struktury mohou mít libovolnou skladbu i hierarchické členění.
- Pro data předávaná mezi jednotlivými programy lze definovat jejich struktury jako samostatné členy knihovny projektu, přiřazované při kompilaci programů příkazem "copy".

I/O charakteristika programu

Norma pro Cobol vyžaduje, aby každý datový soubor i jeho datové struktury byly na předepsaných místech programu samostatně definovány. Při vhodné kombinaci tohoto pravidla s jednotlivými definicemi popisu dat a přehledným umístěním příkazů "open input/i-o/output" může zdrojový text programu sám o sobě popisovat i/o charakteristiku datových vazeb.

Podmínková jména

Umožňují velice účelně definovat srovnávací seznamy a intervaly hodnot i stavy řídicích přepínačů a tak snižovat závislost algoritmu programu na zpracovávaných datech.

2. "CASE" nástroje OMS

Objektově modulární stavebnice (OMS), vyvinutá firmou Čevela jako původní, objektově orientované vývojové prostředí pro kompilátory MS/MF a MX Cobol, poskytuje celou řadu užitečných nástrojů typu "CASE". Vycházejí, obdobně jako koncepce Cobolu, z již zmíněného principu "aktivní dokumentace", t.j. formát definice datových i programových prvků je navržen tak, aby sloužil v maximální míře též k popisu systému. Stručná charakteristika některých z nich je uvedena v dalších odstavcích.

bdir - cdir - mdir

Tvorba seznamů souborů dle kritérií MS DOS příkazu "dir". Jména souborů v seznamech mohou být automaticky komentována. Seznamy souborů jsou základním řídicím prvkem pro hypertextovou dokumentaci.

for - do

Možnost opakovaného provádění dávky pro skupinu souborů (parametrů), definovaných seznamem. Velice užitečná při hromadných kompilacích, přesunech souborů i při archivaci.

fdir

Vyhledání řetězce ve skupině textových souborů, definované seznamem nebo adresářem.

Objektově orientovaný generátor programových textů

Skládání výsledných zdrojových programů Cobol, dávek MS DOS či jiných textových souborů z předem definovaných typů (tříd) s rozsáhlými možnostmi parametrizace.

crf - cri - cmm

Parametricky řízená tvorba a konverze dávkových souborů a seznamů.

rychlý hypertextový editor

- možnost přímého přístupu k souborům i s případným vyhledáváním řetězce v libovolně složité datové základně či dokumentaci
- přímý přístup ke všem dílčím zdrojům programové jednotky (copy členy, typové moduly, aplikační moduly)
- možnost hypertextového ovládání programů nebo dávek přímo z dokumentačních textů

ICDIR

Systémový podprogram kompilátoru MX Cobol, umožňující přímo v programu získat kompletní informace o obsahu adresářů MS DOS a tak řešit různé úlohy, související s časovým rozlišením stavu vývoje.

3. Zkušenosti z inovace Cobol EC1026/MX Cobol PC

Zadáním byl převod něco přes 200 programů v rozsahu od několika desítek do cca 3 tisíc zdrojových řádků.

Programy vznikaly a byly aktualizovány po dobu asi 20 let různými lidmi v různých stylech, včetně zdrojů pro generátor DOGA. Různorodost byla doplněna nutností nově vytvořit interaktivní vstupy pro pořizování dat s verifikací v datastanici i pro přímou práci uživatelů.

Výsledkem je programátorsky řízené a dále udržované kombinované interaktivně-dávkové zpracování, které pomocí asi 50 nabídek řídí kolem stovky výpočtových kroků.

Složitost úkolu nebyla ani tak v převodu vlastních programů, protože MX Cobol je (s výjimkou velkých programů, které je nutno rozdělit) poměrně dobře kompatibilní na EC Cobol a generátor OMS dokáže zpracovat i zdroje formátu DOGA.

Hlavním problémem pro řešitele byla orientace a zachování správného předávání dat a předávání řízení ve zcela neznámém projektu se stovkami prvků a návazností a udržení průběžné bezproblémové komunikace se zástupcem zadavatele.

Ruční vedení potřebné dokumentace nepřicházelo v úvahu a nesourodost i finanční a časová omezení vylučovaly nasazení klasického CASE. Proto byly využity dále popsané "CASE" nástroje MX Cobol 1.55 OMS.

Nabídkový manažer

- univerzální prostředek pro definici a spouštění libovolně složité struktury programů i dávek
- umožňuje kontextovou nápovědu, parametrizaci při spouštění jednotlivých kroků i různé úrovně ochrany proti omylu i zneužití včetně volitelného protokolování prováděných akcí
- jeho součástí je též programová nadstavba k průběžné dokumentaci projektových vazeb

i/o charakteristiky

Do všech zdrojových programů a definic výpočtových kroků byly doplněny jejich stručné, několikařádkové vstup/výstupní charakteristiky v předepsaném formátu.

Dále byla zpracována cca 2 stránková i/o charakteristika celého projektu, tj. přehled všech přebíraných, aktualizovaných (archivovaných) a předávaných datových souborů.

Automaticky odvozovaná dokumentace

S využitím konfiguračních souborů, které definují výpočtové kroky v nabídce, zdrojových textů programů a i/o charakteristiky projektu jsou automaticky odvozovány následující informační sestavy:

- datové soubory ve výpočtových krocích
- výpočtové kroky a jejich i/o charakteristika
- vazby nadřazenosti a podřazenosti mezi nabídkami
- posloupnost a i/o charakteristika programů v rámci výpočtových kroků v nabídce

Z uvedených sestav lze vysledovat datové toky, formální úplnost datových vazeb na úrovni souborů i předávání řízení mezi nabídkami.

S využitím hypertextového editoru je pak přístupný prakticky libovolný detail celého řešení.

Závěr

V žádném případě nebylo cílem tohoto příspěvku zpochybňovat význam systematických metodik a výkonných CASE nástrojů pro tvorbu SW produktů. Jejich uplatňování je však časově (osvojení si nové metody práce) i finančně (ceny systému CASE v desítkách až stovkách tisíc Kč) poměrně náročné, takže je výhodné především pro práci na rozsáhlejších projektech ve větších řešitelských kolektivech.

Výše podané informace byly vedeny snahou ukázat na možnosti nástrojů, které mohou být užitečné především při programování jednotlivých programů a tvorbě menších projektů, ale mohou též významně podpořit inovaci existujících programových systémů. Měly by být proto chápány jako specializovanější doplněk k problematice tohoto semináře.

Literatura

- [1] Bébr R., *Ecce Cobol sapiens* (recenze MX Cobol), *Softwarové noviny* č. 10/92, str. 68
- [2] Čevela V., *Počítačová tvorba dokumentace projektu zpracování hromadných dat s využitím prostředků jazyka Cobol Tesla 200*, Sb. *Používání jazyka Cobol*, DT Pardubice 1979
- [3] Drbal P., *Historie Cobolu*, Sb. *Základy technologie programování v jazyku Cobol*, DT Pardubice 1981
- [4] Kanisová H., *Case Systems Engineer*, Sb. *Programování '94*, DT Ostrava
- [5] Ryant I., *Brontosaurus to přežil* (recenze MX Cobol), *Bajt* č. 6/92, str. 64

Autor

Ing. Vlastimil Čevela
 Programátorské služby Cobol
 Benešova 279
 664 42 Modřice

tel. 05 - 303 367 (6-10 hod)
 fax 05 - 773 044 (8-16 hod)