

METAINFORMAČNÍ SYSTÉM NA MIKROPOČÍTAČI

Jiří Pechlát

Metainformační systém je velmi účinný ale i náročný nástroj. Proto jsem začal uvažovat o ulehčení cesty k němu.

1. Metainformační systém, jeho zdroje a obsah

Považuji za rozumné ujasnit si některé pojmy. Metainformační systém (MIS) je systém popisující jiný (tzv. objektový) systém. Teprve tento popisovaný systém (existující nebo projektovaný) zobrazuje informace o zájmových objektech. Metainformační systém je vlastně hierarchicky vyšší informační systém v jisté struktuře informačních systémů. Jeho postavení je však odlišné od objektového systému.

Prvotním impulsem k vývoji metasystémů byla snaha organizačně a mentálně zvládnout složitý systém provozovaný na počítači. Vývoj v tomto směru vrcholí zavedením slovníků a adresářů dat (SAD), které popisují datovou a později i procesní část systému.

Zvláštní kapitolou jsou snahy o věcné zmapování datové základny. Charakteristické je spoluúčast a někdy i iniciativa uživatelů. Jde přitom o racionalizaci celého datového fondu nejen jeho automatizované části. Koncepce Jednotné údajové základny organizací (JÚZO) předpokládá existenci jednotné soustavy ukazatelů založené na katalogích definic prvků, katalogích informačních vět, katalogích algoritmů pro odvozování prvků, katalogích číselníků.

Vedle toho vznikaly metody a postupy pro modelování a projektování ASŘ později s počítačovou podporou, které smě-

řovaly k systémům automatizovaného projektování (SAP). Jde vlastně o nadstavbu MIS, vznikaly však nezávisle na BAD a vytvářely vlastní datovou základnu s charakterem MIS.

Všechny tyto snahy a odpovídající nástroje jsou určitou vývojovou etapou. Postupně dochází ke sblížení všech tří pohledů a vykrytalizování pojmu metainformační systém.

Náplň MIS souvisí s funkcemi, které má MIS zajišťovat nebo informačně podporovat. Jsou to projektování ASŘ, jeho řízení a dokumentace. MIS by měl zahrnovat

- datovou základnu popisující objektový systém,
- systém řízení báze dat,
- popisný metajazyk,
- dotazovací jazyk vysoké úrovně,
- generátor sestav a dokumentace.

Nad vlastním MIS musí nutně existovat nadstavba v podobě nástrojů pro řešení úloh systémové analýzy a úloh optimalizačních, které umožní prověřovat strukturu modelovaného objektového systému.

	! konstr. prvky!	funkční model	! datový model
koncept	- prostředky	- procesy	- entity
systému		- atributy	- atributy
	- charakteristiky	- vztahy	- vztahy
implementace!		- programy	- soubory
	- vztahy	- atributy	- záznamy
		- vztahy	- položky

Princip datové základny MIS ilustruje uvedené schéma. Vzniklo na půdě datové analýzy a dovolil jsem si upravit. Zařazení konstrukčních prvků je vynuceno projekční činností. Jde vlastně o (potenciální) realizační prostředky procesů.

jako jsou programy, projekty, ale také technické prostředky využívané při automatizovaném nebo ručním zpracování dat i lidé, kteří data zpracovávají.

2. Zavedení a využívání MIS

Rozhodující otázky, které je třeba řešit, se dají shrnout do tří bodů:

1. ujasnění účelu a struktury MIS,
2. volba vhodného nástroje,
3. vytvoření příznivého prostředí.

Účel může být různý podle zaměření uživatele. Zásadní význam má otázka rozsahu zobrazovaného systému. Nejkomplexnější přístupy považují za objektový systém celý systém řízení včetně neautomatizované části.

Bylo by nejlepší mít pro různé třídy modelovaných systémů připraveny odpovídající základní struktury MIS. mnohdy ovšem vzniká MIS teprve dodatečně ve chvíli, kdy je realizován rozsáhlý systém s řadou nedostatků. Pak je nutné počítat se zobrazením systému včetně všech nekonzistencí.

Volba vhodného nástroje souvisí s účelem, pro který je MIS vytvářen. Často jsou používány slovníkové systémy, zejména pokud jsou součástí dodávky databankového systému.

Posledním bodem rozumím organizační a metodické předpoklady spočívající v dostatečném stupni standardizace a formalizace, v jednotné metodice pokud možno v celém řešitelském kolektivu, v dobré přípravě uživatelů i řešitelů ASŘ. Patří sem i překonání jistých psychických bariér, které se objevují zejména u zkušených projektantů a programátorů.

3. MIS na mikropočítači

Vešmi stručná charakteristika z předchozích kapitol platí i pro MIS na mikropočítači. Přesto jsou tu jisté zvláštnosti. Za nejvýznamnější považují tvorbu mikropočítačových sítí (hlavně LAN), které zřejmě budou nejčastější aplikací sítí v dalších letech. Je nejvýše aktuální uvažovat o popisu distribuovaných systémů.

Pro nejbližší období však předpokládám jiné cíle, dané situací na vlastním pracovišti (a patrně i jinde), kde dosud není využíván slovníkový systém ani podobný racionalizační prostředek. Příležitostí ke změně, tj k založení a využívání slovníkového systému a jeho postupnému rozvinutí v MIS, by mohla být instalace nového centrálního počítače. Velký systém však bude vyžadovat velké množství práce z hlediska ASŘ vlastně "neproduktivní". A jsme u psychické bariéry. Vážnou překážkou je i nedostatečné standardizace a formalizace a žádná zkušenost.

Kromě centrálního výpočetního střediska vzniká v podniku soustava rozděleného zpracování dat na mikropočítačích. Mikropočítače jsou samozřejmě k dispozici i projektantům a to mne přivedlo k představě, jak zmenšit propast, kterou musíme překročit, jakmile budeme chtít využívat velký slovníkový systém.

Udávám se, že i nepřilíže složitý podpůrný systém může přinést zlepšení podmínek pro rozvoj ASŘ při poměrně malých nákladech. Musí jít o systém aplikovatelný v malém rozsahu (třeba i jediným pracovníkem), který by přinášel prokazatelné výsledky a mohl být nasazován postupně a nezávisle na prostředí. Jeho používání pomůže vytvořit normy a standardy projekčních a programátorských činností, podpoří u řešitel-

lů pozitivní návyky a postoje k MIS. Lze uvažovat i o přímém automatickém převzetí jeho datové základny do budoucího MIS.

Určité omezení představuje v této chvíli malý rozsah vnitřní i vnější paměti disponibilního mikropočítače. Omezení postupně zmizí se zvětšováním schopností mikropočítače.

Ze všech funkcí, které MIS podporuje, má na mikropočítači největší šanci vytvoření modelu systému pro projektování spojené s ověřováním některých vlastností navrhovaného systému (konzistence) a s vygenerováním kostry dokumentace.

4. Systém APP

Hlavním cílem je tedy vytvořit jednoduchý systém zaměřený na podporu projektování ASŘ. V další části budu tento systém nazývat "automatizovaná podpora projektování na mikropočítači" (zkráceně APP).

Druhým cílem je zapůsobit prostřednictvím využívání systému APP na postoje řešitelů ASŘ. Lze, myslím, reálně předpokládat postupné prosazení systému APP do používání.

Systém je realizován na osmibitovém mikropočítači Robotron 1715. Operační systém SCPY je kompatibilní se systémem CP/M. K realizaci byl zvolen pseudorelační databázový systém dBaseII vybavený účinným dotazovacím jazykem a možností programování. Umožňuje i snadnou modifikaci programu, což je výhodné pro přizpůsobování konkrétní potřebě uživatele.

5. Struktura APP

Struktura je zachycena následujícím diagramem. Nepovažuji za nutné jít do podrobností.

10
APP

11	12	13	14	15
Tvorba modelu	Prohlížení Iní modelů	Test konzistence	Generování	Pomoc

Tvorba modelu zajišťuje interaktivní vytváření a modifikaci modelu (prvků a vztahů). Přitom jsou provedeny základní kontroly (duplicita jmen, vztahy, které musí být povinně definovány atd).

Prohlížení modelu je také interaktivní a zajišťuje pohyb v rámci definovaného modelu buď o krok na související prvek nebo skokem na libovolný prvek modelu podle jména. Přitom je možné kdykoliv přejít do tvorby modelu. Lze prohlížet pro libovolný prvek modelu všechny jeho definované vztahy.

Test konzistence probíhá v dávkovém režimu. Jsou prováděny základní kontroly jako při tvorbě modelu a navíc další testy. Zásadně musí platit, že každý prvek modelu musí mít definován alespoň jeden vztah k jinému prvku modelu. Výstupem je signalizace nekonzistencí v tiskové sestavě.

Generování probíhá v dávkovém režimu. Zajišťuje výstup tiskových sestav buď přímo na tiskárnu nebo do textových souborů pro další zpracování textovým editorem. Výstupem může být popis struktury datového záznamu nebo programová specifikace.

Funkce "pomoc" umožňuje dotaz na vysvětlení pravidel používání celého systému APP nebo jeho vybraného příkazu.

Komunikační jazyk systému APP je český jazyk na bázi češtiny. Jeho příkazy jsou implementovány jako příkazové

soubory dBaseII. Tím je umožněn dvojitý způsob vyvolávání příkazů APP. Jednak z navrhovaného systému jménem příkazu, jednak přímo z dBaseII příkazem DO, jímž se vyvolává činnost příkazových souborů. Systém APP nabízí standardní parametry, jimiž jsou platné typy prvků a jména platných výskytů prvků modelu (tj prvků, s nimiž naposledy pracoval).

Přehled jednotlivých příkazů:

Tvorba modelu:

MODEL - založí nový model a jeho kořenový proces,
ZALOZ - založí nový prvek modelu,
ZMEN - změní popis platného prvku,
ZRUS - zruší platný prvek a všechny odvolávky na něj,
ALIAS - přiřadí platnému prvku alias jméno,
SPOJ - je-li to přípustné, založí vztah mezi prvky,
ROZPOJ - zruší zobrazený vztah.

Prohlížení modelu:

SKOK - zobrazí prvek uvedeného typu a uvedeného jména,
KROK - zobrazí další výskyt prvku téhož typu,
SEZNAM - zobrazí jména všech prvků uvedeného typu,
PREPNI - změní platnost prvku, jsou-li zobrazeny dva,
VZTAH - pro platný prvek zobrazí definovaný vztah a zároveň partnerský prvek.

Testování modelu:

TEST - testuje konzistenci modelu.

Generování výstupů:

GENER - generuje popis věty nebo specifikaci programu

Podobí:

POMOC - zobrazí uživatelskou příručku,
<příkaz>? - zobrazí popis uvedeného příkazu.

Jiné:

KONEC - ukončí práci APP i dBaseII,
klávesa ET (end of text) - opakuje předchozí příkaz,
x: - změni diskovou mechaniku s modelem.

6. Zkušenosti a perspektivy

Projekt vznikl postupně jako moduly pro ulehčení dokumentace projektu, na kterém jsem pracoval. Moduly měly pragmatický ráz. Teprve postupně jsem se snažil z jednotlivých modulů vybudovat systém.

Během detailního návrhu jsem si uvědomil, že dBaseII není příliš vhodný nástroj pro krokování v databázi jako souvislé síti prvků. Uvažuji o převodu snad do Fandu.

Zkušenosti i z neúplné verze jsou příjemné, zejména snadnost provádění změn a kontrola konzistence, i když je na nízké úrovni. Dobré bylo i rozhodnutí ponechat uživateli volnost v popisu vztahů mezi prvky. Domnívám se, že díky zcela zjevným přínosům se mi podaří systém APP prosadit.

Rýsuji se další možnosti rozvoje systému APP:

- generování struktury celého modelu,
- generování incidenční matice procesů a entit,
- implementace datové základny navrhovaného systému.

Bylo by možné zobrazovat i časové návaznosti procesů a programů, které jsou zprostředkovány daty. Ze vztahů procesů a entit resp programů a souborů lze vytvořit síťový diagram procesů resp programů. Po doplnění časových odhadů je pak možné odhadovat celkovou časovou náročnost a vytypovat kritická místa modelovaného systému. K zásadní změně celého návrhu však může dojít teprve při jeho implementaci na mikro počítači vyšší třídy.