

Václav Škarka, p.n.

Ústav výpočetní techniky Ostrava

POSTAVENÍ A VÝZNAM PODROBNÉHO NÁVRHU PROGRAMOVÉHO SYSTÉMU PRO REALIZACI PROVÁDĚCÍCH PROJEKTŮ AIS/ASŘ/

Existující metodické pokyny pro vytváření automatizovaných systémů řízení podniků často vedou spolu s neefektivní organizací práce ve výpočetním středisku ke stavu, kdy jediným podkladem pro vlastní programování je dokumentace technického projektu subsystému, resp. skupiny úloh. Uvedený stupeň dokumentace obsahuje podrobný popis chování navrženého systému vůči uživateli, jím je také obvykle schvalován. Chování systému ve vztahu k provozovateli jakož i struktura a vlastnosti budoucího programového systému jsou definovány v obecné rovině /pokud vůbec/ a vyjasňují se zdlouhavě a nekomplexně v průběhu programování, ověřování, zkušebního provozu, nebo i později. A tak průběžným předěláváním, upravováním a zpřesňováním konečné rutinní "tváře" vyvíjeného programového systému přicházejí o případné efekty, získané použitím zdokonalených programovacích technik.

Zařazením tvorby a schvalování podrobného návrhu programového systému do metodiky pro postup vytváření AIS /ASŘ/ vznikají předpoklady pro odstranění uvedených nedostatků.

V předkládaném sdělení se autor pokouší o popis a diskusi uvedeného okruhu problémů.

CO SE PO NÁS CHCE ?

Jsou-li mezi námi ještě programátoři, kteří jednoho krásného dne pozorně vyslechnou přání svého Uživatele, druhého dne napíší program, sami si jej vyděrují a odladí, třetího dne si dojdou pro vstupní data a odpoledne vlastnoručně odnesou usmívajícímu se Uživateli výstupní sestavy, aby se poté pohroužili do čtení osmdesátého pátého článku o strukturovaném /normovaném, úsporném, segmentovaném, .../ programování, tak těm je určena tato část příspěvku. Ostatní to už jistě znají ze své nelehké praxe.

Čím je charakteristická centralizovaná tvorba a provozování velkých programových systémů servisním způsobem pro více uživatelů? Vzhledem k značnému rozsahu řešené problematiky dochází k výrazné profesi specializaci. Na našem pracovišti jde o profesie projektant /informačního/ systému, analytik /ekvivalentní označení hlavní programátor, samostatný matematik-analytik/ a programátor. Projektanti jsou soustředěni v samostatném útvaru, analytici a programátoři jsou začleněni do několika výpočetních středisek, kde pracují ve skupinách podle jednotlivých problémových oblastí [5]. Vznikající programové vybavení je koncipováno pro vícenásobné použití u všech uživatelů v rámci výrobně-hospodářské jednotky. Programové systémy jsou rutinně provozovány ve více výpočetních střediscích pro několik uživatelů současně, bez možnosti kontaktu s autory. To vše klade specifické nároky nejenom na funkční cíle každého projektu, ale i řadu požadavků na jeho vlastnosti, zejména spolehlivost, adaptabilitu, udržitelnost, dostatečnou obecnost řešení, bezpečnost, aktuálnost dokumentace, odolnost systému [3].

JAK TO ZAJISTIT ?

Pro to, abychom dosáhli co nejvíce z požadovaných vlastností programového systému, je třeba vytvořit předpoklady ještě před zahájením jeho vlastní realizace /výroby/. Poté, co projektant dokončí a odsouhlasí rozhodující částí technického projektu řešené úlohy, zahajuje analytik řešení podrobného návrhu programového systému, který zajistí automatizované funkce projektu. Výsledkem je dokumentace, která obsahuje komplexní popis struktury programového systému a všech jeho komponent.

Pokud se pokusíme najít analogii s některým existujícím členěním procesu tvorby software, jde například u Mayerse [3] přibližně o činnosti: návrh architektury systému, návrh struktury programů, vnější návrh modulů a návrh datové základny. V metodických pokynech pro vytváření ASŘP lze při dobré vůli považovat za totéž fázi přípravy projektování v rámci vypracování prováděcího projektu.

Popis podrobného návrhu programového systému /dále jen návrhu/ je obvykle tvořen těmito částmi:

- členění programového systému na programové chody a jednotlivé programy /stylizované vývojové diagramy/,
- popis datové základny, struktura, obsah, rozsah a návaznosti datových souborů, jejich umístění na médiích,
- popis vstupních informací, vzory vstupních dokladů,
- popis výstupních informací, vzory výstupních sestav,
- zadávací listy /specifikace/ jednotlivých programů /modulů/,
- popis vstupních parametrů, způsob jejich zadávání,
- popis komunikace programového vybavení s obsluhou počítače,
- návrh systému automatizovaných /vnitřních/ i provozních /vnějších/ kontrol, případně způsobu zabezpečení nepovoleného přístupu k datům,

- nároky na obsazení periferních zařízení počítače,
- návrh na vytvoření procedur s ohledem na minimalizaci potřeby zásahů obsluhy počítače do průběhu automatizovaného zpracování.

Větší část uvedených informací se zaznamenává formou vyplnění předtištěných formulářů. Některé části se přeberou z technického projektu.

Programy, soubory a všechny další složky programového systému jsou v návrhu již označeny svým konečným jménem /přehledně v [2] /.

Ponechejme stranou formální stránku popisu jednotlivých částí návrhu, která z hlediska cíle tohoto sdělení není rozhodující. Je zřejmé, že musí splňovat požadavky na jednoznačnost, srozumitelnost a musí korespondovat s existující soustavou projektové dokumentace v organizaci.

CO TÍM LZE ZÍSKAT ?

Ptejme se po účelu a přínosech zpracování podrobného návrhu pro návazné práce na prováděcím projektu. V našem případě slouží písemně podchycený návrh k:

- Schválení, případně opanování návrhu mezi analytikem a projektantem. Cílem je:
 - * prověřit, zda jsou v návrhu dodrženy algoritmy, předepsané technickým projektem, a zda chování systému vůči uživateli odpovídá projektu,
 - * umožnit projektantovi získat dostatek informací /zpětná vazba/ o systému, který bude zabezpečovat automatizované funkce jím navrženého projektu,
 - * zpracovat podrobnou bilanci kapacit, potřebných pro realizaci programového vybavení, s ohledem na zde provedený odhad pracnosti a dostupné řešitelské i počítačové kapacity upřesnit plán vypracování prováděcího projektu, připravit podklady pro uzavření hospodářské smlouvy.

- Schválení, případně oponování návrhu mezi analytikem a zástupcem provozních útvarů výpočetních středisek. Jde zejména o posouzení, zda navrhovaný programový systém vyhovuje podmínkám rutinního provozu. Důraz se klade zejména na dobře navržené vstupní doklady, jednoduchost obsluhy systému a komunikace s ním, odpovídající kontrolní systém a dle povahy projektu i další. Přitom by měly být zkoumány hlavně ty složky návrhu, jejichž tvorba není dosud metodicky sjednocena.

Vzhledem ke komplexnosti předkládané dokumentace je možné ji vhodně využít i jako podklad pro strukturované posuzování v rámci řešitelské skupiny analytika, nebo k prověrce návrhu vedoucím útvaru [7].

Jak se zdá, alespoň z pohledu našeho pracoviště, mají všechny právě uvedené činnosti značný, ne-li rozhodující význam pro výslednou efektivnost prací na realizaci prováděcího projektu. Ve všech předcházejících fázích řešení byl převážně vynakládán pouze pracovní čas řešitelů, výsledkem dosud vynaložené práce je soustava dokumentace, jejíž obsah lze, zjednodušeně řečeno, přijmout, odmítnout nebo opravit. V následujících činnostech vejdou do hry obvykle značné finanční prostředky /velký tým programátorů, strojočas počítače, atd./.

Předpokládáme, že uživatelské aspekty projektu jsou již schváleny oponentací technického projektu s budoucími uživateli. Potom podrobný návrh programového systému musí být jednoznačným podkladem pro posouzení a schválení architektury systému a jeho chování vůči provozovateli. Podstatné změny struktury nebo provozních charakteristik systému lze provádět jen do okamžiku schválení návrhu výše uvedenými stranami.

Později - po zahájení programování - nelze tyto změny připouštět. Tento "stop-stav", dodržovaný od chvíle schválení návrhu až do ověření programového systému je namístě. Známe ze své praxe,

že právě dodatečné zásahy a úpravy některých složek systému během programování mohou způsobit zhoršení vlastností systému, vznik nekompatibilních složek a řadu problémů vůbec. Obvykle se totiž v tomto stadiu prací neprověří před vlastním provedením úpravy její možný dopad v kontextu celého systému. Jen na okraj poznamenejme, že totéž platí pro úpravy vnějších projevů projektu.

Zpracování diskutovaného návrhu se příznivě odráží do sjednocování metodiky programování. Spolu s účelnou organizací řešitelského kolektivu umožňuje účinné prosazování zvolené metodiky do každodenní praxe. Při tvorbě návrhu je možné a také výhodně používat více či méně prostředků z oblastí zdokonalených programovacích technik [7]. V první fázi návrhu jde o vývoj shora dolů spolu s uplatňováním požadavku na stavebnicovou strukturu systému [1]. V dalších fázích zpodrobňování návrhu lze uplatnit především strukturální diagram dat, strukturovaný návrh programu, rozhodovací tabulky, pseudokod a jistě i další.

Je velmi důležité, aby vypracování návrhu programového systému byl pověřen analytik s velkými odbornými zkušenostmi, obvykle vedoucí příslušné řešitelské skupiny analytiků a programátorů, který bude řídit i návazné práce. Pokud se nám daří získat tyto vedoucí skupin pro využívání zdokonalených programovacích technik a umožníme jim tyto metody dobře zvládnout, máme v nich záruku, že pro použití uvedených technik budou v souladu s potřebami projektu vytvářeny od počátku prací na programovém systému co nejlepší předpoklady. Čím detailnější bude zpracování návrhu programového systému, tím celistvějším dojmem bude působit výsledné programové vybavení. Například, jsou-li součástí diskutovaného materiálu i strukturované návrhy programů /modulů/, lze očekávat, že dokončený systém bude tvořen množinou dobře strukturovaných

programů, přestože to jsou díle různorodé skupiny programátorů /samozřejmě obeznámených s metodou strukturovaného programování/. Obdobný vliv má včasné vytvoření centrální knihovny s popisy datových souborů systému, nebo souběžně s tvorbou návrhu odladit moduly pro realizaci často se opakujících funkcí systému. Analytik může /a musí/ ovlivnit při tvorbě podrobného návrhu programového systému všechny rozhodující složky systému a prosadit uplatnění všech platných /a smysluplných/ standardů.

Postup, popisovaný v tomto sdělení, vytváří vhodné předpoklady pro potřebnou, co nejširší, dělbou práce při kodování, ladění a ověřování. Schválený návrh obsahuje všechny potřebné podklady pro provedení těchto prací. Jednotlivé programy, nebo lépe ucelené programové chody jsou přidělovány v první řadě programátorům z příslušné problémově orientované řešitelské skupiny [5]. Vyžaduje-li to situace /a to bývá ve vytiženém útvaru často/ je možné zapojit do těchto prací všechny programátory, kteří jsou v daném období k dispozici, tedy pružně přeskupovat programátorské kapacity.

Již jsme se zmínili o tom, že podrobný návrh programového systému je vhodným podkladem pro bilanci potřebných kapacit /pracovní hodiny, strojočas/, vycházíme z něj také při sestavení harmonogramu, nebo síťového diagramu prací na prováděcím projektu.

Zařazení tvorby podrobného návrhu mezi vypracování technického a prováděcího projektu má příznivý vliv na zvýšení úrovně provozní a programové dokumentace. Návrh totiž obsahuje podstatnou část této dokumentace. Je v zájmu analytika /autora/, aby schvalovaná dokumentace návrhu byla úplná a bez chyb, a je v zájmu programátora, aby nalezené nedostatky v zadání byly analytikem vysvětleny a do zadání opraveny.

To, co jsme popisovali, lze označit za "předvýrobní přípravu" programování /v širším slova smyslu/. Do kapacitně i finančně náročné výroby se pouštíme teprve v situaci, kdy existují přesné popisy struktury a požadovaných funkcí "výrobku", kdy jsme schopni odhadnout s dostatečnou přesností kolik nás jeho zhotovení bude stát a kdy budeme s dostupnými kapacitami schopni "výrobek" dokončit. Význam všeho se zvětšuje spolu s rostoucí složitostí a rozsahem realizovaného programového vybavení. U malých projektů, kde problém vyřeší několik jednoduchých /příp. parametrických/ programů, se zpracování podrobného návrhu vhodně zjednoduší. U jednorázových a operativních úkolů odpadne někdy zcela.

Formální stránka zpracování podrobného návrhu a jeho začlenění do řetězce prací při vytváření automatizovaných systémů nemusí být v různých organizacích stejné a je zřejmé poplatné organizačním podmínkám i povaze řešených projektů. My sami jsme nejvíce poučili ve výpočetním středisku Ingstav, n. p. Brno, kde k mnohému uvedenému dospěli již dříve. K popisované metodice jsme dospěli postupně, tak jak jsme museli na vlastním pracovišti řešit vznikající nedostatky v koordinaci a kvalitě práce, i s ohledem na potřebu získat objektivní podklady pro plánování.

Srovnání s metodikou, používanou ve srovnatelných výpočetních střediscích, nás přesvědčuje, že by mohlo jít o životaschopné řešení. Například ve výpočetních střediscích PVT [4] se dospělo v podobných podmínkách bez vzájemné koordinace k obdobnému řešení /označováno jako analytický projekt/.

Všechny "zaručené" techniky z oblasti programování mají svá "ale", je tomu tak i zde. Návrh vyvíjejí analytici - pracovníci, kteří musí splňovat nároky na vysokou odbornou úroveň a jakých není v žádném řešitelském kolektivu dost.

Jejich kapacitami je v podstatě limitována "průchodnost" útvaru analýzy a programování, pokud jde o počet současně realizovaných projektů. Řešení je v účelném plánování. Problémy mohou vzniknout při zajištění kvalitního posouzení a schválení podrobného návrhu ze strany projektanta a zástupce provozu. Musí být snahou všech nalézt co nejlepší a úplné řešení, nepřehlížet nedořešené složky systému. To lze ovlivnit účelným stanovením odpovědnosti pracovníků a vytvořením potřebné motivace.

Dosavadní praktické zkušenosti potvrzují oprávněnost zvoleného postupu. Vzhledem k danému rozsahu příspěvku se autor pokusil pouze naznačit hlavní otázky, související s problematikou. Nevyhnul se přitom terminologickým nejasnostem, zejména pokud jde o slovní označení diskutovaného návrhu. Svoji činnost při rozvíjení metodiky v popisované oblasti proto nemůžeme považovat za ukončenou. Přivítáme, pokud toto sdělení vyvolá další diskusi a případnou spolupráci.

LITERATURA:

- 1 BĚBR R.: Programové řešení typových projektů pro různorodé uživatele, sborník semináře "Metody programování počítačů 3. generace", DT ČVTS Ostrava, 1976
- 2 ČEVELA V.: Tvorba jmen v analytické a programové dokumentaci, sborník semináře "Programování '80", DT ČVTS Ostrava, 1980
- 3 ČIMBURA V., TVRDÍK J.: Problémy návrhu modulárních programů, sborník semináře "Programování '80", DT ČVTS Ostrava, 1980
- 4 Kolektiv: Sborník semináře "Projekční a programovací řád výpočetních středisek", DT ČVTS Praha, 1978
- 5 LACKO B.: Technologicky nebo problemově uspořádaný tým?, sborník semináře "Programování '80", DT ČVTS Ostrava, 1980
- 6 MAŠEK V., ŠKARKA V.: Aktualita programování č. 13, Systém práce s uživatelským software VS Ostrava, interní materiál ÚVT Ostrava, 1979
- 7 METZL K., TVRDÍK J.: Zdokonalené programovací techniky, sborník semináře "Programování '79", DT ČVTS Ostrava, 1979
- 8 ŠKARKA V.: Zásady řízení analyticko-programovacích prací ve VS Ostrava Ústavu výpočetní techniky, interní materiál, ÚVT Ostrava, 1979