Karol BARON, prom.mat.
VÚEPK Ostrava

PROGRAMÁTOR NA PŘECHODU MEZI 2. a 3. GENERACÍ

Někde jsem četl, že v roce 2000 budou existovat jen dvě profese : uklízečky a programátoři. Tento citát se mi vybavuje vždy, když řeším problémy, jež se právě zdají neřešitelnými. Tu si říkám, že s ohledem na onen citát jsem si vybral správné povolání a jde už jen o to, jakmile těch 25 let vydržet. Jsem povinen se distancovat od názorů citovaného autora a přiznat se, že jsem citátu použil jen k zaujetí úzkého okruhu posluchačů, jimž patří tento příspěvek a pro přechod k formulaci první diskutované otázky.

Je programátorství posláním ?

Pokusím se přesněji definovat smysl otázky a předem se omluvím, že nepoužiji žádné dostatečně formalizované, složité a efektní formy definice, ale lidské definice příkladem.

Stane-li se člověk ve 30 letech vědcem, může růst a tvořit až do sedmdesátí věkem a dokonce i potom. Může vědě zasvětit život. Totéž se říká o učitelích, inženýrech, zlepšovatelích, spisovatelích, obchodnících, funkcionářích, lékařích, rybářích a zahrádkářích. Mluvím o podmnožině lidí, kteří své celoživotní dílo konají s láskou k němu a právě proto ho konají celý život, jsou šťastní a spokojení. Pak si myslím, že jejich profese, práce nebo koníček jsou posláním.

Ptejme se tedy znovu. Může člověk být programátorem celý život, konat tuto práci tak, aby byl společnosti maximálně užitečný a byl přitom šťastný, svou práci měl rád a netoužil po něčem jiném ? Není programátorství jen něco mezi děrovačkou a analytikem ? Dělá dobře dobrý programátor proto, aby byl lepším programátorem, nebo proto, aby jím

přestal být ? Jsem-li programátorem, mám být na to hrdý nebo se stydět ?

Přesto, že jsem programátorem již 15 let a bez ohledu na krátkou historii využití počítačů v ČSSR si neosvojuji právo na tvrzení ze zkušeností, ale pokusíme se kladnou odpověď na položenou otázku dokazovat.

Myslím, že podmnožině lidí o nichž je zde řeč přináší pocit štěstí a radosti z konané práce stav dynamické rovnováhy mezi společenskou užitečností a tvořivým, mnohotvárným charakterem činností vyžadujících maximálního duševního vypětí v kritických okamžicích řešení a konec konců pak stále nový neopakující se charakter řešených problémů.

Je nesporné, že velmi rychlé inovace technického vybavení vytvářejí předpoklad takové tvořivé práce. Myslím však, že okruh řešených problémů a způsob jejich řešení není pro charakter programátorských prací zanedbatelný.

Jsem přesvědčen, že práce v továrně na programy, jak klidně můžeme nazvat řadu výpočetních středisek s druhogenerační technikou, nepřináší tomu, kdo pracuje dále ony pocity na něž má právo a je pochopitelné, že po 5-8 letech touží dělat něco jiného nebo aspoň jinde. Přeneseme se však zpět o těch 7-8 let, kdy se v Ostravě objevily první druhogenerační počítače. Jaký to byl pokrok, jaké vzrušení nám připravily první odladěné programy v FORTRANU nebo COBOLU, nám, kteří jsme až dosud pracovali v různých strojových kódech nebo autokodech a polovinu mládí jsme věnovali tvorbě vlastních čtecích a třídících programů pro magnetické pásky, což se pro nás přes noc stalo záležitostí několika parametrických štítků.

Vezměme si poučení z onéch časové etapy první počítačové inovace mezi I. a II. generací. Někteří přešli v prvních

okamžicích a docela. Ti prožili to nové objevování, hledání, učení i neúspěchy. Jiní zůstali a buď přestali být po jisté době programátory a "postoupili", nebo přešli k hotovému - a ti ale přišli o "lahůdky".

Druhá generace přinesla i jiný způsob práce a jiné úkoly často nazývané agendovým zpracováním dat. Tedy nejen formální, ale i obsahové změny. Dnes druhá generace dožívá a znovu vedle sebe stojí dva ročníky počítačů. Zdá se však jakoby ony lahůdky, ony pionýrské problémy byly proti minulé inovaci značně oslabeny. Tento stav považuji za normální, protože došlo jen k technické inovaci, ale obsah a způsob práce zůstal nezměněn. Továrny na potištěný papír už nelze zastavit. Každá vyšší technika musí umět to, co uměla nižší. To nové, vzrušující, objevné a pionýrské nelze tedy hledat tam, kde 3.generace nahradila druhou, ale tam, kde 3.generace teprve hledá uplatnění, kde je novou stránkou aplikace počítačů a to je bezesporu v oblasti řízení výrobních procesů v reálném čase. Tu je možnost začít znovu, tvořivě uplatnit zkušenosti, učit se a objevovat nové. Tu lze zůstat programátorem, netoužit po postupu do jiné funkce. Je ovšem nejvyšší čas. "Lahůdky" se právě rozdělují. Pokusím se dále vyjmenovat některé problémy, které rozhodnutí přejít na takovou profesi před 2 roky přinesla.

Považuji za správné ještě přesněji specifikovat prostředí jímž jsou zde uváděné názory ovlivněny. Zhruba před dvěma lety bylo rozhodnuto zasturovat státní úkol P-04-135-018 "Automatizační a řídící systém hlubinného dolu" řídícím počítačem RPP 16, jehož duchovním otcem je ÚTK SAV reprezentovaný zodpovědným řešitelem doc.Plandorem. Výrobcem počítače je TESLA Orava, dodávající organizací ÚVT Tesla. V lednu 1975 byla pro potřeby státního úkolu dodána základní konfigurace ZJ, 16 K

paměti a referenční periferie, která byla 16.1.75 oživena. Během roku 1975 a 76 bude konfigurace postupně rozšířena na 64 K paměti (maximum), magnetické pásky (4), disky (3), jednotka styku s prostředím, jednotka přerušovacích signálů, displeye, dálnopisy.

Pod vedením VÚEPE byla koncipována skupina 16 programátorů tří organizací kooperujících na řešení státního úkolu ZAM, Stařič a VÚEPE. Tento kolektiv zahájil práci prakticky v okamžiku, kdy technické vybavení RPP-16 bylo předáno výzkumem k tovární výrobě a dnes hotové programové vybavení teprve začínalo vznikat. Pracovali jsme na všech postupně oživených kusech RPP-16 počínaje laboratorně postaveným procesorem v ÚTK SAV, přes funkční vzorky VVÚ a prototypy z Trenčína a TESLY Orava po první továrně vyrobené kusy.

Prvním problémem byl výběr vhodného kolektivu. Dnes už můžeme některé zkušenosti zobecnit natolik, že dovedeme formulovat dva základní aspekty, které ovlivňují vhodnost výběru kádrů.

1. Faktor dosavadní praxe ukazuje na dvě skupiny vhodných pracovníků: a) programátoři, kteří začínali nebo aspoň jistou dobu pracovali u malého počítače ve strojovém kódu nebo málo dokonalém vyšším jazyku typu autokód. Zvláště takoví, kterým minulý vývoj výpočetní techniky vzal v posledních letechbpřímý kontakt s počítačem prožívají návrat k hardware, možnost přímého ladění u stroje s velmi příjemnými pocity.

   b) Úplní začátečníci pro něž je první bezchybná posloupnost instrukcí působící vytištění nebo přečtení několika znaků osobním vítězstvím ducha nad technikou.

Z tohoto pohledu za nevhodné pracovníky možno počítat zkušené programátorské kádry vzdělané a vychované na vyšších programovacích jazycích. Totem hodnocení chápu spíš z jejich vnitřních pocitů a radosti z vykonané práce. Podstatně menší účinnost jimi psaných instrukcí a příkazů spolu s obtížemi ladění a hledání chyb jim již nikdy nedovolí vážit si řešených úkolů a být na ně hrdý.

2. Faktor ochoty a schopnosti proniknout hluboko do řešeného problému. Řešíme-li řízení výrobního procesu je tento problém otázkou, která stojí nad jednotlivcem a týká se celého kolektivu, který problém řeší. Úspěšné řešení problémů je podmíněno, aby programátorský kolektiv byl značně heterogenní co do specializací a vzdělání jeho jednotlivých členů. Je třeba programátorů-inženýrů profese jejichž řízení řešíme, elektroinženýrů, ekonomů, i klasických matematiků a systémových inženýrů. To z hlediska vzdělání, ale zároveň u všech členů kolektivů je potřebná snaha a schopnost se do nejzažších mezí chtít pochopit řešený problém. Stručně řečeno je potřeba negace čistého programátora - kódaře.

K orientaci zde uvádím v tabulce i některé informace o struktuře našeho kolektivu, který považujeme za vhodný pro řešení daného úkolu.

Tabulka 1. Struktura kolektivu:

Střední věk:	31,3
Ženy - počet:	5
Muži - počet:	13

Vzdělání - profese

syst.inženýr	1
ekonom-inženýr	2
elektro inženýr	3
profesní inženýr	4
matematik	6
středoškolák	2

Praxe u počítačů před přechodem na RPP 16

8 - 15 let	4
3 - 8 let	8
0 - 2 let	2
0 let	12

Velmi cenné zkušenosti jsme získali při tvorbě kolektivu a jeho přípravou na nový způsob práce. Podařilo se nám zajistit účast jednoho zkušeného pracovníka na prvním informačním školení pracovníků ÚVT Tesla. Školení prováděli autoři systému RPP-16 z ÚTK SAV. Teprve pak byl celý rodící se kolektiv (tehdy 8 pracovníků) vyslán na programátorské školení spolu s proškoleným pracovníkem. Tento postup podstatně zvýšil účinnost programátorského školení, což jsme si ověřili při příštích školeních, jichž se zúčastnili jednotliví pracovníci, kteří byli přijímáni později. Ihned po školení se přecházelo na řešení praktických problémů v semenáři.

Dvouleté období naší práce bylo ve znamení stálého odkládání dodávky vlastního počítače, prací na řadě počítačů v nejrůznějších podmínkách, prakticky bez jakéhokoliv pomocného personálu a někdy i bez pomocného technického vybavení, takže opravy programů ručním děrovátkem se pro nás staly běžnou rutinou.

Přitom bylo nutné zajišťovat standardizaci programovacích prací na řešeném úkolu a seznámování s dokončovanými programovanými systémy, ať už vlastními nebo získanými z jiných pracovišť. Všechna další školení a seznámení jsme již dělali vlastními silami v 2-3 denních seminářích nebo operativně svolávaných schůzkách. Získané programy ověřoval vždy jeden pracovník, který zpracoval pro vlastní vnitřní potřeby

přístupný materiál a sám o něm referoval a odpovídal na řadu otázek ostatních. Tak byly zvládnuty v kolektivu 3 operační systémy RTOS, MOS, MULOS, standardní aritmetika, knihovna funkcí, RPP-FORTRAN, ladící systém a jiné. V neposlední řadě má dobrý vliv na informovanost kolektivu zásada, že všechny ukončovací práce se opomají v programátorském kolektivu, což na druhé straně zvýšilo úroveň oponentur.

Realizovat v praxi řídící systém pracující v reálném čase v režimu on-line je značně komplikovaný problém. Počítač a jeho fungující programové vybavení je jenom jedním z dílčích úkolů navazující na složitou problematiku ostatních problémů. Bylo nám od začátku jasné, že po dodávce počítače bude následovat několik etap v nichž bude režim práce počítače a tedy i programové vybavení jiné. Komplikace v našich podmínkách nastávají ještě nemožností kompletní dodávky konfigurace hlavně pokud jde o vnitřní paměť, vnější hromadné paměti a v neposlední řadě zpožďování v dodávce operačních systémů. Po oživení počítače je nutné počítat s dosti dlouhým obdobím pro připojení, ověření a oživení vstupní strany, zajišťující spojení počítače s výrobním procesem. Odborníci uvádějí, že toto období trvá 4 - 5 let.

Proto jsme rozhodli, že v této fázi budeme užívat počítač klasicky. Měli jsme k tomu předpoklady, protože autoři dvou podsystémů vyvinuli své technické vybavení tak, aby produkovalo děrnou pásku. Zpracovali jsme potřebné programy, které nepotřebují pro svou funkci nic než počítač a sebrnam. Říkáme jim programy pro prázdný počítač. Ukázalo se, že tento postup byl naprosto správný. Dnes po dodávce počítače v konfiguraci, která nedovoluje přímé připojení vstupů z výrobního procesu jsme byli schopni okamžitě zahájit praktické zkoušky těchto podsystémů, i když v režimu off-line.

V druhém podsystému, jsme se rozhodli postupovat jinou cestou. Ač jsme nevěděli jaký operační systém budeme užívat, řešili jsme problém pro provoz on-line. Velmi důsledně jsme oddělovali problémy s operačním systémem související od ostatních. Tyto problémy jsme naprogramovali simulačně pro použití v nejjednodušší konfiguraci bez oper.systému a dali k všeobecnému užívání ve formě standardních podprogramů. Programátoři mohli pracovat tak jako by měli k dispozici definitivní konfiguraci s operačním systémem o jehož vlastnosti se nemuseli vůbec zajímat. Výhodou takového řešení bylo, že celkovou funkci programovaného podsystému bylo možno odladit a nyní po dodávce počítače i provozovat jako off-line podsystém. Pro zařazení pod operační systém je nutné předělat jen ony standardní podprogramy.

Takové řešení přineslo možnost spojitého přechodu od klasického režimu k přímému napojení na výrobní proces. Budeme off-line podsystémy provozovat i po dobu ladění přímého zapojení a teprve po dokonalém ověření on-line režimů původní způsob práce opustíme. I potom bude dnes provozovaný způsob práce sloužit jako rezerva pro poruchy systému.

Dnes jsme ve stavu, kdy již máme zvládnutý počítač v režimu off-line, fungují nám programy, které jsme odladili na všech možných dostupných počítačích a přecházíme do etapy práce pod operačním systémem. Vystupuje tu před nás mnoho nových problémů jako :

- naše programy budou provozovány v multiprogramovém režimu velmi náročném na preciznost funkce programů,
- dělíme se o vstupní i výstupní periferie v čase i prostoru s mnoha jinými uživateli,
- naše služební standardní programy musí být reentrantní, to znamená současně použitelné více uživateli,

- námi zpracovávaná data mohou i jiným programům a obráceně
- naše programy musí být volně přemístitelné v paměti
- sestavujeme programy pro řízení výrobního procesu, kde se chyba nedá odstranit restartem a novým chodem, ale může znamenat katastrofu.

Vyjmenoval jsem problémy, o nichž víme, že se vyskytnou. V cestě stojí řada otazníků, které jsme ještě nevyzkoušeli jako :
Dynamická řádění multiprogramového systému. Otázka jak máme reagovat, když izolovaně odladěné programy budou při současném spuštění havarovat.
Budeme mít odvahu jakkoliv dále rozšiřovat první fungující programový systém, který bude skutečně zapojený na výrobní proces ?

Nechci srovnávat naše problémy s tím, co může programátor v klasickém výpočetním středisku nebo závodu výpočetní techniky od své práce očekávat. Chci, aby si každý udělal své závěry. Proto chci jen zdůraznit, že v dnes klasické oblasti využití výpočetní techniky - v oblasti hromadného zpracování dat přechod na třetí generaci znamená pro programátora jen další vzdálení od počítače, zjednodušení programovacích činností a tím zvýšení jeho kapacity a mamalizaci práce zase většině počtu pracovníků.

Aplikace rychlých procesorů na přímé řízení výroby naopak vrací programátora zpět k procesoru a ještě více, zatahuje ho blíže k výrobním procesům a přináší mu nové dosud nevyřešené problémy.