

Pojmy a dojmy

Rok 1990: Klec právě spadla. Konečně je dovoleno zamýšlení nad skutečnými problémy (kterých je víc než dost). Pokoušíme se definovat své místo ve společnosti i ve světě.

Dojmy jsou laciné efektní chiméry, které někteří nekalifikovaní a nekompetentní lidé vyvozují z obecně platných pojmů. Prosazování realizace takových dojmů může mít přinejmenším zlé důsledky ekonomické, provozní i společenské. Prostě je nebezpečné plést si pojmy s dojmy.

Pro ilustraci toho, o čem budeme uvažovat si uvedeme příklad z doby nedávné. Osoby a obsazení:

V - vedoucí: vysokou školu absolvoval přes výkonnostní volejbal, odpovědného místa dosáhl pomocí včas získané červené knížky (pozn.: před nedávnem ji zahodil a nahradil trikolourou a libvým odznakem, takže dále řídí, kvete a bují).

P - výkonná, solidní, pečlivá a spolehlivá pracovnice, která na rozdíl od vedoucího umí samostatně myslet (a je tudíž celkově mírně nežádoucí).

Úryvek dialogu z výrobní porady:

V: Je třeba okamžitě vymstitit a odprodat náš zastaralý počítač EC 1057 a nahradit jej sítí PC!

P: Máme na ty PC peníze?

V: Koupíme moderní čs. stroje Didaktik Gama za levnou cenu!

P: Bez winchesterů a bez tiskáren?

V: S tím už si musí poradit naši programátoři. Společnost jim umožnila vzdělání, tak ať teď ukáží, co se naučili!

P: A co bude s hotovými agendami, kterých jedeme poměrně dost?

V: Jste detailistka a kverulantka, narušujete svými výpady smělé perspektivy naší práce! Stávající agendy prostě přepíšete pro PC pracovníci provozu po pracovní době. A nebo se to udělá nějak jinak.

P: Myslím, pane inženýre, že si pletete pojmy s dojmy.

V: Úkoly jsou vám všem jasné, končím radu.

Všichni víme, že takřka všude existují borci, kteří pochytí z novin nebo jiných masmédií pár chimerických nápadů a ke své větší slávě se je snaží bez ohledu na důsledky prosadit v praxi. A nedělejte si iluze, že naše soft-revoluce nějak zabránila další katastrofické činnosti těchto dovedů a rejdičků. Aby bylo jasno: nechci prosazovat přizemní realitu proti technickému pokroku. Podporuji však používání vzdělaného a zkušeného rozumu proti hurá-akcím, které by nám v neúprosných podmínkách tržního hospodářství tahaly z kapes naše výplaty.

Proto uvedu namátkou několik příkladů dosti rozšířených dojmů a pokusím se je konfrontovat s obecně platnými pojmy.

Systémy

Naprogramovali jsme u nás už spoustu různých systémů a určitá část z nich dokonce "chodí". Není tedy problémem programovat i systémy další, "na současnou úroveň". Analýza systému je přece jednoduchá: uživatel řekne, co potřebuje, my to nějak rozhodíme do souborů a už se může programovat.

Když takový výtvar prohlédne znalec, zjistí interesantní a často zertovná fakta: V evidenci pracovníků je i položka "barva očí", která se ovšem na žádné sestavě nevyskytne. Dále jsou ve větě "osobní záliby"; pečlivý analytik navrhl i výstup "přehled pracovníků dle zálib". Jelikož však není zajištěna aktualizace (žádný pracovník nepřijde sám od sebe ohlásit, že nechal ubraňení a věnuje se spíše slečnám), nemá výstup žádnou informační hodnotu.

To jsou sice poněkud extrémní případy, ale obecně je uchovávaní dat bez informačního významu značně rozšířeno.

Máme evidovat zakázky a pracovníky, kteří na nich pracují; uživatel žádá i evidenci měsíčních náběhů hodin na jednotlivé zakázky. Ve větě zakázky deklarujeme 12 položek pro měsíční náběhy, ve větě pracovníka odkazy na zakázky. Řada potenciálních informací (kolik udělal určitý pracovník na dané zakázce apod.) je "mrtvá" - nelze ji z taktu navržených souborů získat. Je ovšem tragické, že to celé se odehrává v relační databance, kde stačilo zřídit relaci "zakázka-pracovník-měsíc-hodiny" a ve "větě" zakázky si měsíční náběhy odpustit. Z těchto dat by bylo možno získat všechny potřebné (dokonce všechny možné) informace.

Máme tedy v systémech data, která prakticky nenesou informaci i data, která díky nevhodnému strukturování dávají mnohem méně informací, než by mohla.

Nestěštím teorie systémů je její zdánlivě jasný a jednoduchý základ: systém jsou prvky, spojené vazbami. Všichni vědí z tisku, že nebyť systémového přístupu, nestanuli lidé na Měsíc apod., takže i politické strany systémově zpracovávají své programy a systémově přistupují ke společenskému dění (výsledky bývají většinou tristní).

Každý jen trochu ambiciózní "odborník" uvede k jakémukoliv problému několik systémových aspektů a když je dovedně propojí s variacemi na téma "všechno se vyvíjí - všechno souvisí se vším - ve všem je rozpor" (dnes ovšem "demokratizace integruje realitu - trh stimuluje produkci - privatizace je bázi kvality") máme odborně fundovaný spis, projev či "materiál", začasté vhodný i k získání několika písmen za jméno. Školy pak vykládají systémový přístup jako sled mnoha kroků, které musí student znát nazpaměť a u zkoušky je bezchybně v daném pořadí odříkat.

Zatím jsme mnoho neslyšeli o tom, že dobrý analytik řeší tři okruhy vztahů v systému: člověk-stroj, stroj-stroj, člověk-člověk. Jedním z uznávaných dojmů je názor, že systémový analytik je prostý technik a do lidí mu tedy nic není. Je však ověřeným pojmem, že systém bez dobře vyřešených vztahů mezi lidmi (jako jeho prvky) je systém nedořešený a v praxi mívá havarijní sklony. Systémový analytik si má přibírat

k ruce odborníky na organizaci, psychology, lékaře, filozofy a sociology (i když poslední dvě profese jsme soustředěným úsilím prakticky zlikvidovali) a s jejich pomocí řešit různé vztahy v systému.

Móda a legrace

Utěšeně se šíří různé viry a jen co budeme mít nějaké ty síte, nastoupí i hackeři (kteří se probourávají ochranami systému a ničí data a programy; v lepším případě pouze zveřejní, jak zase převezl pány programátory nebo vám na disku nechají grafickou vitztku).

Neodborný a často i "odborný" tisk píše o těchto jevech s obdivem. Cetil jsem, jak geniální studentík celé noci oka nezamhouří, mobilizuje svůj neobyčejný důvtip a pomocí počítače typu Sinclair vnikne do bankovního systému, kdež si nechá na své konto připsat symbolických 1000 \$, neboť je tak ušlechtilý, že pohrdá miliónem.

Neustále se povídá a píše, jak je těžké a pracné vyrobít účinný virus a že jeho autor musí být výtečným programátorem. Uznávám, že je to pravda, ale odmítám se autorovi virů obdivovat. Řekneme si na rovinu, že tvorba virů je jen určitou formou vandalismu: jaký je rozdíl mezi zničením pracně vytvořených programů nebo informací a zničením sochy v parku? Sebegeniálnější hacker je obyčejný lump a terorista: jaký je rozdíl mezi tím, když se vbourám do systému (a něco tam ukradnu nebo poničím) a tím, když se vám vbourám do soukromého bytu? V čem se liší anonymní otravování počítačových systémů od anonymního otravování lidí telefonem?

Promiňte mě až nenáležité rozhořčení, ale slyším z různých stran konejšivé řeči o milé klukovské recesi a dokonce i o tom, že "ten strach z virů je teď v módě, ale nesmí se to přehánět". Vímě všichni, kde je kořen tohoto zlehčování a bagatelizování situace: kdybychom si skutečně nebezpečí virů a hackerů připustili, museli bychom veškerý software snad dokonce kupovat (!) a museli bychom určitý (nemalý) peněz investovat do

ochrany systémů. A tak je lépe zůstat u optimistických dojmů a nepřipouštět si k tělu reálné drsné pojmy.

Umělá inteligence

Po tristi úvaze o vandalech a teroristech něco veselého: problém umělé inteligence. Tady je pro dojmy a ležce široké a často neorané pole působnosti. Dokonce i technické zprávy (zejména výzkumné) s oblibou hovoří o řešení MTZ pomocí umělé inteligence, o programech pro evidenci základních prostředků s využitím heuristiky apod. Podívejme se na problém blíže:

Nejprve se věřilo panu Turingovi, že inteligentní stroj je takový, se kterým si můžeme popovídat. Na truc (a schválně) napsal pan Weizenbaum z MIT program ELIZA, který zcela splňuje Turingův požadavek inteligence a přitom je neuvěřitelně pitomý.

Vytvořily se tedy obory jako "rozpoznávání obrazů", "porozumění řeči" apod., přičemž vrcholem snažení je simulovat určitou schopnost člověka, ke které podle běžných pojmů žádná inteligence není potřeba. I blbec v cukrárně rozpozná dort, avšak inteligentní program s tím bude mít moc práce, zvláště bude-li dort šišaté ukrojen. O spotřebě času na rozpoznání dortu u člověka a u počítače raději nehovořím. Prostě se zjistilo, že současně programovací a technické prostředky nějaké inteligentní uvažování nezvládnou. Je to asi dobře; o tom, co by se stalo, kdyby nám počítače začaly opravdu myslet, existuje bohatá literatura v oblasti sci-fi (dva příklady naleznete i v této knížce).

Na druhé straně se však musíme rázně postavit proti dojmům, že práce na umělé inteligenci byly. Jsou a budou zbytečné a nesmyslné. Mnohé výsledky dnes běžně využíváme (např. v oblasti grafiky, optimalizace, některých simulačních úloh atd.).

I když dojem, že počítače nabudou značné inteligence je nereálný (dnes !!), musíme pojem "umělá inteligence" a jeho aplikace zevrubně studovat a v praxi využívat.

Je ovšem pozoruhodné - a odporuje to asi přírodním úkazům - že ac skutečná inteligence v umělé podobě neexistuje, podařilo

se nám už dávno realizovat negaci: všichni víme, kolik se v našich programech vyskytuje umělé blbosti!

Automatizace - nic jednoduššího

S příchodem PC vypukl nezřízený jásot v řadách počítačových hochštaplerů. Počítadla typu mainframe a mini už totiž přestala být exotickou hračkou a hrozilo nebezpečí, že někdo bude chtít, aby se na nich dělala pořádná práce. Oblíbeným trikem salonních systémových analytiků je tedy okamžitý výprodej stávající "velké" techniky (pokud možno do šrotu) s tím, že se časem porídí 80 až 140 personálů. Když pak přízemní ekonom neuvolní potřebné množství peněz, zakoupí se dva výkonné čs. stroje PP 01. Programátorům, kteří se k nim neprozřetelně přiblíží, je uloženo urychleně na tomto vrcholu techniky zavést zpracování mezd. To se zpravidla nepodaří, mzdy se zadají do PVT a bystře se vyrobí program pro evidenci zlepšovacích návrhů, který je předváděn postupně všem vedoucím, publikován v tisku a popisován na seminářích. Tyž program zůstane po dlouhou dobu jedinou jakž takž chodící agendou na naši progresivní technice.

Nechci naříkat nad likvidací větších počítačů (statistiky prodeje ukazují, že zájem o tuto zastaralou techniku v civilizovaném světě neklesá), ale spíše nad tím, s jakými dojmy se leckde přistupuje k nasazení PC. Sotva jsme se naučili integrovat úlohy a využívat databanky, již trháme systém a jeho data na malé kousičky. Že existují sítě a distribuovaná data? Někde možná ano, ale když zjistíme, že u nás po velmi slušném spojovém okruhu přenášíme mizerný 1 MB prakticky 1 hodinu, přejde nás optimistický úsměv (pro silné povahy jsem spočítal, že po normální telefonní lince přenášíme tyž 1 MB více než 11 hodin).

Stejně tak opomíjíme zabezpečení systému. Na streamer nezbyly peníze; když si ověříme strašlivou pracnost pořízení kopie winchesteru na diskety, raději zamhouříme oči a využijme uvolněných vztahů v oblasti náboženství dáme na modlení a věříme, že to s pomocí Boží nějak dopadne. A to už nemluví o tom, že se nám po kancelářích povalují diskety (často ve funkci

podšálků) s důvěrnými informacemi, které si může kdokoliv kdykoliv opsat a zkopírovat pro svou potřebu.

Jazyky n-té generace

Výrobci dBASů a spreadsheetů (pravopis viz "Generální záprak") horlivě ujišťují, že s jejich produktem si kdokoliv snadno naprogramuje cokoli. Jistě: znám lajky-neprogramátory, kteří si pomocí FoxBase udělali dokonce telefonní seznam svých slečen!

Protože však se chytré systémy rozrostly do značných rozměrů, běžný lajk shlédnuv několik oběžných manuálů (kterými jest se mu prokousati, aby zvládl základy systému) zbáběle prchá. Profesionál tyto systémy uvítá, rychle se je naučí a již masí jeden program za druhým (většinou jako melouch), aniž by se zdržoval nějakými hlubšími analýzami. Aby to zas nevypadalo tak přízemně, prohlásí, že pracuje s jazykem čtvrté (ve společnosti obdivně vrkajících dam pak páté) generace, čímž zajistě povznesou svou odbornou pověst.

Děje se ale něco horšího: naše software-housy vedeny dojmem rychlých zisků bez investic počaly chrlit účelové programy dle objednávek (psané samozřejmě s použitím "jazyků n-té generace") a tak trochu zapoměly na vlastní výzkum a vývoj.

Je tedy naprostá většina našich uživatelských programů (pro PC) vytvořena pomocí produktů zámořského původu. Charakter a styl těchto prostředků se zřetelně promítá i do výsledných programů. Naskytá se otázka: je to tak dobře?

Není závažné, když program občas na uživatele bařne nějaké to "ADD NEW RECORDS (Y/N)" nebo "CANNOT OPEN FILE". Zajímavější je otázka pojetí zmíněného již vztahu "člověk-stroj". Např. systém s mnoha překrývajícími se okny, podle mého názoru, vyhovuje americké mentalitě, založené na jakémsi dryáčnictví a smyslu pro efektní show, kteréžto vlastnosti nejsou ale našemu uživateli příliš blízké. Spousta oken s rozsáhlými menu a ještě k tomu s "ikonami" Američana nadchne, kdežto náš uživatel se nad takovou obrazovku spíše zasmuší. Nebylo by dobře trochu se zamyslet nad mentalitou našeho uživatele a vytvořit takové produkty, které by jí

vyhovovaly?

Jako příklad z této sféry uvedu ještě geniální hru TETRIS: Američanům připadala příliš strohá a suchá i doplnil ji pestrými obrázky v klídkavém provedení, objevujícími se překvapivě bez jakéhokoli zřejmého smyslu; průzkumem jsem zjistil, že naši hráči dávají přednost původní jednoduché a elegantní ruské verzi. Poláci - jejichž mentalita je nám bližší než americká - převedli TETRIS do prostoru (!) bez lacině vnější okázalosti se sympatickou profesionální střídmostí; hráč není nijak rušen a může se soustředit na vskutku ďábelský obsah hry.

Suchý, střízlivý a profesionálně realistický mladý ekonom (např. má vlastní dcera) okamžitě rozmetá tyto romantické úvahy věcnou poznámkou, že na otevřeném softwarovém trhu nejsme schopni konkurovat americké razanci, podepřené špičkovými znalostmi a rutinovanou praxí; slovenská jemnost a něha bude převálcována nájezdem drsných cowboyů. V současném naruby obráceném světě přísluší optimismus a idealismus zřejmě pouze starcům; dovolte tedy autorovi vyslovit hříšnou myšlenku: až se přeženou různé bouře, náš Programátor (tak jako již mnohokrát v historii) ohromí svět a přinutí i softwarové giganty obdivně zmlknout.

Programování je pouze logika a rutina

Kdysi jsem se ozval proti dojmu, že "programátor potřebuje ke své práci exaktní logiku". K hlubšímu rozboru nyní použijeme ukázkou logiky z odborné literatury: "Vyslovíme větu: Tato věta je nepravdivá.

Jsou nyní dvě možnosti:

- a) předpokládáme-li, že věta je pravdivá, vyvodíme, že je nepravdivá, neboť to sama o sobě tvrdí; tento předpoklad je tedy vyvrácen
- b) je-li věta nepravdivá, vyplývá z ní, že je pravdivá a to je předpoklad "a", který byl vyvrácen dříve."

Každý praktik uzná, že uvedená úvaha je chytrá, elegantní a zajímavá, v programátorské praxi však stěžl nalezne uplatnění.

Řada matematiků slušně programuje; je to však hlavně tím, že využívají své matematické pečlivosti. Dobře programuje spousta lidí humanitního zaměření, u kterých by to málokdo čekal. Dobře programují i zarytí nepřátelé matematiky a logiky (např. moje již zmíněná dcera). Na druhé straně známe matematiky, logiky a exaktní pumpřdenty, kteří nedají dohromady tři řádky chodivého programu.

Dojem o exaktní logice nahradíme platnými pojmy: Programování je zcela specifickou kombinací fantazie a pečlivosti, tvůrčího rozletu a přezemního přání, neustálého dodržování a současného porušování řádu věcí. Programátor musí být v jedné osobě bohem a pedantský úředník. Kdo neumí soustavně střídat obrazotvornost, hodnou Isaaca Asimova s počítáním písmenek, řádek a sloupců není dobrým programátorem.

Bébrův zákon (uvedený v předchozím textu - viz "Na úrovni") to říká jasně: všechny přístupy, které užívá muž když si podmaňuje ženu a všechny přístupy, které užívá žena pro svedení muže jsou v programování přínosem. Tyto přístupy jsou - jak známo - založeny na imaginaci celkové koncepce a současně na pečlivém provedení detailů.

Tím považují za vyvrácené dojmy o exaktní logice v programování. Citlivým znalecům pak doporučuji zevrubně studium jemných rozdílů v programech, psaných programátorskými samci a samičkami.

Mnoho lidí také podléhá dojmu, že rutina = profesionalita. Rutinovaný číšník vám prostě přinese to, co jste chtěli, nezarmoutí vás, ale zůstává "mimo hru". Profesionál přidá v přesně odhadnuté chvíli tu úsměv, tu slovo, tu milé gesto; váš pobyt v hospodě povznese na velmi příjemný zážitek. Stejně tak profesionální programátor neupadá do studené rutiny; každý program řeší tak, aby zpříjemnil konkrétnímu uživateli jeho neblahou činnost.

Rutinér napíše program, jaký se po něm žádá. Profesionál dbá na své odborné jméno a snaží se dodržovat morálku profese; odmítá psát programy, falšující skutečnost, deformující informace, degradující data; odmítá vyrábět desivé kreatury jen proto, aby se někdo mohl vytahovat splněním nesmyslným plánem. Profesionál se snaží, aby se na něj vztahoval ten dávno

zapomenutý pojem "na slovo vzatý odborník". Trvá vždy na tom, že jeho podpis je zárukou kvality toho, pod co se podepsal.

A co psýcha?

Ač je to s podivem, bývá programátor často i člověk. Je tedy třeba studovat též jeho duševno (donedávna panoval obecně rozšířený dojem, že takové úvahy jsou k ničemu; tento dojem byl silně propagován, neboť co kdyby někoho napadlo zabývat se psýchou některých vůdčích osobností a "specialistů").

Často propadáme dojmu, že nezbytnými vlastnostmi programátora jsou chytrost a píle. Již Bismarck zavedl pro vojáky klasifikaci:

líný + chytrý = velitel
pilný + chytrý = štábní důstojník
líný + blbý = řadový voják
pilný + blbý = zastřelit za úsvitu

Uvedené členění můžeme snadno aplikovat na programátory. Nemusíme tedy roli chytrosti a píle absolutizovat.

Opomíjeným pojmem bývají i základní povahové rysy a mentalita programátora. Ukažme si na jednoduchém příkladu chování lidských typů. Příklad zapíšeme v prokletém jazyku BASIC (na který kdekdo nadává, ale všichni ho umíme):

V závislosti na hodnotě proměnné I má být vykonán určitý příkaz. Programátor - flegmatik nehne brvou a píše

```
10 ON I GOTO 20,30,40, ...  
20 bflm: GOTO 100  
30 hchkr: GOTO 100  
...
```

Sanguiník s lehkým srdcem napíše poněkud frivolně

```
10 IF = 1 THEN bflm  
20 IF = 2 THEN hchkr  
30 IF = 3 THEN mlfb ...
```

Naprotí tomu cholerik se na problém oboří

```
10 IF I>0 THEN IF I>1 THEN IF I>2 THEN ...
```

a po chvíli psaní narazí na vlnobítky ELSů, ve kterých se zcela ztratí; i zařve silným hlasem, roztrhává vše co dosud napsal a případně prokopne dveře u skříně.

Programátor - melancholik dosud nenapsal nic, neboť smutně dumá o tíži úkolu, který na něj dolehl.

Při ladění programů se povahy projeví ještě výrazněji: Flegmatik pečlivě fajkuje větve, které už prozkoumal a pomocí barevných dvanácti fixů značí do listingu opravy a úpravy.

Sanguinik se nesmírně baví tím, že tiskárnou proletěla jako blesk celá krabice papíru 1+2, aniž by se cokoliv vytisklo (omylem se mu do jakéhosi nulovacího cyklu dostal příkaz PAGE).

Melancholik sedí s hlavou v dlaních nad poněkud zmatenou sestavou a ujišťuje sám sebe, že tohle nemůže nikdy ladit.

Cholerik při výskytu první chyby zastaví stroj, vymaže, vynuluje a prokleje všechno, roztrhává listing, vytiskne nový, zjistí, že zapomněl o jakou chybu vlastně šlo a vyřítí se na chodbu, aby se uklidnil cigaretou. V půli cigarety na chybu přijde (v duchu), zahodí hořící půlku do krabice s makulaturou, spustí editor, opraví chybu, zapomene opravený source uložit, kompiluje a linkuje, při spuštění vyjede táž chyba a celé divadlo se několikrát opakuje; přitom divý muž stihne ještě uhasit vzniklý požár a založit nový.

Jak vidno, je užitečné zabývat se i pojmy z oblasti psychy programátorů, neboť tak lépe pochopíme jejich projevy a můžeme jim např. přidělovat práci podle individuálního naturelu.

Závěr

Je nebezpečné plést si pojmy s dojmy, jakož i dotýkat se funkcionářů i na zem spadlých.

Literatura

Zich. O.: Moderní logika. Orbis, 1958.
