Národní prostředí v softwarových systémech

Richard Bébr

Český telekomunikační úřad, Klimentská 27, 225 02 Praha 1, Česká republika

Abstrakt
Příspěvek pojednává o softwarových problémech národního jazykového prostředí. Zabývá se jazykovou kompatibilitou v různých národních prostředích, převody a překlady programových systémů i dat a specializovanou problematikou češtiny v systémech.

1. Úvod

V pravěku počítačů neexistoval jiný jazyk než angličtina. Český programátor se musel obejít bez hacku a carek. Tato ocesana cestina byla vsak kupodivu dobre srozumitelna. Nicméně v jiných národních prostředích speciální znaky místních abeced bolestně chyběly a texty v základní počítačové abecedě byly leckde takřka nesrozumitelné. I v Česku bylo jasné, že existují úlohy, kde se bez diakritiky neobejdeme. Proto již koncem šedesátých let byly prováděny pokusy se zaváděním české abecedy do některých agend. České znaky se tenkrát vyjadřovaly dvoubajtově se speciálně kódovanou diakritikou před nebo za písmenem (což někdy komplikovalo pořizování dat, i když tato metoda byla podobná psaní háčků a čárek u velkých písmen na psacím stroji).

Pokrok, business a uživatelé si posléze vynutili nejen používání národních abeced, ale i národního jazyka jak v konverzaci uživatele s počítačem tak i v textových datech. Zde se ukázal blahodárný vliv kapitalismu, neboť do roku 1990 se při aplikaci češtiny potkával uživatel s ďábelskými - z dnešního hlediska leckdy až žertovnými - obtížemi. Od uvedeného zlomového roku však postupovalo efektivní a elegantní zavádění češtiny (včetně výborných překladů cizích systémů) nesmírně rychle. Nejdůležitějším přínosem nového pojetí je - dle mého názoru - to, že zatímco dříve se musel o české prostředí a komplikované instalace lokalizovaných verzí starat sám uživatel, dnes je to výhradně věcí softwarových firem, které - chtějí-li na trhu obstát - musí dodat kompletně všechno v co nejpřitulnějším provedení. Samozřejmě existuje v této oblasti ještě řada problémů; na některé z nich poukazuje tento příspěvek.

2. Obecné úvahy

Tzv. "národní prostředí" (národní abeceda a národní jazyk) je dnes nezbytnou součástí všech softwarových systémů, neboť

- komunikace s počítačem v národním jazyku přibližuje počítač uživateli, dává mu zapomenout na to, že pracuje s jakousi necitlivou technikou a zejména usnadňuje jeho práci a omezuje chybovost,
- ve většině systémů je *naprosto nutné* ukládat data (jména, názvy, texty, ...) ve správné národní podobě (aby se vyloučily záměny a omyly) a dokonce zajišťovat sémantické hledání v textech s použitím ohýbání slov.

Dále se vynořil problém překladu a převodu programů a dat z jednoho jazyka do druhého. Světové firmy se tzv. "lokalizací" seriozně zabývají a hledají metody, jak napsat programový systém v jednom jazyce tak, aby byl snadno a s co nejmenším úsilím (a náklady) převeditelný do jiného jazyka.

3. Základní problémy

3.1 Hardware

O technických problémech pojednáme jen stručně, neboť jejich detailní analýza by si vyžádala samostatný spis.

Všechna výstupní zařízení (displeje, tiskárny, plottery apod.) musejí umět zobrazit znaky národní abecedy ve velkých i malých písmenech i v různých velikostech a typech písem. Tento problém je v současnosti celkem dobře vyřešen.

Vstupní zařízení (klávesnice, snímače, scannery apod.) musejí národní abecedu zvládnout v plném rozsahu, aniž by omezila interpretaci standardních mezinárodních speciálních znaků (např. @ # $ ^ & *). Určitý problém zde představuje volba klávesnice. Pro češtinu se již vžilo uspořádání českého psacího stroje (v horní řádce ě š č ř atd. bez shiftu, číslice se shiftem). Interpretace speciálních znaků včetně % \ ( ) { } [ ] < > však není všude důsledně sjednocena. Osvědčuje se softwarové ošetření "druhého shiftu", kdy pomocí kombinace kláves (CTRL SHIFT, ALT ... apod.) přecházíme na speciální znaky mezinárodní klávesnice. Má být ošetřen i přepínatelný režim QWERTY a QWERTZ. Je tedy důležité používat takové fyzické uspořádání klávesnice, které má na klávesách uvedeny všechny možné znaky s rozlišením (např. barevným) národní a standardní světové množiny znaků. Takové klávesnice jsou k dostání a je pouze třeba

- zajistit dobrou a obsažnou klávesnici s čitelným a přehledným označením kláves,
- zvolit vhodné softwarové ošetření všech potřebných (i zdánlivě nepotřebných) znaků.

Mělo by být zákonem, aby všechna pracoviště téhož podniku byla vybavena stejnou klávesnicí a stejným softwarovým řešením přístupu ke speciálním znakům. Jen tak předejdeme nadměrné chybovosti a zajistíme zastupitelnost pracovníků.

3.2 Kódování

V ČR jsou používány různé kódové soustavy pro vyjádření národních znaků. Můžeme jmenovat kód bratří Kamenických, Latin2, ISOLatin2, KOI8 atd. V novějších systémech (nové verze MS DOS, WINDOWS, ...) jsou národní kódové stránky přímo zabudovány. S různým kódováním se musejí vyrovnat především veřejně dostupné informační systémy, se kterými pracují různí uživatelé s různými kódovými soustavami.

Všechny užívané kódové tabulky národních abeced mají určitou nevýhodu v tom, že oblasti určené pro národní znaky se u různých národních abeced překrývají a nemůžeme tedy současně pracovat s různými množinami znaků různých abeced. Tuto nevýhodu se snaží odstranit UNICODE, který je dvoubytový (pro každý znak dva byty) a který zajišťuje pro každý znak jakékoliv evropské a podobné abecedy unikátní kódování. Nutno upozornit, že vývoj systémů, obhospodařujících současně různé abecedy je v počátcích, při sílící mezinárodní výměně dat se však stává vysoce aktuálním.

3.3 Zvláštnosti češtiny

Při práci s češtinou je nutno ošetřit i třídění. Norma pro řazení českých slov je dosti komplikovaná a jen některé současné softwarové produkty ji přesně splňují. V systémech, kde je na řazení slov kladen velký důraz, je třeba kontrolovat a případně revidovat dodavatelem navržené třídění.

Dále je - zvláště v plnotextových systémech - potřebné zabývat se i ohýbáním českých slov, které je důležité při vyhledávání v textech.

4. Jazyková kompatibilita

4.1 Multinárodní produkty

Tato kapitola se opírá převážně o lit. [1], kde je celá problematika zpracována přehledně a důkladně včetně obsáhlé bibliografie. Úloha zní: vytvořit programový produkt, který by bylo možno snadno a hospodárně převést do jakéhokoliv jazyka. Řešení této úlohy je v zájmu výrobců software, neboť - jak již bylo řečeno - dnešní programové produkty jsou ve světě prodejné jen s příslušným národním prostředím. Ukázalo se, že automatický překlad a převod do jinojazyčného prostředí pomocí počítače (strojový překlad) je vyloučen, neboť současná úroveň počítačových překladových programů je pro daný účel nevyhovující. Musí tedy být zajištěna týmová spolupráce programátorů se specialisty na cílové jazykové prostředí. Avšak již v samotném základním řešení programového produktu musejí být zakotveny určité zásady, které převod (do jakéhokoliv jazyka) umožní a usnadní.

Ideálem je jakýsi "jazykově prázdný" systém, který obecně a technicky plní všechny požadované funkce; pro implementaci do určitého národního prostředí postačí pouze doplnit veškeré texty (menu, nápovědy, hlášení, ...) v příslušném jazyce do určených textových souborů, aniž bychom jakkoliv zasahovali do vlastních programů. Vzhledem k problémům, na které dále poukážeme, bývá nutno doplnit texty určitými řídicími znaky, na které při interpretaci reagují příslušné programy.

4.2 Technologické problémy

Při tvorbě obecného multinárodního programového produktu musí brát programátor ohled na "technické" zvláštnosti různých jazyků, např.:

- Paměťový prostor: text (např. nápověda, HELP), který zabírá v určitém jazyku určitý paměťový prostor, může v jiném jazyku zabírat paměť větší nebo menší. Byla přijata zásada, že k paměťovému objemu pro anglický text se přidává rezerva cca 30 %, což

vystačí pro překlad do všech možných jazyků. Je zajímavé, že v "hladkém" souvislém textu je jedním ze *znakově nejúspornějších evropských jazyků čeština!*

- Některé jazyky (japonština) používají pro vyjádření jednoho psaného (zobrazeného) znaku 2 byty. Počet znaků tudíž není shodný s počtem bytů, takže např. dimensování oken nelze přímo přebírat z bytových čitačů (totéž platí pro výše zmíněný UNICODE).
- Běžně se užívá zápis a čtení zleva doprava, v některých jazycích se čte a píše zprava doleva a může nastat i případ, kdy v jednom textu jsou smíšené oba užívané způsoby (citace hebrejštiny v anglickém textu). Opět je nutno tuto skutečnost respektovat např. v čitačích. Zápisem shora dolů se zatím informační systémy nezabývají.

4.3 Syntaktické problémy

Důležité je národní formátování - např. datum, čas, měna. V různých národních prostředích se používá různých způsobů zápisu, který je nutno při převodu respektovat (přitom tyto položky bývají použity jako "proměnné", což situaci komplikuje - viz níže). Musíme ošetřit i používání desetinné tečky nebo čárky a značení tisíců a milionů. V časových údajích respektujeme 12 i 24 hodinový režim.

Často se vyskytují texty, v nichž některá slova jsou nahrazena proměnnými, jejichž obsah se určuje v okamžiku výstupu textu. Zde je nutno brát v úvahu slovosled v různých jazycích, který má leckdy závažný význam. Proměnná může být tedy při překladu umístěna na různých místech věty. Obrovské problémy zde přináší ohýbání slov, souvisejících s proměnnými. Ukážeme si to na příkladu české věty, oznamující počet nalezených výskytů (proměnná je uvedena v hranaté závorce):

Byl_ nalezen_ [1] výskyt_
Byly nalezeny [2] výskyty
Bylo nalezeno [5] výskytů apod.

(v angličtině jsou v tomto případě jen dvě verze).

4.4 Sémantické problémy

- Sexistika:
  Některé jazyky nejsou závislé na gramatickém rodu (angličtina), jiné však silně (čeština). Navíc díky hnutí šílených feministek se v některých zemích (USA) záměrně odbourává závislost na rodu (viz: "spokesman" nahrazen "spokesperson", "postman" nahrazen "postperson" apod.). Původní verze programového produktu v rodově nezávislém jazyku se musí vyrovnat s tím, že převod do jiného jazyka může být rodově závislý.

- Terminologie:
  V systémech zásadně omezujeme používání odborné počítačové terminologie. Moderní systém má vždy počítat s uživatelem - laikem v oboru počítačů a texty by se měly specializovaným termínům vyhýbat. Navíc leckdy nebývá národní terminologie ustálená (v češtině adresář = katalog = direktorář) a vhodný překlad se obtížně hledá. Jestliže však je počítačovou terminologii nutno někde použít, musíme počítat s tím, že v jiných jazycích nemusí odpovídající slova vůbec existovat (např. LOAD-UNLOAD, v ruštině ZAGRUZIŤ-VYGRUZIŤ nemá v češtině obdobu). Určité vžité pojmy můžeme přebírat bez překladu (pojem "OK" nebývá nutno do češtiny překládat), co však třeba se souslovím "laser printer down-load"?

Nevyhneme se ovšem odborné terminologii, používané v dané aplikaci (účetnictví, výroba, odbyt, zásobování, ...). Překlad termínů tohoto typu musí zpracovat kvalifikovaný specialista (znalý odborné terminologie a ustálených pojmů i vazeb ve zdrojovém i cílovém jazyku) včetně správné stylistiky a slovosledu.

- Zkratky a akronymy:
  Ty představují samostatný problém. Někdy se nepřekládají (IBM, COBOL, ...), někdy však je překlad nutný a zde pozor: jak třeba přeložit zkratku KWIC (key word in context)? Pokud v cílovém jazyku existuje užívaná a známá zkratka, máme vyhráno (ale v jiných jazycích tomu tak nemusí být). Obecný systém musí počítat s tím, že některá zkratka v jiném jazyku prostě neexistuje; pokud je to možné, ponecháme tedy textový prostor pro plné slovní vyjádření, pokud to možné není, musíme pro daný jazyk vytvořit (a zavést) novou zkratku, což je úkol velice obtížný.

  Běžně používané zkratky, které není nutno vysvětlovat mohou být obecné (USA), oborové (ERP v radiotechnice) a z oblasti výpočetní techniky (CD ROM). Pak ale existují účelové zkratky, používané jenom v dané úloze (často se vyskytují např. při popisu významu kláves v nápovědné řádce). Pozor na zkratky různých délek (angl. PG = německy S = česky STR). Každou jednotlivou zkratku a její použití musíme pro každý jazyk pečlivě analyzovat, aby nezpůsobila nejasnost nebo zkomolení významu.

  Často je každý byte drahý a tak např. anglické NXT PG (6 znaků) nepřekládáme DALŠÍ STR (9 znaků), ale spíše STR+. Zkušenost ukazuje, že běžný (nikoliv vzteklý, debilní nebo pumprdentní) uživatel si rychle zvykne, pokud má zkratka jasnou logiku.

  K problému zkratek v češtině se vrátíme v kap. 5.

4.5 Národnostní, kulturní a jiné problémy

Při tvorbě obecných, jazykově kompatibilních programových produktů mějme neustále na zřeteli, že překlady budou muset respektovat nejen jazykové, ale i národnostní, sociální a kulturní prostředí určitého národa. Zatímco americký uživatel bez problémů přijme strohý příkaz PRESS ENTER, pak pro Němce je lépe použít neosobního "je nutno stisknout ENTER", pro Čecha pak "stiskněte ENTER" (o tykání se zmíníme ještě v kap. 5).

Některé národy odmítají použití cizích slov a nechtějí je přebírat (Francouzi jsou zvlášť alergičtí na anglicizmy), některé národy přebírají cizí výrazy bez potíží, někde se používají cizojazyčná slova v národních tvarech (čeština: ohavné slovo "adventura" pro dobrodružnou hru, přijatelné "lejzrovka" nebo "displej", roztomilé "písičko") - u těchto lidových tvarů však doporučujeme maximální opatrnost. Slang se v počítačových systémech nepřipouští.

U arabských národů pozor na postavení ženy ve společnosti (středoevropsky přípustné ikony nebo loga s funkčně využitou ženskou postavou budou odmítnuty), totéž - avšak zcela opačně - platí pro dnešní feminismem zamořené USA (tytéž ikony nebo loga budou také odmítnuty, ale z důvodu ponižování ženy). Při použití grafické nebo textové symboliky je nutno hlídat, aby nepřipomínala nějaké náboženské nebo společensky vžité nepřípustné symboly v různých náboženských a etnických skupinách.

Některé národy, které se nedávno vymanily z otrocké závislosti na bývalém SSSR jsou citlivé na symboliku, která by jakkoli připomínala některý z dříve posvátných pojmů nebo obrazů socialismu. Přitom český uživatel bude takovou symbolikou obveselen, kdežto polský popuzen.

Potíže by mohly nastat při převodu do národního jazyka, který se ve většině národa neužívá (na Bílé Rusi se běloruštinou domluvíte málokde).

4.6 Obsluha jinojazyčných dat

Nebývá to sice pravidlem, ale programový systém, pracující v určitém jazyku, může zpracovávat data, uložená ve zcela jiném jazyku. Moduly pro práci s daty v takovém případě nemají být závislé na základním (komunikačním) jazyku systému. Pro každý řešený systém pak musíme zvolit jednu z těchto možností:

- systém je po lokalizaci pevně nastaven na jediný jazyk, užívaný v datech,
- uživatel má možnost volby jazyka pro práci s daty,
- v datových souborech je indikace použitého jazyka a systém sám volí vhodné procedury.

Otázka několikajazyčných dat v jediném systému začíná být aktuální (informační systémy pro evropskou legislativu a komunitární právo, mezinárodně propojené systémy s topografickými prvky apod.).

5. Čeština

5.1 Syntaxe - sémantika - lingvistika

Čeština má svébytnou syntaxi a sémantiku, pracuje s předponami a příponami, je silně závislá na rodu, má obtížnou tvorbu množného čísla a používá jevu, v jiných jazycích většinou neslýchaného - slovesných vidů. Proto je čeština v softwarových systémech jedním z nejobtížněji zvládnutelných jazyků.

Zvláště při sémantickém vyhledávání v textech (fulltext) nemůžeme žádat na uživateli, aby zadal všechny požadované tvary hledaného slova; ani hvězdičková konvence mnoho nepomůže (hledám-li texty o dolech, pak zadání DOL* sice najde DOLY, DOLŮ, DOLŮM, ..., ale nenajde DŮL a naopak najde DOLAR, DOLNÍ LHOTU, DOLIČNÝ PŘEDMĚT apod.). Musíme tedy zajistit, aby softwarový systém sám vytvořil všechny možné tvary zadaného slova (přitom např. od slovesa "kupovat" máme v češtině 220 tvarů). K tomu slouží lingvistické programové moduly a slovníky. Ucelený systém byl vypracován např. na Matematicko fyzikální fakultě UK v Praze, některé české softwarehousy mají vlastní lingvistiku. Uživatel si může většinou vybrat dva typy lingvistiky - úplnou a kmenovou. Úplná lingvistika generuje všechny tvary zadaného slova (nebo sousloví), kmenová určí jen možné kmeny a dosadí hvězdičkovou konvenci (pro zadání DOLY vytvoří dotaz na DOL* OR DŮL*).

Velkým problémem jsou cizí a odborné termíny. Slovníky pro českou lingvistiku pokrývají sice velký slovní rozsah, ale může se stát, že některý často užívaný termín chybí (např. TELEKOMUNIKACE) a tudíž chybějí i jeho gramatické tvary. Lingvistický modul by měl uživatele upozornit, že dané slovo nezná a nabídnout třeba hvězdičkovou konvenci.

Aktualizace a adaptace lingvistických slovníků by měla být běžnou praxí (dosud tomu tak nebývá).

Lingvistika se musí vyrovnat i s homonymy. Zadám-li slovo HNÁT, může to být podstatné jméno, označující končetinu kostlivce nebo sloveso s tvary ženu -ženeš -...-hnal-...-poženu-... kde tvar "ženu" je však také 4. pád podstatného jména "žena" atd. Různé lingvistiky se s touto problematikou různě vyrovnávají. V systémech, které lingvistiku používají je vždy nutné v příručkách a při školení uživatelů věnovat práci s lingvistikou samostatný oddíl (a upozornit např. uživatele, že nemůže hledat autorský zákon zadáním AUTOR, neboť to je podstatné jméno s tvary autora, autoru, ..., kdežto "autorský" je jméno přídavné se zcela jinými tvary).

5.2 Česká komunikace s počítačem

Problematika české komunikace s počítačem (menu, nápověda, hlášení, ...) je známá, učí se na školách a je zpracována i v literatuře. Snad tedy jen poznámka, že solidní softwarový systém má uživateli vykat (např. televizní a rozhlasové reklamy rychle od tykání upustily, neboť diváka/posluchače popuzovaly); je nepříjemné, čtu-li na obrazovce CHCEŠ NÁVOD? (ihned se mi vybaví CHCEŠ FACKU?).

Narazíme-li na gramatický rod a neznáme-li identifikovaného operátora/operátorku, volíme mužský rod; v Česku nebezpečí útoku feministek nehrozí, neboť ženy jsou zde dostatečně inteligentní a nad nepatřičným oslovením jen mávnou rukou. Máme-li však o obsluze dostatek informací (ke vstupnímu heslu bývá dobré přidat i rodné číslo operátora/operátorky - je to účelné z mnoha důvodů), pak můžeme gramatického rodu použít (u žen to má vždy velký úspěch).

5.3 České zkratky

I když je čeština jedním z nejúspornějších jazyků, zkratky se tvoří velmi obtížně. Anglická metoda vynechávání samohlásek je v češtině vyloučena - viz PRDJ = prodej, ZVTL = uživatel, PCTPRCVNK = počet pracovníků apod.). Přitom pro texty je na obrazovce vždy málo místa, zejména ve sloupcových hlavičkách nebo v nápovědné či stavové řádce. Způsob řešení problému zkratek záleží výhradně na řešiteli systému a vypovídá mnoho o jeho kvalitách.

6. Ostatní problémy

6.1 Kooperace

Jak z předchozího textu vyplývá, musí při tvorbě systému v národním prostředí spolupracovat analytik (programátor) se zkušeným a odborně zdatným lingvistou. Při návrhu multinárodních produktů musí kooperovat příslušní lingvisté pro všechny cílové jazyky. Zde se nevyplácí šetřit, neboť - jak již bylo řečeno - jazykový design softwarového systému je jedním z klíčových parametrů a závažně ovlivňuje prodejnost.

6.2 Překlady odborných textů

Textová data, která mají být překládána do určitého národního prostředí vyžadují zvláštní péči. Pokud jsou odborně zaměřena, bývá překlad až 10x dražší než běžný překlad třeba novinového článku. Např. data právního informačního systému vyžadují

- překlad lingvistou, znalým zdrojového i cílového jazyka,
- úpravu a opravu právníkem, znalým právnické terminologie a zvyklostí ve zdrojovém i cílovém jazyku,
- úpravu a opravu odborníkem, jehož profese se daný právní dokument dotýká (znalým profesní terminologie ve zdrojovém i cílovém jazyku).

Zvláštní pozornost věnujeme datům, ve kterých existují hypertextové vazby. Tyto vazby mohou být v některých případech jazykově nebo obecně národně závislé a je tedy nutná kontrola a případná úprava *všech hypertextových vazeb* před jejich přenesením do cílového textu.

6.3 Dokumentace a jiné problémy

S tvorbou a překlady softwarových systémů souvisí řada dalších problémů. Je to např. čokumetace, jejíž tvorba pro národní prostředí by si vyžádala samostatný článek. Dále je to např. způsob balení produktu, způsob instalace národní verze, distribuce produktu atd. Podrobnější rozbor těchto jistě zajímavých problémů se však vymyká omezenému rozsahu tohoto příspěvku.

7. Závěr

Lokalizace softwarových produktů i dat do různých národních prostředí je významná z hlediska uživatelova komfortu, ale i z hlediska správnosti a přesnosti funkcí systému a z hlediska omezení chybovosti. S rozvojem mezinárodních kontaktů při výměně dat a programů i při přímém online propojování různých národních systémů nabývá otázka národních prostředí a jejich kompatibility zcela nových rozměrů.

Literatura

1. Chroust, Gerhard: National Language Translation, IDIMT 95 Proceedings, Vysoká škola ekonomická, Praha, 1995
2. Bébr,R.: Články o českém prostředí - Softwarové noviny, Příspěvky pro seminář Ostrava