

# JEZIK, KULTURNI ELEMENTI IN NABORI ZNAKOV ZA INTERNETOVE UPORABNIKE

Franci Močilar, Borka Jerman-Blažič

## Povzetek

Ljudje hočemo in potrebujemo medsebojno komunikacijo. Internet nam to omogoča, potrebno ga je le približati končnemu uporabniku. Idealno bi bilo, če bi uporabnik lahko komuniciral v jeziku, ki ga uporablja v svojem domačem okolju. V praksi pa bo že veliko doseženo, če bo lahko glavne ukaze računalniku podajal v svojem jeziku, če bodo na njegovem računalniškem zaslonu razumljivi znaki (izbran pravi nabor znakov) in bodo elementi kulturnega okolja (datum, ura, decimalna vejica, decimalna pika, merski sistem) predstavljeni v razumljivi obliki.

Na praktičnih izkušnjah je nastal model transparentne obdelave naravnega jezika. Obsega rešitve za različne jezike sveta, glede zmešnjave kodnih tabel in naborov znakov, pretvorb med tabelami, transliteracije in internacionalizacije. Zadnji nivo modela naj bi na podlagi jezikovne analize omogočil tudi strojno prevajanje besedil.

## Abstract

*People want communication and they need it. Internet makes it possible, still it is necessary to approach it to the end user. It would be ideal to communicate with the others in your own language. A lot will be done if the input of the main computer commands in native language will be enabled, if the characters on the screen will be comprehensible (the right character set will be chosen) and if the culture specific characteristics (date, time and numeric formatting, measurement system) will be presented in a comprehensible way.*

*On practical basis the transparent language processing (TLP) model has been developed. It solves the problems of different languages, the confusion of different character sets and code sets, table conversions, transliteration, internationalization. The last level of the model should enable machine translation, on basis of language analysis.*



## Uvod

Uporabniki Interneta danes delajo v sodobnih telekomunikacijskih omrežjih z multi-medijskimi storitvami. Slike in zvok multi-medijskih aplikacij prispevajo k prijaznosti storitve, a kljub temu prikazani tekst, ki je sestavni del multi-medijske aplikacije, uporabniku pomeni še zmeraj zaporo za lažje razumevanje in komuniciranje. Ljudje smo navajeni komunicirati v jeziku okolja, v katerem živimo in delamo. Idealna storitev v sodobnih omrežjih je storitev, ki bi uporabnikom omogočala komunikacijo v jeziku, ki ga uporabljajo v svojem domačem okolju. Povprečen uporabnik ne govori in ne piše angleško. Žal poteka večina aplikacij v mednarodnih omrežjih, kot je Internet (univerzalni direktorij po standardu X.500, knjižnični katalogi, elektronska pošta, učenje na daljavo, nakupovanje, poslovanje in pridobivanje informacij z uporabo orodij svetovnega spleta), v angleškem jeziku. Informacij in tekstovnih zapisov, pripravljenih v drugem jeziku (na primer v slovenščini), ni možno ustrezno prikazati na zaslonih doma in v tujini, ker zahtevani in potrebni znaki niso podprti v aplikacijah, ki tečejo v omrežju.

Pogoj za učinkovito in produktivno komunikacijo v več jezikih je uporaba mrežnih storitev v jeziku, naboru znakov in kulturnem okolju uporabnika. Model, ki nam

to zagotavlja, je model transparentne obdelave naravnega jezika ali TLP (Transparent Language Processing), ki ga predstavljamo v tem prispevku. Kljub eksplozivni rasti po vsem svetu sodobne storitve Interneta še nimajo izdelanega standardnega načina predstavitve, upravljanja in prikazovanja podatkov v različnih jezikih.

## Nabori znakov in kodne tabele

Informacija, ki potuje med ljudmi, je sestavljena iz besed, stavkov, odstavkov itd. Osnovni element je znak (character). Različne pisave sveta uporabljajo različne znake, ki so spravljeni v različne nabore znakov (character sets). Poleg znakov za prikaz podatkov (črke, številke, simboli) imamo v naborih znakov še posebne znake za organizacijo in kontrolo prenosa. Za prenos prek sistema informacijske tehnologije je znake potrebno kodirati, ker računalniki operirajo le z binarnimi podatki. Kodirani znaki so spravljeni v kodnih tabelah (code sets). Kodna tabela je torej preslikava nabora znakov v ustrezne številčne kode. Pravimo, da je tabela  $n$ -bitna, če vsak znak kodiramo z  $n$  biti. To pomeni, da lahko zakodiramo  $2^n$  znakov. S sedem bitno kodno

tabelo lahko predstavimo  $2^7=128$  znakov, z osem bitno  $2^8=256$  znakov, s šestnajst bitno  $2^{16}=65536$  znakov.

Da bi si lahko ljudje izmenjevali podatke, je kodne tabele potrebno standardizirati. Na različnih jezikovnih področjih so se uveljavili različni standardi. Edino standard ASCII (American Standard Code for Information Interchange) in njegova sedem bitna kodna tabela se uporabljata povsod, problem pa je v tem, da podpira le angleščino, jezik svahili in havajščino. Vsi drugi jeziki vsebujejo znake, ki jih v ASCII ni. Mednarodna inačica tega standarda je ISO 646 (International Standard Organization - mednarodna organizacija za standarde).

## Unicode

Unicode je 16-bitna kodna tabela, ki jo je izdelal konzorcij proizvajalcev Unicode. Je univerzalna, kar pomeni, da pokriva znake za vse pisave sveta: latinico, cirilico, grščino, kitajske znake, indijske in še mnoge druge. Do danes je izmed 65536 kod zasedenih okrog 39000, preostale so namenjene znakom, ki mogoče še niso zaobjeti, in nadaljnjim razširitvam. Kode Unicode 2.0 so identične s 16-bitnimi znaki ISO/IEC 10646, standardom ISO, ki je nastajal vzporedno z Unicode-om; [Unicode].

## Različni nabori znakov pri nas

Tudi če pogledamo samo slovenski jezik, trenutno uporabljamo vsaj pet različnih predstavitev znakov oziroma tako imenovanih kodnih tabel.

V uporabi so naslednje kodne tabele:

- **YUS I.B1.002/ISO 646-YU:** to je še jugoslovanski 7-bitni standard, ki poleg slovenščine pokriva še hrvaščino. Neangleški znaki so nameščeni na mesta manj uporabljenih ASCII znakov: č namesto ~, š namesto {, ž namesto ', é namesto }, d namesto |, Č namesto ^, Š namesto [, Ž namesto @, Ć namesto | in Đ namesto \.
- **ISO/IEC 8859-2 (Latin 2):** 8-bitna kodna tabela vsebuje znake, ki se uporabljajo v jezikih centralne Evrope: češki, hrvaški, madžarski, poljski, romunski, slovaški, slovenski in lužiško srbski jezik. Ta nabor znakov vsebuje tudi vse potrebne znake za albanski, angleški, finski, irski in nemški jezik, ki pa so tako in tako vsebovani že v naboru ISO 8859-1 (Latin 1).
- **IBM Code Page 852:** IBM-ova 8-bitna kodna tabela, ki pokriva jezike centralne Evrope. Uporablja se v MS-DOS in OS/2.
- **Microsoft Code Page 1250:** 8-bitna kodna tabela, ki se uporablja v Microsoft Windows EE (vzhodno evropska izdaja Windows).
- **ISO 10646/Unicode:** univerzalna kodna tabela za vse pisave sveta (latinica, cirilica, grščina, kitajski znaki, indijski, ...). Trenutno se uporablja še zelo redko. Predvideva se, da bo v prihodnosti Unicode prevladal.

**CSZcsz:** nekateri uporabljajo namesto šumnikov kar znake brez strešic, kar lahko pripelje tudi do zmešnjave. Seveda pa je ideja uporabna, če sistem ne podpira katerega izmed zgoraj naštetih naborov znakov (varianta je tudi \*c, \*s, \*z ali "c, "s, "z).

Pogljemo si praktičen primer. Na spletnem strežniku (Web Server) objavimo svoj dokument. Namenjen je uporabnikom, ki do njega dostopajo z različnih računalniških sistemov - uporabljajo različne kodne tabele znakov. Če hočemo, da bodo informacije iz dokumenta res uporabne in razumljive, potem moramo dokument pripraviti ter shraniti v več različnih kodnih tabelah. Vsak uporabnik izbere dokument v kodni tabeli, ki mu ustreza glede na njegov računalniški sistem. To pomeni, da podatki zasedejo mnogo več prostora, hkrati pa mora uporabnik misliti še na to, kateri dokument (v kateri kodni tabeli) naj izbere.

Seveda bi bilo ugodnejše, če bi podatke hranili v eni sami univerzalni kodni tabeli (npr. Unicode), ki bi se po potrebi pretvorili v uporabniku razumljive znake. S stališča uporabnika naj bi vse potekalo transparentno, torej bi se avtomatsko izbrala kodna tabela, ki jo uporabnik trenutno uporablja oziroma tabela, ki jo podpira njegov računalniški sistem. Uporabniku se s tem ne bi bilo potrebno obremenjevati.

## Internacionalizacija in lokalizatorji

Posamezni jeziki in države različno obdelujejo in shranjujejo podatke. To so na primer različni nabori znakov, pravila za sortiranje, datumski in časovni formati, zapis števil (decimalna vejica oz. decimalna pika). Internacionalizacija<sup>1</sup> pomeni razvoj programskih izdelkov [i18n Guide], ki delujejo v več kot samo enem naravnem jeziku oziroma okolju (jeziku je dodano še okolje, ker imamo lahko za isti jezik različna pravila: npr. angleščina v Veliki Britaniji, ZDA in Avstraliji). Izdelki so narejeni tako, da lahko pokrivajo različne kulturne in jezikovne potrebe uporabnikov [Windows i18n].

Specifikacije posameznega kulturnega okolja oziroma tako imenovani lokalni elementi so specificirani v registru "lokalnih kulturnih elementov" ali v tki. lokalizatorjih. V registru so specificirane tako kodne tabele, ki se uporabljajo v določenem okolju, kot ostale kulturne konvencije ter naravni jezik. Vsako jezikovno okolje ima svoja pravila shranjena v svojem lokalizatorju, ki ga uporabniška aplikacija oziroma program uporabi, ko je to potrebno; [Lokalizator]. Internacionalizacija omrežnih storitev naj bi omogočila:

<sup>1</sup> Internacionalizacija prihaja iz angleške besede internationalization. Iz praktičnih razlogov so jo skrajšali v **i18n** (med **i** in **n** je 18 črk). Enako pomeni tudi izraz globalizacija, ki je nasprotje izrazu lokalizacija (localization oziroma skrajšano **l10n**). **L10n** pomeni proces prilagoditve programskega produkta specifičnemu okolju. Internacionalizacija je širši pojem in vsebuje tudi lokalizacijo.

- povezavo uporabniškega programskega vmesnika (API - Application Programming Interface) odjemalca s funkcijo, ki opravi identifikacijo lokalizatorja uporabnika. Strežniku poda informacijo o uporabljeni kodni tabeli, jeziku in drugih lokalnih elementih, pomembnih za določeno kulturno okolje.
- zagotovitev komunikacije človek-stroj v zahtevanem lokalizatorju ter možnost preklapljanja med različnimi lokalizatorji.
- možnost izbire preferenčnega lokalizatorja v posamezni storitvi.
- zagotovitev povezave ter klic uporabniškega programskega vmesnika v strežniku, ki bo zagotovil ustrezno prevajanje in konverzijo zahtevanih kodnih tabel.

### Preslikave z zamenjavo znakov

Pri pretvorbi znakov oziroma črk ene abecede v drugo (npr. nemške črke v slovenske) v lokalnem naboru znakov ni mogoče prikazati in shraniti vseh znakov. V tem primeru je potrebna preslikava z zamenjavo, pri čemer problematičen znak zamenjamo s kakšnim drugim znakom oziroma z več znaki. Ločimo tri različne tipe preslikav:

- preslikava en-znak-v-enegega, z izgubo informacije

Nekaj primerov:

|   |   |   |
|---|---|---|
| A | v | A |
| Ä | v | A |
| C | v | C |
| Č | v | C |
| Đ | v | D |
| @ | v | a |
| © | v | C |
| & | v | & |

Pri nekaterih znakih izgubimo informacijo, saj npr. ne vemo ali nam predstavitevni znak C v resnici predstavlja C ali Č.

- preslikava en-znak-v-mnogo, z izgubo informacije

Nekaj primerov:

|   |   |    |
|---|---|----|
| A | v | A  |
| Ä | v | AE |
| C | v | C  |
| Č | v | C< |
| Đ | v | DJ |
| @ | v | at |
| © | v | Co |
| & | v | &  |

Ta tip preslikave je malo boljši, ker problematičen znak zamenjamo z več znaki, a še vedno izgubimo informacijo. Sedaj sicer vemo, da C predstavlja C, toda C< je še vedno lahko Č ali pa zaporedje dveh znakov - C in <.

- preslikava en-znak-v-mnogo, z ohranitvijo informacije na nivoju znaka

Nekaj primerov:

|   |   |     |
|---|---|-----|
| A | v | A   |
| Ä | v | &AE |
| C | v | C   |
| Č | v | &C< |
| Đ | v | &DJ |
| @ | v | &at |
| © | v | &Co |
| & | v | &&  |

Pri tretjem tipu vedno vemo, kateri znak predstavlja zaporedje znakov, ker se vsi problematični znaki začnejo z znakom & (sam znak & pa predstavimo z dvojnim &&).

Vse te preslikave se izvajajo nekje v ozadju računalniškega sistema in končnemu uporabniku zanje sploh ni potrebno vedeti. Zanj je pomembno le, da so znaki prikazani pravilno.

### Transliteracija

Transliteracija je proces pretvorbe znakov iz ene pisave v drugo (npr. cirilico v latinico, grške črke v latinske...). Je jezikovno odvisna, kar pomeni, da se npr. cirilica drugače prevede v slovenščino kot v angleščino, deluje pa na osnovi transliteracijskih tabel.

### Standardizacija

Z internacionalizacijo se je pojavila tudi potreba po standardizaciji na tem področju. Obstaja precej standardizacijskih teles, ki poskušajo uskladiti različne pristope k internacionalizaciji, kot tudi k prenosljivosti produktov med različnimi sistemi. Nekateri od teh standardizacijskih teles so:

- ANSI skrbi za ameriške standarde s področja računalništva. Nekateri od njih so vključeni v mednarodne ISO standarde.
- ISO (International Organization for Standardization): mednarodna organizacija za standardizacijo, ki izdaja mednarodne standarde, na katere poskušajo vplivati tudi vsa ostala standardizacijska telesa.
- POSIX (Portable Operating System Interface for Exchange) je družina standardov za prenosljivost med UNIX operacijskimi sistemi. Standard je skladen s standardom ISO-IEC 9945-2 2 CD 1991.
- RFC (Request For Comments) dokumenti so standard za Internet. To so dokumenti z definicijami protokolov in politike Interneta in tvorijo osnovo Internetove tehnične dokumentacije.

Pomembno je, da tudi Slovenija sodeluje v standardizacijskih procesih, da bodo mednarodni standardi s področja internacionalizacije uporabni tudi pri nas.

## Model transparentne obdelave naravnega jezika (TLP-Transparent Language Processing)

Za lažjo razvrstitev različnih potreb uporabnika ter storitev, ki jih potrebuje, je bil razvit štiri nivojski model transparentne obdelave naravnega jezika. To pomeni, da je obdelava za končnega uporabnika transparentna in se izvaja nekje v ozadju računalniškega sistema. Uporabniku zanj sploh ni potrebno vedeti in se z njo obremenjevati. Kot primer naj navedem uporabo elektronske pošte v Sloveniji, ko moramo pri pošiljanju vedno misliti na to, kako bo sprejemnik videl šumnike na svojem računalniškem sistemu (če to sploh vemo), pri sprejemanju pošte pa se mučimo z dešifriranjem čudnih znakov, ki naj bi predstavljali šumnike. Mnogi se temu problemu izognejo kar s pisanjem šumnikov brez strešic, kar pa zopet pripelje do dvoumnih situacij.

Rešitev je v izvedbi modela transparentne obdelave naravnega jezika. Model sestavljajo štiri nivoji, od nivoja 0, ki predstavlja minimum pri neangleški uporabi mrežnih storitev, do nivoja 3, ki posega že v področje umetne inteligence in avtomatskega prevajanja besedila.

### Nivo 0

Prvi nivo, nivo 0, je najnižja stopnja, potrebna za dostop do podatkov. Osnovni namen je pretvorba znakov v kodirani format, ki ga razume računalniški sistem uporabnika. Pretvorba znakov je transparentna in temelji na standardiziranih imenih naborov znakov, pretvorbenih tabelah in opisnih datotekah za pretvorbo, ki se razlikujejo glede na jezik.

Ta nivo je že izveden na različnih področjih, ni pa standardiziran. Izveden je bil delno in neskladno med prodajalci, dobavitelji in proizvajalci programske opreme.

### Nivo 1

Naslednji nivo omogoča uporabniku, da bere besedilo, ki je bilo prvotno v nekem drugem sistemu znakov ali pisavi (transliteracija) in da ima specifikacije posameznega kulturnega okolja (datum, ura, številski format...) ustrezno predstavljene, npr. datum 8.2.1848 je v ZDA prikazan kot 2/8/1848.

### Nivo 2

V mrežah lahko neko sporočilo nastane v kateremkoli

|                                                            |
|------------------------------------------------------------|
| <b>Nivo 3: strojno prevajanje</b>                          |
| <b>Nivo 2: osnovno prevajanje shranjenih izrazov</b>       |
| <b>Nivo 1: transliteracija in kulturni elementi okolja</b> |
| <b>Nivo 0: pretvorba znakov</b>                            |

Slika: Nivoji transparentne obdelave naravnega jezika

vozlšču, skozi katero potujejo podatki, od izvora podatkov do končnega uporabnika. Če pošiljatelj sporočila ne ve, kateri jezik uporablja končni uporabnik, se sporočilo lahko pojavi v uporabniku nerazumljivem jeziku. Zgodi se tudi, da pri iskanju podatkov z oddaljenega sistema le-teh ne najdemo, ker so shranjeni v kakšnem drugem jeziku.

Te probleme rešuje nivo 2. Cilj je zagotoviti mehanizem konsistentnega prevajanja izrazov za potrebne pare jezikov. Mrežna aplikacija glede na jezikovni par, identifikator vsebine in seveda glede na niz besedila poišče ustrezen prevod. Na tem nivoju se ne prevajajo celotna besedila, temveč samo nekatere besede in sporočila (npr. "Out of memory" se nadomesti z "Ni dovolj pomnilnika").

### Nivo 3

Ta nivo naj bi omogočil strojno prevajanje besedil. Za razliko od nivoja 2, kjer imamo pare besed oziroma sporočil, deluje na podlagi jezikovne analize in sodi v področje umetne inteligence. V svetu obstaja kar nekaj sistemov za strojno prevajanje (v glavnem iz in v angleščino), ki so iz dneva v dan kvalitetnejši, toda trenutno še vedno ne morejo nadomestiti človeka v vlogi prevajalca. Lahko pa mu pomagajo in olajšajo delo.

## Projekt MAITS (Multilingual Application Interface for Telematic Services<sup>2</sup>)

Poznamo več rešitev za prilagoditev in lokalizacijo programske opreme kulturnemu okolju. MAITS [MAITS] je projekt iz četrtega okvirnega programa Evropske skupnosti. Projekt želi združiti in nadgraditi različne tehnologije v celoto. V okviru MAITS-a se razvija uporabniški vmesnik (API), ki bo zagotavljal štiristopenjsko podporo jezikovnim potrebam uporabnikov. Stopnje so skladne z nivoji transparentne obdelave naravnega jezika:

- 0) pretvorba znakov med odjemalcem in strežnikom
- 1) transliteracija in kulturni elementi okolja
- 2) osnovno prevajanje shranjenih izrazov
- 3) dostop do strojnega prevajanja

Rezultati projekta bodo uporabljani v mrežnih storitvah, kot so X.400, X.500, Internetova elektronska pošta in v svetovnem spletu (WWW) za več jezikov. Izvajalci projekta bodo tudi aktivno sodelovali in prenesli rezultate v ustrezna standardizacijska telesa. V uporabniški skupini projekta so vključeni še ponudniki telematskih storitev in njihove stranke, razvijalci programske opreme in ponudniki informacij. Dolgoročno bodo lahko MAITS uporabljali vsi uporabniki X.400 in Internetove elektronske pošte, uporabniki imenika po

<sup>2</sup> Večjezikovni uporabniški vmesnik za telematske storitve.

standardu X.500 in uporabniki svetovnega spleta. MAITS API bodo lahko uporabljali tudi neodvisni proizvajalci programske opreme za internacionalizirane izdelke.

Tipičen uporabnik MAITS-a s svojim osebnim računalnikom (PC ali Mac) zajema podatke, ki so shranjeni na strežniku v drugačni kodni tabeli znakov. V mnogih primerih bi bili brez ustrezne pretvorbe akcentirani znaki nepravilno predstavljeni. Na najnižji stopnji v aplikaciji MAITS bo pretvorba za uporabnika transparentna. Na višjih nivojih bo npr. grški tekst predstavljen z latinico, ključne besede pri iskanju v imeniku X.500 bodo prevedene in uporabljeno bo strojno prevajanje za prevod tekstov v svetovnem spletu (WWW).

Pričakovani rezultat projekta MAITS je boljša komunikacija med uporabniki, ki govorijo različne jezike, kot tudi boljša dostopnost podatkov na različnih krajih, zapisanih z različnimi nabori znakov v različnih kodnih tabelah, v različnih pisavah in v različnih jezikih.

Narejeni in trženi bodo X.400, X.500, pregledovalnik za svetovni splet (WWW) in programsko razvojno orodje. Firme Sybase<sup>3</sup> in NEXOR<sup>4</sup> bosta vključili MAITS med svoje produkte. Pričakujemo tudi, da bodo standardi, razviti v projektu, splošno sprejeti v standardizacijskih in telematskih skupnostih.

## Pričakovanja

Pričakuje se, da bo v prihodnosti zmeda glede različnih kodnih tabel urejena z univerzalno kodno tabelo ISO 10646 (Unicode). Problem bo rešen univerzalno, za vse jezike in pisave. V vmesnem času, v prehodnem obdobju, ko produkti še ne podpirajo Unicode-a, bodo še vedno potrebna orodja za prevajanje in pretvorbo med različnimi kodnimi tabelami.

Večina programskih hiš danes ne razmišlja več lokalno temveč globalno. Z internacionalizacijo se ukvarjajo velike kot tudi male programske hiše. V konzorciju Unicode sodelujejo IBM, Microsoft, Novell,... Pri

IBM-u je izšla publikacija [IBM] o njihovi arhitekturi znakovnih naborov, ki vključuje strategije in ideje za pretvorbo in prikaz znakov iz različnih naborov znakov. Microsoft je operacijski sistem Windows NT zgradil na osnovi Unicode; [Win32, NLSAPI]. Novellov NetWare 4.01 Directory Services shranjuje vse objekte in druge informacije v Unicode-u. Unicode je bil izbran, ker so tako zmanjšali probleme, ki nastanejo z globalnimi mrežami in z mrežnimi odjemalci, ki uporabljajo neskladne kodne tabele. Ker Unicode vključuje vse glavne nacionalne in industrijske standarde, ne prihaja do izgube informacij pri pretvarjanju iz Unicode-a v kodno tabelo odjemalca.

Naj uporabim še slogan konzorcija Unicode: Ko se svet hoče pogovarjati, govori Unicode<sup>5</sup>.

Celovita rešitev za model transparentne obdelave naravnega jezika še ne obstaja. Na trgu obstaja precej produktov, nobeden pa ne ustreza vsem zahtevam modela. Projekt MAITS bo pripomogel, da bo model prešel v praktično uporabo in razbremenil uporabnika.

## Literatura

- [i18n Guide] Sandra Martin O'Donnell: *Programming for the world, A Guide to Internationalization*, Prentice Hall 1994
- [IBM] IBM Character Data Representation Architecture, Reference and Registry, SC09-2190-00, December 1995
- [Lokalizator] Borka Jerman - Blažič: *Lokalizator informacijske tehnologije za Slovenijo, zaključno poročilo projekta IJS in Centra vlade za informatiko (T2-6492)*
- [MAITS] <<http://www.dkuug.dk/maits/>>
- [Win32] *Unicode Support in Win32, 1992-1995 Microsoft Corporation, May 1994*
- [NLSAPI] *NLSAPI Functional Specification 5.4 - Microsoft Developer Network Library, Draft 5.4, January 5, 1994*
- [Unicode] <<http://www.unicode.org/>>
- [Windows i18n] Nadine Kano (Microsoft): *Developing international software for Windows 95 and Windows NT*

<sup>3</sup> Sybase je ameriška programska hiša, ki razvija programske produkte tipa odjemalec/strežnik.

<sup>4</sup> Nexor je angleška programska hiša, specializirana predvsem za sisteme elektronskih sporočil in univerzalnih direktorijev.

<sup>5</sup> When the world wants to talk, it speaks Unicode.

Franci Močilar je diplomiral leta 1993 na Fakulteti za elektrotehniko in računalništvo in Ljubljani. Trenutno je zaposlen kot mladi raziskovalec na Institutu Jožef Stefan, v Laboratoriju za odprte sisteme in mreže. Njegovo delovno področje je internacionalizacija omrežnih storitev in sodeluje pri projektu MAITS, iz četrtega okvirnega programa Evropske skupnosti.

Prof. dr. Borka Jerman-Blažič je vodja laboratorija za odprte sisteme in mreže na Institutu Jožef Stefan. Je članica strokovnega sveta ARNES, tehničnega komiteja TERENA (Trans European Research and Academic Networks Association). Predseduje slovenskemu standardizacijskemu komiteju za informacijsko tehnologijo (JTC1) in je predstavnik Slovenije v ISO JTC1 SC2, JTC2/WG20 in CEN TC 304. Je tudi predsednica Slovenskega dela mednarodnega združenja Internet Society (ISOC-SI). Trenutno se njeno področje raziskovanja osredotoča na mrežne aplikacije, na varnostne politike v omrežju in na internacionalizacijo omrežnih storitev.