

DEVELOPMENT OF **xindy** SORT AND MERGE RULES FOR INDIC LANGUAGES

Zdeněk Wagner, Praha, Česká republika

Anshuman Pandey, Univ. Michigan, USA

Jaya Saraswati, Mumbai, India

Note on pronunciation of xindy

Note on pronunciation of xindy

Czech: *ks*

Note on pronunciation of *xindy*

Czech: *ks*

English: usually as *z*

Note on pronunciation of xindy

Czech: ks

English: usually as z

Hindi: as क्श, लक्ष्मी can be transliterated either Lakshmi or Laxmi

Note on pronunciation of xindy

Czech: *ks*

English: usually as *z*

Hindi: as *kṣ*, लक्ष्मी can be transliterated either Lakshmi or Laxmi

Chinese: as *sh*

Note on pronunciation of *xindy*

Czech: *ks*

English: usually as *z*

Hindi: as *ks*, लक्ष्मी can be transliterated either Lakshmi or Laxmi

Chinese: as *sh*

Russian: *хинди* (meaning Hindi)

Note on pronunciation of x̣indy

Czech: *ks*

English: usually as *z*

Hindi: as *kṣ*, लक्ष्मी can be transliterated either Lakshmi or Laxmi

Chinese: as *sh*

Russian: *хинди* (meaning Hindi)

x̣indy sorts Hindi

MakeIndex

- version for English and German
- *CIndex* – version for Czech and Slovak
- unpublished version for Sanskrit (Mark Csernel)

Tables defining the sort algorithm are hard-wired in the program source code. Modification for other languages is difficult and leads rather to confusion than to development of a universal tool.

International MakeIndex

- Tables defining the sort algorithm present in external files.
- Sort rules defined by regular expressions.



- Program written in Common LISP.
- Definition tables in external files.
- Tables in the LISP syntax are generated from a human readable files by perl script `make-rules.pl`
- **xindy** is usable not only in \TeX and deals with more complex problems than *MakeIndex*.

Text a indic languages

- Indic languages make use of non-latin scripts:
 - derived from brāhmī
 - modified and extended arabic script (Urdu, Kashmiri)

TeX a indic languages

- Indic languages make use of non-latin scripts:
 - derived from brāhmī
 - modified and extended arabic script (Urdu, Kashmiri)

Lowercase and uppercase not used.

TeX a indic languages

- Indic languages make use of non-latin scripts:
 - derived from brāhmī
 - modified and extended arabic script (Urdu, Kashmiri)

Lowercase and uppercase not used.

- Two approaches:
 1. Latin transliteration
 2. Unicode

Examples of transliterations

- Velthuis Devanāgarī for T_EX: Sanskrit, Hindi, Marathi, Nepali, Bhojpuri, Maithili, . . .

Examples of transliterations

- Velthuis Devanāgarī for \TeX : Sanskrit, Hindi, Marathi, Nepali, Bhojpuri, Maithili, . . .
- Similar transliteration for bengālī and for gurmukhī (Panjabi language)

Examples of transliterations

- **Velthuis Devanāgarī for T_EX**: Sanskrit, Hindi, Marathi, Nepali, Bhojpuri, Maithili, . . .
- Similar transliteration for **bengālī** and for **gurmukhī** (Panjabi language)
- **ITRANS** for devanāgarī, bengālī, gurmukhī (no longer supported)

Examples of transliterations

- **Velthuis Devanāgarī for T_EX**: Sanskrit, Hindi, Marathi, Nepali, Bhojpuri, Maithili, . . .
- Similar transliteration for **bengālī** and for **gurmukhī** (Punjabi language)
- **ITRANS** for devanāgarī, bengālī, gurmukhī (no longer supported)
- System not based on T_EX: **IITK** (used for Hindi and Marathi by *Resource Center for Indian Language Technology Solutions, Indian Institute of Technology Bombay*)

Usage of transliteration systems

- Input for an indexing software in a transliteration.
- Output from an indexing software will be processed by a program that expects the text in the very same transliteration.

The indexing software should work directly in the transliteration otherwise we would need two more code conversions.

Work in Unicode

Ω - \TeX a $X\TeX$ work internally in Unicode, input and output is usually in Unicode (UTF-8, UTF-16).

Indexing software should work in Unicode.

Alphabetical sorting in Czech

1. Accented characters: *č, ř, š, ž*; conjunct *ch*
2. Other accented characters distinguished only if the words are otherwise equal:

plast < plást < plat < plát < platno < plátno < platnost

3. Lowercase/uppercase, formerly:

daněk < Daněk < rybička < Rybička < sojka < Sojka

now:

Daněk < daněk < Rybička < rybička < Sojka < sojka

Two uppercase variants of *ch*: *Ch, CH*.

xindy features

1. Definition of alphabet elements including conjuncts (ligatures).
2. Definition of sort order.
3. Definition of equivalences (accented characters).
4. Ligatures (e. g. β in German).
5. Ordering uppercase and lowercase (aA, Aa).
6. Usage of the same rules in different encodings (UTF-8, ISO 8859-2, CP 1250, CP 852).

Alphabetical sort order in Indic languages

1. Vowels : अ आ इ ई उ ऊ ऋ ॠ ऌ ॡ ए ऐ ओ औ

2. Consonants according to position of pronunciation

- *k* (guttural, कंठ्य , कवर्ग)
- *c* (palatal, तालव्य , चवर्ग)
- *ṭ* (retroflex, मूर्धन्य , टवर्ग)
- *t* (dental, दंत्य , तवर्ग)
- *p* (labial, ओष्ठ्य , पवर्ग)
- *y* (liquids, अंतस्थ)
- *ś* (sibilants, ऊष्म)
- *h* (aspirate, संघर्षी)
- special conjuncts *kṣ* and *jñ* in some languages

Sort order of vowels

Short and long vowels are distinguished.

अगर < आ

(agar < ā)

कम < का

(kam < kā)

Sort order of anusvara and candrabindu

आँख < आकाश

पलंग < पलँग < पल

संभव < सँभाल < संभावना < सच < सभी

काम < किंतु < कि < किताब

नंगा < न < नगर

इंद्रधनुष < इधर < उधर < ऊँट < ऊपर

Consonants with nuktas

Nukta is treated similarly as an acute accent in Czech:

कुतुब < कुतुब < कुतुबनुमा < कुत्ता
(kutub < qutub < qutub'numā < kuttā)

Translation: books (Arabic internal plural); pole, the pole star; compass; dog

खाना < खाना
(khānā < khānā)

Translation: meal; house

Properties of Unicode

- Independent and dependent vowels distinguished:
आसान = āsān
- Inherent *a* after a consonant can be changed by a *matra* sign or suppressed by a *virama*: प = pa, पु = pu, पे = pe, प् = p
- Conjuncts formed by inserting a *virama* between consonants: शक्ति = शक्ति = श + क + ् + त + ि

Solution in Unicode

1. Candrabindu treated as if it were an uppercase variant of anusvara
2. Characters with anusvara and candrabindu precede the corresponding character but belong to the same group (आँख will be in group आ)
3. Characters with nuktas defines as ligatures in order to simplify the rules
4. Lowercase characters precede uppercase

Part of the Hindi definition table

```
$alphabet = [
...
['आ', ['आं', 'आँ']],
['आ', ['आ']],
...
['इ', ['इँ']],
['इ', ['इँ']],
...
['क', ['कं', 'कँ']],
['क', ['क']],
...
['ऋ', ['ऋं', 'ऋँ']],
['ऋ', ['ऋ']],
...
['ॠ', ['ॠं', 'ॠँ']],
['ॠ', ['ॠ']],
...
['ऌ', ['ऌं', 'ऌँ']],
['ऌ', ['ऌ']],
...
['ॡ', ['ॡं', 'ॡँ']],
['ॡ', ['ॡ']],
...
];

$ligatures = [
[['क'], 'after', [['क']],
[['ख'], 'after', [['ख']],
...
];

$sortcase = 'aA';
```

This table will correctly sort:

कई < कल < किताब < की < कुआँ < कुत्ता

*Translation: several; yesterday/tomorrow; a book; . . . ; a well;
a dog*

Problems with nuktas

Although characters with nuktas such as क़, ज़, ड़, ढ़, फ़ are defined in Unicode, some people write them as the base consonant followed by combining diacritical mark U+093C.

Handled optionally by `-M dvngnukta` (not loaded automatically)

Sample PDF file available!

Properties of transliterations

Examples given in Velthuis transliteration

- Independent and dependent vowels not distinguished: आसान = AsAn
- Inherent *a* must always be written: कम = kam
- Conjuncts formed by writing consonants adjacent to each other: स्त्री = strI, पक्षी = pak.sI
- Duality possible in some transliterations: भूमि = BUmi = bhuumi = Buumi = bhUmi

Transliterations differ by assignment of Latin characters:

(पाठ्यपुस्तक = pA.thyapustak [Velthuis], pATyapuswaka [IITK]).

Solution for transliterations

1. Perl scripts for conversion from UTF-8 \Rightarrow transliteration (Velthuis, IITK)
2. Definition file written in UTF-8 (using independent vowels)
3. Definition file converted to a transliteration
4. Converted file translated by `make-rules.pl` to the LISP syntax for `xindy`
5. Duality handled in *merge rules*: *aa* sorted as *A*, *bh* sorted as *B*. The same mechanism is used to handle \TeX sequences generated by the `inputenc` package.

Merge rules for Velthuis transliteration

```
(merge-rule "aa" "A" :string)
(merge-rule "ii" "I" :string)
(merge-rule "uu" "U" :string)
(merge-rule "kh" "K" :string)
(merge-rule "gh" "G" :string)
(merge-rule "ch" "C" :string)
(merge-rule "jh" "J" :string)
(merge-rule "th" "T" :string)
(merge-rule "dh" "D" :string)
(merge-rule "ph" "P" :string)
(merge-rule "bh" "B" :string)
(merge-rule ".m" "M" :string)
(merge-rule "~~m" "/" :string)
(merge-rule "&" "_" :string)
```

Problem of अँ

Its alphabetical order is not defined in dictionaries.

हिंदी - चेक शब्दकोश lists it as the ओम् word.

It therefore seems that अँ is a symbol, not a character. It is not defined as an alphabet element.

Problem of आँ

आँ is not an original character in Indic scripts and textbooks and dictionaries do not define its alphabetic position. It was introduced to transliterate words of English origin. However, words as आँफ़िस , आँयल etc. are widely used nowadays.

Currently it is treated as if it were an uppercase variant of आ.

Problem of viramas

Ambiguities in dictionaries:

महान् < महानगर < महानुभाव

वाक्रिया < वाक् < वाक्छल

Devanāgarī numeric location class

Difficult to handle, needs to be programmed in LISP.

Handled by TECKit map in X₃TEX, arabic numbering used in the document but the map displays it in Devanāgarī

Markup rules for Velthuis transliteration

```
(markup-index :open
"\begin{theindex}
  \providecommand*\lettergroupDefault[1]{}
  \providecommand*\lettergroup[1]{%
    \par\textbf{#1}\par
    \nopagebreak
  }
@modernhindi
  {\dn\dnum
"
      :close "~n~n}\end{theindex}~n"
      :tree)
```

Current status

- Development system built as a reusable tool allowing
 - addition of languages
 - addition of transliterations
- Hindi and Marathi finished, just final check needed
- Works in T_EX Live 2008 and 2009 but not 2010, should not be difficult to make it work
- Almost ready for inclusion into xindy distribution
- URL of the β -version:
<http://icebearsoft.euweb.cz/xindy-devanagari/>

ॐ तत्सत्