TUSTEP and Culturally Correct Searching in Multilingual Corpora on the Web

Marc Wilhelm Küster

What it is all about

My talk will focus on two issues:

- Specifically European requirements in the field of Browsing and Matching;
- Concrete implementation strategies in TUSTEP.

A short live demonstration of actual applications − for obvious reasons not reproducible in this paper − will conclude the presentation.

Much of the justification of this talk hinges on the question why specifically European requirements of Browsing and Matching, the two technologies at the heart of information retrieval, are an issue in the first place.

HTML and with it the WWW − and this talk will focus on information retrieval over the Internet, though much of it is valid also for information retrieval through local databases − is a European invention, but it is hardly an overstatement that in its current incarnation it is, for better or worse, US-dominated. Especially the mechanisms for information retrieval are unsatisfactory from a European viewpoint.

Browsing and Matching

Mission and institutional background

The European Commission, in its aim to safeguard European needs in the Global Information Society, decided to investigate into the problem of information retrieval in Europe and founded a project with the aim of defining the precise scope for future action. The project, the manager and editor of which is identical to the author of this paper, was charged with investigating

> the European needs and problems with searching and browsing, in relation to character sets, transliteration, matching and ordering rules and other cultural specific elements.

In other words, it must explore how information retrieval can be facilitated for Europe and its citizens.

The project is part of CEN TC304's initiative to protect Europe's cultural diversity in the field of Information Technology (IT). TC304, a technical committee of CEN, has the mission to act on behalf of “European Localization Requirements” − this, at the same time, is its official name.

CEN, the *Comité Européen de Normalisation*, is the European standards body with a membership that encompasses Austria, Belgium, the Czech Republic, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland, and the United Kingdom. Affiliate members include Albania, Bulgaria, Croatia, Cyprus, Estonia, Hungary, Latvia, Lithuania, Malta, Poland, Romania, Slovakia, Slovenia, and Turkey.

Specific issues (selection)

Certain issues have already been identified in the scope of the project. It should look into topics such as:

− Character sets;
− Transliteration and fallback;
− Matching rules;
− Ordering rules.

Character sets

Certainly, the UCS is the encoding scheme of choice also for Europe − including full support for the W3C's "Requirements for String Identity Matching and String Indexing", once certain European concerns have been addressed. Web users should be consistently encouraged to publish new data on the Web in a UCS conformant storage format (most likely UTF–8).

Nevertheless, legacy character sets − both 7 and 8 Bit − are going to play an important, if not a dominating role for the foreseeable future. While it is a high priority that data be converted to the UCS, we are not likely to see even an approximate completion of that task for the next years, maybe not even for the next decades. Therefore, the intelligent handling of legacy character sets by information retrieval tools is of great importance. Ideally, no encoding differences would be exposed to the end user at all, though, for obvious reasons, this is a target that is impossible to attain in perfection.

Right now, the user is often fully confronted with encoding differences. Many search engines do not even properly support Latin 1 in its entirety or are even unable to equate entities such as `&eacute;` with their corresponding Latin 1 equivalent `é`.

Ordering rules

Ordering rules − the ABC − differ significantly between European languages and cultures. Portal sites, i. e. sites which structure information on the contents of other sites, with a Europe-wide target audience should employ standardized default orderings. CEN TC304 has developed such a standard, the *European Ordering Rules*, ENV 13710.

Transliteration and fallback

Transliteration is the one-to-one mapping of one script into another one. Transcription is the rendering of the pronunciation of one language in another script. Fallback is the representation of words in a character set that lacks certain letters which are really needed. In most real-life situations the wanting character set is, of course, ASCII.

These dry definitions can easily be filled with life. The great king דָּוִד is usually transliterated as *David*, the great Athenian leader Περικλῆς is usually transcribed as *Pericles* in English, and Herr Schröder usually renders his name as *Schroeder* when abroad.

Unfortunately, there are a great many different such transliteration and transcription schemes around. *Pericles* calls himself *Perikles* in Germany, the Russian ex-president Ельцин goes by the name of *Yeltsin* or (less frequently) *Yelzin* in English, *Yeltsin* in French, and *Jelzin* in German.

Transliteration, transcription, and fallback are everyday phenomena in Europe, yet most search engines offer little support even for very basic fallback rules, let alone transliteration or transcription. This makes pan-European information retrieval often very difficult.

Phonetically aware matching

To some degree similar is the problem of phonetically aware matching. Certain names − often very common ones − have a number of orthographic variants which represent the same sound pattern. Similarly, there are a number of spelling differences between varieties of the same word, e. g. *colour* vs. *color*.

For English, several working solutions are in existence for this problem. For many other European languages the situation is poor indeed.

Unorthodox orthographies

Standardized spelling as we know it is a pretty young phenomenon. Most historical documents show significant divergencies both from today's orthography, from that of their contemporaries and even within the document itself. Intelligent information retrieval that protects our Cultural Heritage must take account of this.

A straightforward example is the most famous soliloquy in the English language in its original folio spelling:

> To be, or not to be, that is the Question:
> Whether 'tis Nobler in the minde to suffer
> The Slings and Arrowes of outragious Fortune,
> Or to take Armes against a Sea of troubles,
> And by opposing end them: to dye, to sleepe
> No more; and by a sleepe, to say we end
> The Heartake, and the thousand Naturall shockes
> That Flesh is heyre too? 'Tis a consummation
> Deuoutly to be wish'd. To dye to sleepe,
> To sleepe, perchance to Dreame;

Current state of the Browsing and Matching project

The project has by now published its first draft which is available for public review. Comments are invited from all interested parties, both in and outside of Europe.

Contact details are:

- Draft: `http://www.stri.is/TC304/Matching`
- Mail: `kuester@zdv.uni-tuebingen.de`

TUSTEP

What is TUSTEP?

Before explaining how TUSTEP allows to design Culturally Correct searching on XML-databases on the web, I'll give a brief overview over TUSTEP, its main design principles and a small selection of its features.

TUSTEP, the TUebinger System of TExt Processing programs, is being developed by Prof. Ott and his team. He started work in the late 60s in Tübingen / Germany to automate the metrical analysis of Latin hexameters. The groundwork came to be used as the base for future accomplishments. TUSTEP now has a history of over thirty years of continuous enhancement. It is being used in many projects in both commercial and scholarly environments.

The original design principles are still valid: TUSTEP is built around

- Professionality;
- Integration;
- Portability;
- Modularity.

Professionality

From the beginning TUSTEP had to face the complex requirements of scholars working with the many different scripts which play an important part in our European Heritage. It was − and is − targeted to meet the needs of *professional* text data processing of the most varied kinds. It had to offer high performance; a top priority in the days when the price of processing time was at a prime. It still profits from algorithms that have been optimized to a degree rare in the industry.

Furthermore, it had to be able to cope with large data volumes. A single document can − and always could − contain up to 2GB of data. Such data which often contains many years and even decades worth of hard work can only be handled in a system where stability is central. The internal file format are designed in such a way as to make unwanted serious loss of data unlikely indeed.

Integration

TUSTEP offers tools for all stages of a project, from data entry to output on paper and, via CGI, on the web without ever leaving the system. However, since many projects use TUSTEP in a larger, more diversified environment, open interfaces such as XML allow for its well-defined integration. It is than at the user's discretion to decide which modules of TUSTEP to use.

Portability

There has never been a time when TUSTEP was used only on one type of system. While it was named TUSTEP in the early 70s when it was ported to the TR440, it was also used on other mainframe architectures. Nowadays, it runs under Windows 95 / 98 / NT / 2000 and various flavours of UNIX, including Linux. It is currently installed on systems ranging from high-performance compute servers to notebooks.

A corollary of this is the need for one data model for all platforms. The file model is strictly upwards compatible; files from the early 70s are still readable, and in fact there are large scale projects that depend on this every day.

Modularity

TUSTEP is a toolbox in which some fifty modules, most in themselves highly configurable, can be combined to handle even the most demanding tasks in the arena of text data processing. TUSTEP's powerful macro language coordinates the different tasks. All of this makes for TUSTEP's extreme flexibility.

Modules include:

- Typesetting;
- Index building;
- Intelligent file comparison;
- FCD.3 14651 conformant collation;
- File transformation and analysis.

Professional multilingual typesetting

TUSTEP's typesetting module is being used in many publishing houses for high-quality production of such diverse products as scholarly editions, monographs, bibliographies, dictionaries, encyclopedias, and large-scale database publishing. Books which were typeset with TUSTEP have won several of the prestigious "Schönstes deutsches Buch des Jahres" (most beautiful German book of the year) awards.

TUSTEP has been used early on for multilingual typesetting. It currently supports, amongst others, Arabic, Coptic, Cyrillic, Devanagari, Greek (classical and modern), Hebrew (fully vocalized), Latin, Phonetics (IPA), ...

TUSTEP and the UCS

Multilingual text

For historical reasons, multilingual text is stored internally in transliteration. This means, e. g., that דוד (the unvocalized equivalent of David) is stored as `#h+dud#h-` and the great Russian poetess Анна Ахматова as `#r+Anna Axmatova#r-`.

A similar approach is used for handling combining diacritics. To demonstrate the idea with a somewhat complex example, here the (to my knowledge non-existent) letter `LATIN SMALL LETTER A WITH OGONEK AND MACRON AND ACUTE` ą̄́ which is encoded as `%/%–%;;a`.

This approach was originally designed to allow for the handling of multilingual data on punched cards and simple terminals, but has many advantages even today. With only little training, it is possible to enter even long non-Latin texts conveniently on an ASCII keyboard. In addition, the user input can now immediately be visualized under Windows and can be displayed with any Unicode-conformant true-type-font.

A transformation utility allows for transparent mapping of TUSTEP's internal format to and from the UCS in both its UTF–8 and UTF–16 incarnations.

Output of multilingual texts

Multilingual text can be output on paper with the typesetting module. As usual in high quality printing, it allows the user a wide choice of postscript fonts, doing the necessary font mapping as needed.

For output from XML-conformant databases onto the net, TUSTEP uses UTF–8 which can be generated on the fly from its internal format. It can in a similar manner also react on queries using UTF–8 as their character set. The accompanying HTML / XML is entirely at the discretion of the implementer. TUSTEP does not force any standard query masks or output styles on anybody.

TUSTEP and XML

Data model

TUSTEP's modules are designed for dealing with data that contains (almost) arbitrary markup. This enables easy handling of structured data. They offer specific support for XML.

Additionally, TUSTEP's files have a special internal structure: Each unit is uniquely identified by a three-part record number, originally envisioned to represent the physical structure of paper documents such as books. This allows the same data model to be a viable choice for both texts and databases − a feature that greatly facilitates the most diverse uses of one and the same file, e. g. for intelligent access via the XML database server and for high-quality publication on paper.

Databases

Databases are in TUSTEP simply files with one or more repetitive structures. These structures can be (almost) arbitrary, though specific support is given for XML-conformant databases. A data field can again have freely selected substructures.

Let us assume a simple text, a bibliography which contains entries of the types `<book>` and `<article>`. Some of the entries are supposed to be obligatory (e. g. the title), others are optional (e. g. the author and the editor).

```
<book>
<editor>Strzych, Marianne; Weiß, Joachim</editor>
<title>Der Brockhaus in fünfzehn Bänden</title>
<year>1997</year>
</book>
<book>
<author>Ожегов, С. И.; Шведова, Н. Ю.</author>
<title>Толковый словарь русского языка</title>
<year>1995</year>
⟨/book⟩
<article>
<author>Einstein, Albert</author>
<title>Zur Elektrodynamik bewegter Körper</title>
<journal>Annalen der Physik</journal>
<year>1905</year>
</article>
```

In TUSTEP's macro language, these structures are mapped on the following `STRUCTURE` statements:

```
STRUCTURE biblio_book
"<book>" dummy
– "<author>" author "</author>"
– "<editor>" editor "</editor>"
+ "<title>" title "</title>"
+ "<year>" year "</year>"
+ "</book>" dummy
ENDSTRUCTURE
STRUCTURE biblio_art
```

```
"<article>" dummy
+ "<author>" author "</author>"
+ "<title>" articletitle "</title>"
+ "<journal>" journal "</journal>"
+ "<year>" year "</year>"
+ "</article>" dummy
ENDSTRUCTURE
```

When accessing the database, the values of the data fields are returned in the variables whose names are given in between the XML tags.

All of these fields are freely adaptable to suit (basically) any type of data which exists in the database. For practical purposes, there is neither a limit on the number of different data fields per structure nor on the length of their contents.

Speed of data access

TUSTEP allows for the construction of search indices that are entirely under the web site builder’s control. At the same time, it allows for efficient access via the record number and thus combines the advantages of flexible index construction and fast, hardware-oriented access.

The TUSTEP file structure allows to handle medium-sized databases (up to 2GB) in a single file. Even using an off-the-shelf PC running, e. g., Linux, as the server, response times are usually within fractions of a second.

TUSTEP and Culturally Correct Searching

TUSTEP offers a simple method to performing flexible and culturally adaptable searches through XML-databases. Any data field can be “normalized” individually for comparison. This allows for a different treatment of, e. g., names and titles or (to assume a customer database) locations and ordered items. (Of course, all fields can also be “normalized” identically, if desired).

This is achieved via the so-called `X_TABLE` mechanism which sports a powerful regular expression syntax which is both as old as the UNIX regular expressions and offers more possibilities.

> Note: TUSTEP also supports “brute force” fuzzy searching which offers a way of locating information that cannot be found via this more systematic approach.

A few (rather simple) examples for an `X_TABLE` that implements “normalized” search − a search type which we characterize as *intelligent fuzzy searching*:

- `˜ä˜a˜ae˜a˜` A with diaeresis and Ae and A are treated equivalently (essential for German names);
- `˜ye˜ie˜ey˜e˜<[co]e˜>=01˜ou<[na]˜o<=01˜<[vo]u<[vo]˜>=01v<=01˜ all˜al˜’d˜ed˜` Treat the Hamlet soliloquy correctly for searching purposes. `<[co]`, `<[na]`, `<[vo]` stand here for character groups which contain consonants, nasals, and vowels, respectively. Such character groups can be arbitrarily defined by the implementer;
- `˜þ˜th˜ð˜d˜` Deals with Icelandic;
- `˜%>@˜˜%>@>@˜˜` All diacritics are ignored for comparison;
- ...

Résumé

The European project team “Browsing and Matching” has identified a number of specifically European requirements in the field of information retrieval over the net. These cover topics such as

- Character sets;
- Transliteration and fallback;
- Unorthodox orthographies;
- Ordering rules.

TUSTEP, the Tuebingen-based toolbox of flexible text processing modules which include XML database support, can answer to many of these needs by offering the implementer a flexible macro language geared towards Culturally Correct Searching.

More information under `http://www.uni-tuebingen.de/zdv/tustep`