LaTeX 3 discussions at Pont-à-Mousson

2005-03-10

Present: B Bayart, J Bezos, T Lotze, J L Braams, F Mittelbach, M Høgholm, R Fairbairns, C A Rowley

1 Gdansk meeting

JB described issues on multilingual issues raised at the meeting.

Many items have been implemented, but not 'ij' ligatures (impossible to achieve within latex, and font designers seem not to be interested).

Changing language→changing hyphenation, now. Other things one may imagine being tied to a language per document: for instance French spacing conventions.

Word order changes appear in Basque (e.g., caption labels being "1 figure:"[1]).

Hyphenation; Portuguese hyphenation conventionally repeats the hyphen character at the start of the new line; French tradition specifies that a compound word hyphen be repeated at start of new line if it's taken as a break.

These things, and others like them, need to be categorised as "necessary for the language" and "typographic fashion".

Jobs like spacing around punctuation (in French) can't sensibly be done by changing markup. But this spacing isn't intrinsic to French (there exists a tradition of doing it in English, too, though now little practised). So perhaps we might add such things as a module, which is typically applied in French, but may be applied in other languages too.

We need also to consider the applicability of changes. Distinctions should give us at least "whole book" and "fragment of text", but intermediate "chapter" and "paragraph" can also be necessary in some cases.

Discussion of the error message language has (to some extent) been addressed by Bernard Gaulle's recent work. No enthusiasm for defining a macro name language...

line breaking re. very long words. tweak hyphen demerits actually helps, so this should be applied ("paragraph parameters") when using in latex. program extensions would help here, but who knows when they will be available. problem with changing for fragments because they *are* paragraph parameters.

[1] check this in the .ldf files

document class-defined typography will vary per language. does it make sense for such changes to be tied to language interface? things like heading design seem to show similarities across languages; but should one try to configure classes according languages? problem is that tagging structure and layout both spec by document class; should one have a tagging structure separately defined? then we can reasonably have separate implementations of (say) "article" per language. there are probably surprises to be found (do asian languages fit?). for example emphasis works oddly in old german (two different forms of emphasis, for german fraqments [sperrsatz] and foreign language fragments [more 'regular' roman font]). such changes can be implemented in existing classes by having \emph recognise the language in use.

note most packages extend the tagging model; that way we avoid an explosion of document classes.

language blocks/fragments recognised in existing latex models. babel currently recognises three classes of language change: change "everything", change "extras"+hyphenation, change hyphenation only.

so changing "everything" probably applies at the class level, since we imagine that chapter names (for example) will apply for the whole of a document. changing such things is actually difficult in current babel, since the language selection commands are not configurable, and appear at a pretty low level.

need to consider tagging hierarchy; want to avoid entwined hierarchies of language tagging and text tagging (e.g., "start language fragment in middle of para n, end it in middle of para $n + 1$").

quote structures interesting: do we have "français «deutsch» français", or "français „deutsch" français"? this quickly gets horrible, since quotation styles are so different in different languages. in practice not necessary to provide a structured arrangement, since there appear to be no strong rules anywhere about how quotations nest.

Note csquotes package *only* has an interface to quote "in the enclosed language"; if one wants to quote in parent language, one has to say:

```
\enquote{\foreignlanguage{inner language}{text...}}
```

Do we want to provide such an interface? Perhaps in a later revision, when we understand what we're doing. Block quotes come more difficult still: do we (as is common in French) want to provide an interface for author information?

Issue deferred.

2 Input encoding and font encoding

As things stand at present, there are strong relationships between both input and output encodings that are implemented within babel. A problem is that there is no unique font for some languages (e.g., many western languages can be expressed in OT1, T1 and LY1 at least). In such a case one might anticipate being tripped up by babel's present behaviour of switching you back to the "standard" encoding for a language, which may be what you were using to start with. Mem doesn't do this: it will restore the previous

language state. (Unfortunately, babel isn't uniform: some languages have an associated encoding, others don't.)

Could we have a set of alternate encodings for each language, and select the appropriate encoding to match language and available fonts? If so, when do we make the decision? And how do we optimise cases, such as where language 'a' may be represented in T2A or T2B, while language 'b' is represented in T2B or T2C.

One may imagine mathematics encodings per language, too. (We know there are differences in Russian and Arabic, at least; babel covers such changes.)

3 Music

We ignore music.

4 Ligatures

Existing ligatures should be suppressed/allowed in certain languages. Consider the SISISI project: could such things be adapted to suppressing ligatures when necessary? This is definitely a language-dependent thing, but probably impossible to deal with at the macro level, with present technology. Perhaps with (external?) OTPs?

Ligatures vs. characters (e.g., 'ij' is not typeset differently in Dutch, 'ö' being a "shift" in German but a different letter in Danish or Swedish[2].

Since TeX doesn't allow you to switch the ligature on/off, we can certainly model the distinction, but we can't implement it until some implementation becomes available (e.g., Omega, PDFTeX using OT fonts, etc.).

Dutch letter 'ij' should be in LICR, but (possibly?) isn't[3]; if not, could such an be added in a future release.

Mathversions per language? If change at language change, there need to be several mathversions, anyway.

Spacing in different languages' mathematics (Hungarian mathematics definitely has different settings)? Upright/italic setting of variables, constants (there exists a french tradition[4] that has them the other way around from that which tex traditionally implements). On the other hand, we don't want to overwhelm the user with optional settings.

5 Writing directions

agreed that we will not consider issue of one language, multiple alphabets or multiple writing directions. so Serbo-Croat/Latin is a different language from serbo-croat/cyrillic for the purposes of latex. similarly (if we ever got that far) mongolian in each writing direction/script would be three different languages.

[2] Have I got this right??

[3] Check ìj'in LICR!

[4] possibly not the only one

issue of margins: should left-right have different meaning as far as margins? what happens to the margins when the writing direction changes? not obviously sensible to change within a line (or even a paragraph) but where *do* we change?

however, in the case in a complete r-l document, since the first page of the document is on the left side, this (sort of) falls out in the wash.

what happens in multilingual books? two books starting at opposite (physical) ends and (sort of) stuck together are easy enough. short quotations of an r-l language in an overwhelmingly l-r text are common and unexceptional. parallel texts, and texts with alternating l-r and r-l regions, need some rather more careful.

in principle, one should traverse the whole document model, to search for instances of left-right bias. for example, bracketing of citations, \left, \right delimiter control. tabular specs and box alignments both include l and r specs; these ought to remain the same.

twocolumn, multicolumn, etc., need rethinking.

dealing with vertical-oriented text *could* be done (in a poor man's sort of way) by rotating paragraph boxes. is this a sensible thing to do?

good argument for rethinking the paragraph model, in the context of galleys and r-l typesetting.

6 collation sequences

we need not concern ourselves with the problem as it applies to indexing programs. however, even the simple matter of \alph does raise problems. some languages have this covered in babel, already, but there are others that don't. and of course, greek actually uses letters as numerals...

7 grouping properties

group properties into groups, for the purposes of enabling/disabling certain classes of language-dependent behaviour en bloc. is this a sensible idea? and if so what interface would one want to have?

we believe that languages don't necessarily have "absolute" properties, so it would be nice to select the properties, defined for a language, in manageable chunks. mem has this facility already. arbitrary groups are a real possibility. but offering large ranges of facilities lets the user confuse himself (and us, come the bug report).

```
\DeclareLanguageProperty{\cmd}[arg spec]
\SetLanguagePropertyValue{language}{\cmd}{action}
```

(commands with arguments in there?; action is a token string for cmd definition, used to create a \newcommand for execution at the time of language change. note we don't allow different argument profiles for the same command in different languages.

```
\SetLanguagePropertyValueMapping{\cmd}{<type or level>}
```

says change this property up to a particular level[5]. different names (\ChapterHeadingName for top level only, \ChapterRefName for all levels) will enable one to describe what the name is actually doing.

By contrast, mem uses modal language specification[6]:

```
\SetLanguage{french}
\DeclareLanguageProperty{\labelenumi}{layout.lists}
```

where the "layout.lists" is a tree structured thing, where you can select this particular issue either by "layout.lists" or just "layout".

User commands are:

```
\setlanguage{french}{layout.lists=ignore}
```

for global change to language spec for this document,

```
\begin{languageblock}[layout.lists=inherit]{french}
```

for a particular instance of french. value "ignore" means use top document level value (perhaps should be "document"), "inherit" means use value from enclosing language.

8 Dependencies

What is a language-dependent thing? What is a tradition?

Hyphenation, though there is an argument to claim that (for example) the "british" patterns are a fad of certain british publishers, only.

Object names (chapter, figure, ...).

Script/direction is a language attribute; we fix a language by the intersection of appropriate attributes.

Script is characterised by a character set, font encoding (possibly more than one) and writing direction. All three together specify input/output methods for the language. Hyphenation depends on the script and its language.

Mathematics representation is probably not necessarily different language–language, tradition–tradition. However, if we know that (say) we have a specific Cyrillic-based maths environment, we might care to establish that at the point where we change input encoding to something Cyrillic. Is this relevant to us? Would a user really want a document in several different languages, *and* have the mathematics format change as the language changed? Note, maths writing direction isn't necessarily inherited from the text writing direction: we know that left-right mathematics is (or has been) used in both Hebrew and Arabic. (There are moves to define an Arabic style of typesetting mathematics, so we must be aware we're not deciding about a fixed matter. See Xanthi preprints.)

Punctuation treatment seems in general to be a typographic tradition (even in French), though space before colon (and semicolon?) is

[5] do we need the "Value" part of that name?

[6] though this really isn't essential to the structure being defined

Quotation marks are style; somewhat script-related, but for example guillemets seem to be usable in German. Repetition of quote marks in French are typographic (tradition).

Date formats: widely different from language to language (and there's always the linguistically neutral ISO format).

Generated text is obviously language-dependent.

Lists seem to be culture-dependent, both bullets, and enumeration structures.

Ligatures seem to be language-dependent (though we can't do anything much about them, in today's world).

Collating sequences: language.

Conventions for emphasis: style.

Word order in generated strings (cf the example earlier, about captions): language.

Repeated hyphenation (at start of following line) and repeated quotes: style.

Open parenthesis (bracket, brace) order is writing-direction, and hence language, dependent.

Currency styles seem to be culturally dependent.

Number format: seem likely language dependent.

Paragraph parameters: hyphendemerits depends on the language (you get better formatting by mucking around with hyphendemerits when the languages excells in long compound words).

9 Summary

Seem to have a half-way split between language- and culture-dependent items. The process of combining a language specification and appropriate culture-dependent properties might be called "localisation".