latex-test
2025-05-28

Ken Moffat

1 Introduction, or my history with LaTeX

A frustrating but sometimes educational experience.

It is easy to forget that TeX is at heart an old-school programming language, with a lot of additional macros added over the years, and many different options. Like all programming languages, it takes a long time to achieve any level of competence.

One day in February 2014, somebody noticed that our Beyond Linux From Scratch (BLFS) build of texlive did not build all of the package (and so, anybody who began by installing the binary *install-tl-unx*, which was our practice, still had programs which were not built from source).

I had no experience of this [*insert profanities here*] typesetting system, and my initial attempts to try to use it found many examples which perhaps worked when they were posted, but did not work for me. Eventually, I found a few routines which gave me a little confidence that some of it worked.

Getting a working version of xindy to build was *fun*.

Eventually I came back to this, got more of it working, and eventually got it all working from source (although on one of my machines the *binary* version of ConTeXt failed - that CPU did not support some SSE options that the contributed binary used, but such is life and anyway we prefer to build from source!). That caused me to add a minimal test for ConTeXt, but I have now removed that, the move from mkiv ConTeXt to LuaMetaTex convinced me that I have no interest in trying to keep up with the developments at the context garden.

Along the way, I have discovered that I really dislike much of TeX itself, and LaTeX too is a bit problematic:

- The default fonts (Latin Modern and variants) are ugly in my opinion. I have no interest in mathematical formulae, only in humanities. For this document I am using xelatex so that I can use my system fonts. In particular the original fonts, and variants such as cm-unicode, sometimes make very strange choices for diacritical markings, particularly for those languages where a cedilla is described by wikipedia as a diacritical comma).

- It is a programming language, but very idiosyncratic and sometimes poorly documented – some parts are well documented, others less so, and the names you need to search for are not always the regular typesetting names. [1]

- All the additions in LaTeX etc. use macros, and many want to be loaded last. In practice, that means that they interact with each other.

- I had no use for outputs other than PDF files, so my starting point was using pdflatex to create PDFs, often with old markup, even though by that date lualatex and unicode could probably have worked well in almost everything I wanted to do.

- As with many things from the past, basic TeX predates unicode. There have been various language-specific versions in the past, but for me the least-painful TeX variant was xelatex. In 2023 I decided I preferred lualatex for new tests (where possible).

---

[1] http://mintaka.sdsu.edu/GF/bibliog/latex/gripe.html

However, as it pulls me further in I realise that for some things it is a tolerable system. But I hope never to reach the stage of loving it.

2 About this package

I aim to test **basic** PDF creation in latex and its derivatives (lualatex, xelatex) using texlive. The tests now also test dvisvgm.

The configure script is there to check that everything it requires is present, and then to check for fonts. This package does not install anything. If some texlive programs, or some fonts, or some latex packages, are missing it will reduce the tests in the Makefile. Where a program is missing (or its tests have been disabled using '–without') I aim to mention where it comes from. Two programs are symbolic links, I have guessed that lualatex comes from collection-basic and is therefore always present, and that xelatex comes from collection-xetex. Where all programs required for a test are present, I test for specific sty files to check if a collection is present. *BUT I do not check packages which are in collection-basic, nor do I check for packages in collection-latex if either collection-latexextra or collection-latex-recommended are needed for that test (those two collections pull in collection-latex).*

I intend that these tests can be used to check a binary install, e.g. during the texlive pretest phase. However, I develop my tests in a full install. Neither scheme-basic nor scheme-plain can run any of my tests. Using scheme-medium a few of these tests can be run, but not my initial pdflatex test which I regarded as very basic.

From scheme-medium I was able to eventually run all my initial tests:

- tlmgr update –self *(might be needed if the binary is not up to date)*

- tlmgr update –all *(to ensure current install is up to date)*

- tlmgr info collections *this is not needed if you know which collections you want, but it lets you check the available collections*

- tlmgr install collection-latexextra *(this also pulled in collection-pictures and collection-mathscience)*. At this point all of my tests except those using chinese, japanese and the biber program were able to run.

- tlmgr install collection-langarabic *(for bidipoem)*

- tlmgr install collection-langjapanese *(for a japanese font and for japanese vertical writing using the old uplatex and the current luatexja). To my surprise this also pulled in collection-langcjk which gave me the needed chinese fonts for my lualatex, biblatex and bibxelatex tests. If I had needed texlive korean fonts (I have system korean fonts) I could have installed collection-langkorean.*

- tlmgr install collection-fontsextra *(*to use texlive fonts for the latin, monotonic greek and cyrillic text in bibxelatex, lualatex and xelatex - instead of the system font I used.)

You can then run 'make' to run the tests. All being well, each will produce a PDF (or svg file) for you to examine in your preferred viewer. Alternatively, you can run the tests individually. Running multiple tests in parallel has NOT been tested.

I have now added lualatexbib which has slightly different dependencies.

Many of the examples are intended to be mildly humorous, and although the files might be useful as examples of how to use some things, they should not be taken as "good" examples, apart from those which I have copied wholesale!

The *Collection:* entries show which collections are needed to enable a test to run if using a binary install. For simplicity I pick one styleshet to represent a collection, and then test for it. I had previously tested for random stylesheets, guessing which represented the 'top of tree' package that pulled in others, but that meant I tested for a lot more stylesheets than I now do, and trying to specify individual packages to be added to a smaller scheme for running these tests ultimately seems like an exercise in pain.

3   What the various tests do

3.1   biblatex-biber

A bibliography using biblatex with the biber backend.
Programs: pdflatex (collection-latex), biber (collection-bibtexextra)
Collections: collection-latex, collection-latexextra, collection-latexrecommended

This uses "traditional" TeX markup for diacriticals and some "extra" latin letters such as Æ. Although I prefer to use UTF-8, this is ASCII with markup to get accents and diacriticals.

3.2   biblatex-bibtex

A bibliography using biblatex with the bibtex backend.
Programs: pdflatex (collection-latex), bibtex (collection-basic, so assumed present)
Collections: collection-latex, collection-latexextra, collection-latexrecommended

Essentially the same as *biblatex-biber*.

3.3   biblatex-unibiber

Biblatex and biber are a modern way of providing bibliographies and here they are used with UTF-8.
Programs: pdflatex (collection-latex), biber (collection-bibtexextra)
Collections: collection-latex, collection-latexextra, collection-latexrecommended

This version uses biber and biblatex with pdflatex. It is "limited" in the UTF-8 it will accept for bibliography keys, although plain (latin-alphabet) text seems fine.

3.4 **bibtexidx**

Bibtex and makeindex used with ASCII input.
Programs: pdflatex (collection-latex), bibtex and makeindex (both collection-basic, assumed present)
Collections: collection-latex, collection-latexrecommended

Having gone to the trouble of creating a document to use bibtex in the old-fashioned way, I decided to go the extra mile and use makeindex on it.

3.5 **bibxelatex**

Bibxelatex tests biber with xelatex as well as using xeCJK.
Programs: xelatex (collection-xetex), biber (collection-bibtexextra)
Collections: collection-langcjk (provides Fandol chinese fonts, I used system fonts for japanese), optionally collection-langjapanese for texlive japanese fonts), collection-latex, collection-latexextra, collection-latexrecommended
Requires: system or texlive OTF or TTF fonts for chinese and japanese, and optionally for korean: for texlive these can be provided by collection-langcjk, collection-langjapanese and collection-langkorean, also requires a system or texlive TTF or OTF font covering latin, greek and cyrillic (collection-fontsextra)

This began as an alternative way of testing a bibliography. Because it is using xelatex, it has no problem with using a cyrillic key to a bib entry, nor any accented latin. It also contains the bibliography within the tex file, using the filecontents package to write it to a bib file.

Along the way I came across the xeCJK package, with its ability to separate blocks of Unified Han unicode glyphs - in the example I copied, Kana (Japanese) and Hangul (Korean) were separated. This seemed like a good idea, but further experimentation showed that not all Japanese is Kana – the ideograms shared with Chinese are known as Kanji. Chinese fonts seem to contain some of these, but coverage varies from one font to another. It is possible to use a fallback font, which should mean that everything is rendered in some form, but that either means using a Chinese font with a Japanese fallback (some things might look odd to a Japanese reader), or the converse with some things looking odd to a reader of Simplified Chinese.

In practice, I do not expect anybody to need a real bibliography with multiple languages in it. For this test I have a new configure switch, *–prefer-* with values of japanese (the default) or chinese.

3.6 bidipoem

Bidi is for bidirectional text (LTR and RTL).
Programs: xelatex (collection-xetex)
Collections: collection-langarabic, collection-latex, collection-latexextra, collection-latexrecommended
Requires: suitable fonts, (for texlive use collection-fontsextra)

This uses XeLaTeX to typeset a piece of Omar Khayyam's poetry in both English and Persian. Uses an OTF/TTF font which will handle both languages, and uses the bidi and bidipoem packages. Also uses double backslashes (insert a newline) for English poetry (I do the same thing in ABOUT.tex to stop lines in certain languages from *excessively* overflowing).

3.7 blended-circles

A test using tikz to produce a colourful picture
Programs: pdflatex (collection-latex)
Collections: collection-pictures

These are blended circles (i.e. the main RGB circles change colour where they overlap). The blend option within Tikz was fairly new when I created this. It should work in a 2014 or later texlive install.

3.8 cylinder

This test uses asy for a 3-Dimensional drawing.
Programs: asy (collection-binextra), libgs (dynamic system lib), pdflatex (collection-latex)
Collections: collection-binextra, collection-latexrecommended

Draws a cylinder. This requires asymptote to have been compiled with freeglut, I do not know of a reliable way to test for that (so if it has not been compiled like that you will need to disable this test). While this is running, it draws on the screen (typically frames, sometimes junk), and then removes this at the end - I regard that as a feature of asy, but it is disconcerting when you first encounter it. This test is slow, as is the rendering of the resulting PDF. On some low-end AMD machines this test does not work, it only shows the heading and I do not know of any way to test that.

3.9 dvisvgm

A minimal test to create an svg image from a plain tex document
Programs: latex (collection-basic, assumed present), dvisvgm (collection-binextra)
Collections: collection-latex, collection-latexextra, collection-pictures

3.10 jalua

This uses lualatex to do vertical japanese typesetting.
Programs: lualatex (collection-basic)
Collections: collection-langjapanese, collection-latex, collection-latexextra, collection-latexrecommended
Requires: the HaranoAJi Gothic and Mincho OTF fonts from collection-langjapanese

3.11 ligatures

The HarfBuzz backend of LuaTex with LaTeX.
Programs: lualatex (collection-basic, assumed present)
Collections: collection-latex, collection-latexrecommended, collection-luatex
Requires: 'LinLibertine_R.otf' font (collection-fontsextra)

The HarfBuzz backend of Luatex can access features of OTF fonts, in this case ligatures (e.g. modified forms of 'fi' and 'fl'). This font has a rich set of ligatures for latin languages.

3.12 lualatex

LuaTeX with LaTeX.
Programs: lualatex (collection-basic, assumed present)
Collections: collection-latex, collection-latexextra, collection-latexrecommended, collection-luatex

Lualatex is able to cope with fontspec and UTF-8. I discovered that it uses luaotfload-tool to find *system* fonts, and runs that when it feels like it! So, my configure script runs that command to ensure that any OTF or TTF font which I have only just made known to fontconfig will be found.

The lualatex test itself provides a greeting in various languages using the latin, cyrillic, and monotonic greek alphabets. It will also do this for Simplified Chinese, Japanese, and Korean if it finds the fonts. It also generates a random number, because lua is a scripting language with many other uses.

3.13 lualatexbib

A major revision of the bibtexidx test to use lualatex.
The content has been rewritten to concentrate on the early years.
Programs: bibtex, luatex and makeindex (all collection-basic, assumed present), *Collections: collection-fontsextra collection-latex, collection-luatex*

Although I had initially intended to continue using Small Caps for personal names, in the end I decided that was not usefull for the revised indexing. But one of the links mentioned by Dodgson and which I now show includes Small Caps for the proceedings number. I did not wish to add extra dependencies, but CharisSIL is among the fonts pulled in by collection-fontsextra and I have been able to specify that as the SC font for DejaVuSerif using the old-style addition.

3.14 mosfet

Another tikz test, this time drawing an electronic circuit diagram. I have revised this to use modern electrolytic capacitor symbols, that apparently prevents the circuit being automatically tested because the old symbols had inverted polarities.
Programs: pdflatex (collection-latex)
Collections: collection-latex, collection-latexextra, collection-mathscience, collection-pictures

A conventional use for Tikz, but it requires further packages. This was why I started to use kpsewhich to check for the stylesheets. Nowadays I instead test for a single stylesheet per collection, so other tests which use xcolor.sty will look for tikz.sty even tough those tests do not use tikz.

3.15 oldjap

Another test of vertical Japanese writing, using UTF-8 but with the old uplatex package followed by dvipdfmx. Not configurable, it uses the Computer Modern font for any English text, with two OpenType japanese fonts which cannot be changed.
Programs: dvipdfmx (collection-basic, assumed present), uplatex (collection-langjapanese)
Collections: collection-langjapanese, collection-latex, collection-latexextras, collection-pictures
Requires: the HaranoAJi Gothic Medium and Mincho Regular OTF fonts from texlive (collection-langjapanese)

3.16 paren

A test to see if ruby works using one of the few remainiing ruby scripts in texlive.
Programs: match_parens (collection-binextra), pdflatex (collection-latex), ruby (system program)
Collections: collection-binextra and collection-latex are sufficient

This test checks if ruby is working, by running match_parens which is a script supplied with texlive. Ruby has not always been installed on my systems, and many of the ruby scripts previously supplied by texlive did not get used by my test files.

But match_parens is useful (it does what the name suggests – check that parentheses or 'curly brackets' are matched) and it also fails (as noted in the script) when run on itself. So, the test checks for the failure, and for success on the tex source which it will then use to produce a PDF to show all is well.

3.17 pdflatex

The most basic part of the tests in my opinion.
Programs: pdflatex (collection-latex)
Collections: collection-latex, collection-latexextra, collection-latexrecommended

This uses pdflatex to check sizing, left / centred / right alignments, produce an example formatted equation, and generate some of the usual lorem ipsum cod latin. It also uses colours.

3.18 periodicsvg

Create an svg file containing the periodic table of the elements.
Programs: latex (collection-basic, assumed present), dvisvgm (collection-binextra)
Collections: collection-latex, collection-latexextra, collection-latexrecommended,
collection-pictures

This creates an svg file for the periodic table of the elements, but with dvisvgm-3.0 the arrangement of the letters of 'Periodic' in the title is poor.

3.19 periodictab

This creates a PDF containing the periodic table of the elements, the original version (and without formatting problems) from which periodicsvg was adapted.
Programs: pdflatex (collection-latex)
Collections: collection-latex, collection-latexextra, collection-pictures

3.20 triangles

This test uses asy to make a 2-Dimensional drawing.
Programs: asy (collection-binextra), libgs (system dynamic lib), pdflatex (collection-latex)
Collections:: collection-binextra, collection-fontsrecommended

Draws coloured triangles.

3.21 xelatex

XeLateX is for UTF-8 and system OTF/TTF fonts.
Programs: xelatex (collection-xetex)
Collections: collection-latex, collection-latexrecommended
Requires: a system or texlive TTF or OTF font covering latin, monotonic greek and cyrillic (can be provided by collection-fontsextra), with OTF or TTF fonts for chinese, japanese and korean: for texlive those can be provided by collection-langchinese, collection-langjapanese and collection-langkorean

This is another way of using UTF-8 and system OTF/TTF fonts to produce a PDF. This test uses separate fonts for latin / cyrillic / greek, and (if found) Simplified Chinese, Japanese, and Korean.

Unfortunately, it seeems that xelatex is no-longer maintained.

3.22 xindy

Xindy, a modern way of indexing T$_{E}$X documents.
Programs: lualatex (collection-basic, assumed present), makeglossaries (collection-latexextra), pdfinfo (system) xindy (collection-binextra)
Collections: collection-binextra,collection-latex, collection-latexextra,i collection-latexrecommended

Because xindy is not built as part of the default texlive source build (it requires an existing latex and pdflatex to build it), I needed to prove that my build worked - in the end I managed to create an early version of this test which worked on the binary, but when I ran it on the self-built version it completed apparently normally, but without an index (that is, it failed).

Xindy was my motivation for creating these test files. I eventually found out what was wrong with the build, and now the test uses pdfinfo to check that the correct number of pages was produced. It uses lualatex (did I mention that I like UTF-8 ?).

I have now added the bookmark package, with hyperref, to provide links to the various parts. This works in epdfview (and yes, as of mid 2023 BLFS still builds a working epdfview), but not for several years in evince which can already list the contents in its own ways. As a side effect, the index links are now clickable (and that *does* work in evince). Of other viewers, mupdf does not provide a left panel for links (nor does it show where you are in the document), and for my purposes okular was similar to evince when I last built okular, but I have not used okular recently. Navigation from the side panel also works when using firefox as the viewer.

3.23 xindynonlua

This is intended to be a more-widely applicable way of testing xindy.
Programs: pdfinfo (system), pdflatex (collection-latex), texindy (collection-binextra)
Collections: collection-binextra, collection-latex, collectionlatexextra,
collection-latexrecommended

I discovered that while xindy worked fine on my x86_64 builds, on i686 it did not work. This test is offered in the hope that it might be useful (no UTF-8, no luatex, so fewer dependencies). It is a cut-down version of the xindy test, using traditional TEX markup. I also dropped the references to Æesir and Þórr because they were now indexed under E and O instead of A and after Z, which may have been because I used incorrect markup. More to the point, the Makefile invokes xindy directly instead of using makeglossaries. This makes it easier to edit the Makefile to add debug switches. After a lot of head-banging, I found the problem hidden among the error messages - xindy has become sensitive to how it is built, perhaps as a result of changes to coreutils.

Please note that it is many years since I stopped building or testing anything on i686.

3.24 Tests not run by default

When I was trying to make sense of the maze of perl modules we referenced in BLFS-8.3, mostly added for biber, I discovered that if we changed our location of ca-certificates to one known to perl core module HTTP::Tidy we could drop Mozilla::CA from the dependencies of biber: it is only a fallback intended for CPAN installs.

But I then needed to find a testcase. I eventually discovered online a couple of MWEs to use with biber when accessing remote bibliographies. When last tested those still usually worked, but no guarantees that the links will remain. I have now altered them to live on the LFS website.

3.24.1 remote1

Access to a remote bibliography which happens to be in latin1 encoding.
Programs: pdflatex (collection-latex), biber (collection-bibtexextra)

The first example I found is described as latin1 when downloaded. That needs an extra switch in biber.

3.24.2 remote2

Access to a remote bibliography in UTF-8 encoding.
Programs: pdflatex (collection-latex), biber (collection-bibtexextra)

This example downloads as UTF-8 so it is straightforward for biber to use it.

4   What is NOT tested

To be honest, *most* of texlive.

Even if I test something, it is only with the options in my testsuite. If you use texlive for real, you need to maintain your own .tex files to be able to confirm that a new version has no regressions for you.

One of the things I do not normally test is monospaced fonts (although, arguably, CJK fonts are monospaced). Here is a listing of a program which used to get refer- enced a lot in my JCL when I was an application programmer, IEFBR14. Taken from http://en.wikipedia.org/wiki/IEFBR14 which is as good an explanation as any of why things need to be tested. Thanks to John Pershing for documenting this.

```
IEFBR14 START
        USING IEFBR14,15  Establish addressability
        B     GO          Skip over our name
        DC    AL1(L'ID)    Length of name
ID      DC    C'IEFBR14'   Name itself
        DS    0H           Force alignment
GO      SR    15,15        Zero out register 15
        BR    14           Branch to return addr in R14
        END   IEFBR14
```

That is about the only use I ever expect to have for monospaced fonts in texlive, apart from when I examine which languages a TTF or OTF font can support. Getting it to render sanely was unpleasant. I suppose that Knuth expects monospaced text to only be used for short commands.

5   dependencies

5.1   Required system dependencies

The configure script is targetted at BLFS, but I have tried to make it system-agnostic. All of the following are required, and tested in this order:

bash : I use 'type -pa' to identify programs.

Once I started to test on FreeBSD-10.1 I was going to change this, but /bin/sh there appears to be a csh variant and code which supposedly works in POSIX sh did not work. But after I had installed Xorg, a desktop, and texlive, I found that bash was present at /usr/local/bin/bash. So, bash is required.

cat : a basic tool. In practice, if this is missing then configure will probably give a messy error as soon as it starts, before running the tests to check for the required programs.

find : a basic tool, used to look for the T$_E$Xfonts

grep : a basic tool.

make : a basic tool.

In the early days I tested FreeBSD in a VM, using xfce. The version of 'make' which was present worked, but since then I have added new tests and I might have made this compatible only with gnu make ('gmake' on BSDs).

rm : a basic tool.

sed : for portability, I do not use in-place edits.

tr : another basic tool, used to look for texlive's TTF/OTF fonts and in testing for programs,

wc : another basic tool, used in testing for programs (needed if /bin is symlinked to /usr/bin).

kpsewhich : I now use this to check for example stylesheets identifying which collection has been used (for testing binaries with a less than full install).

5.2   Optional dependencies

These are used by various tests. If they are not present, the test will not be enabled in the Makefile. With the exceptions of *ruby*, *fc-list*, *libgs*, and the programs from *poppler* they are all part of a **full** texlive install.

pdflatex : I used to make this required, but not everything needs it.

asy : for the *cylinder* and *triangles* tests.

biber : for the *biblatex-biber*, *biblatex-unibiber* and *bibxelatex* tests.

bibtex : for the *biblatex-bibtex*, *bibtexidx* and *lualatexbib* tests.

dejavu-otf : for the *lualatexbib* test.

dvipdfmx : for the *oldjap* test.

dvisvgm : for the *dvisvgm* test.

fc-list : required to help find system OTF or TTF fonts if either *lualatex* or *xelatex* (or tests using them) will be run. Its absence will mean that system fonts cannot be used, but I think I disable all the users if that happens, to keep the logic in *configure* somewhat manageable. In practice, it is a dependency of Xorg so it should always be present.

kpsepath : used to look for texlive OTF/TTF fonts if either *lualatex* or *xelatex* (or tests using them) will be run.

latex : for *dvisvgm*

libgs : for the *asy* tests.

lualatex : for *ligatures*, *lualatex*, *lualatexbib* and *xindy*.

luaotfload-tool : used, if present, to ensure the *lualatex* and *xelatex* tests will find system fonts when configure runs.

makeglossaries : for *xindy*.

makeindex : for *bibtexidx* and *lualatexbib*.

match_parens : for the *parens* test.

pdfinfo : (from poppler) is needed to check that the *xindy* test worked, it is unfortunately common for a broken build of xindy to let the test complete without error but not produce the index page.

pdftotext : (from poppler) is needed to check that the *biber* tests worked.

ruby : for the *parens* test.

TikZ : (also known as pgf) - a drawing tool, used for *blended-circles* and with further packages for *mosfet*.

uplatex : for the *oldjap* test.

xelatex : for the *bibxelatex*, *bidipoem* and *xelatex* tests.

xindy : for the *xindy* and *xindynonlua* tests.

5.3 Required LaTeX packages

I eventually realised that I can, and should, use *kpsewhich* to confirm that the packages used by each of my TeX scripts are present. After *configure* has checked for programs, it now checks for example stylesheets to identify which collections of a binary install are needed.

5.4 Optional OTF/TTF font dependencies

For those tests which use fontspec (all tests using lualatex or xelatex), configure will look for usable fonts. To find texlive fonts it originally used *mkluatexfontdb* because that existed on my x86_64 installs where it updated a cache in $HOME. Unfortunately, that command did not exist on i386 binary versions of texlive. Now I use *kpsepath* and *tr* to find the *textmf-dist/fonts/* directory, and then run *find* there.

I also use *fc-list*. Here, I list the order in which configure searches. I assume that anybody using this will have at least decent serif fonts for Latin, Cyrillic, and modern Greek. For CJK, the order of searching assumes that serif fonts are prefered to sans, (for Chinese, Kai script fonts come between Serif and Sans), and for Japanese and Korean it will use appropriate pan-CJK fonts (technically, those are Chinese) as a fallback in order to try to provide the maximum coverage. Newer, maintained, fonts are usually preferred to older unmaintained. You can alternatively specify system fonts using configure switches to override the process. For system (fontconfig/freetype) fonts I am listing the file name in case anybody wants to track these down for their own use.

The output from configure shows the name of the font it has chosen: For texlive fonts this is the filename, for system fonts it is the name by which fontconfig knows it.

For a main font for Latin, Greek, Cyrillic alphabets:

- Linux Libertine O (texlive *LinLibertine_R.otf*)

- DejaVu Serif (texlive or system, *DejaVuSerif.ttf*)

- FreeSerif (texlive or system, *FreeSerif.otf*)

The texlive fonts come from collection-fonts-extra.

For a Simplified Chinese font:

- FandolSong (texlive *FandolSong-Regular.otf*)

- AR PL UMing CN (system, *uming.ttc*)

- AR PL Sungtil GB (texlive or system, *gbsn00lp.ttf*)

- AR PL New Sung (system, *odosung.ttc*)

- AR PL UKai CN (system, *ukai.ttc*)

- AR PL KaitiM GB (texlive or system, *gkai00mp.ttf*)

- AR PL New Kai (system, *odokai.ttf*)

- FandolHei (texlive *FandolHei-Regular.otf*)

- Noto Sans CJK SC (system, *NotoSansCJKsc-Regular.otf*)

- Odo Hei (system, *odohei.ttf*)

- WenQuanYi Zen Hei (system, *wqy-zenhei.ttc*)

- Droid Sans Fallback (system, *DroidSansFallback.ttf*)

- Droid Sans Fallback Full (system, *DroidSansFallbackFull.ttf*)

The texlive fonts come from collection-langchinese. Droid Sans Fallback Full is no-longer available anywhere.

For a Japanese font:

- HaranoAjiMincho-Regular (texlive, *HaranoAjiMincho-Regular.otf*)

- IPAexMincho (texlive or system, *ipaexm.ttf*)

- IPAMincho (texlive or system, *ipam.ttf*)

- HanaMinA (system, *HanaMinA.ttf*)

- Kochi Mincho (system, *kochi-mincho-subst.ttf*)

- HaranoAjiGothic-Regular (texlive, *HaranoAjiGothic-Regular.otf*)

- IPAexGothic (texlive or system, *ipaexg.ttf*)

- IPAGothic (texlive or system, *ipag.ttf*)

- VL Gothic (system, *VL-Gothic-Regular.ttf*)

- Noto Sans CJK JP (system, *NotoSansCJKjp-Regular.otf*)

- Kochi Gothic (system, *kochi-gothic-subst.ttf*)

- Sawarabi Gothic (system, *sawarabi-gothic-medium.ttf*)

- WenQuanYi Zen Hei (system, *wqy-zenhei.ttc*)

- Droid Sans Fallback (system, *DroidSansFallback.ttf*)

- Droid Sans Fallback Full (system, *DroidSansFallbackFull.ttf*)

The texlive fonts come from collection-langjapanese.

Note: in texlive 2015, although Sawarabi Mincho worksed in xelatex, lualatex was unable to find it. Therefore I did not include it in the tests. I no-longer install it, so not tested. Droid Sans Fallback Full is no-longer available anywhere.

For a Korean font:

- UnGungseo (texlive or system, *UnGungseo.ttf*)

- UnBatang (texlive or system, *UnBatang.ttf*)

- NanumMyeongjo (system, *NanumMyeongjo-Regular.ttf*)

- Baekmuk Batang (texlive or system, *batang.ttf*)

- UnDotum (texlive or system, *UnDotum.ttf*)

- UnGraphic (texlive or system, *UnGraphic.ttf*)

- NanumGothic (system, *NanumGothic.ttf*)

- Baekmuk Gulim (texlive or system, *gulim.ttf*)

- Baekmuk Dotum (texlive or system, *dotum.ttf*)

- Noto Sans CJK KR (system, *NotoSansCJKkr-Regular.otf*)

- Noto Sans CJK JP (system, *NotoSansCJKjp-Regular.otf*)

- Noto Sans CJK SC (system, *NotoSansCJKdc-Regular.otf*)

- Noto Sans CJK TC (system, *NotoSansCJKtc-Regular.otf*) Unusually, all four variants cover korean with identical glyphs.

- WenQuanYi Zen Hei (system, *wqy-zenhei.ttc*)

- Droid Sans Fallback (system, *DroidSansFallback.ttf*)

Note: Droid Sans Fallback Full stopped including Hangul in 2012.

For use in bidipoem, to produce output in *both* English and Persian:

- Persian Modern (this used to be in texlive when I started these tests)

- FreeSerif (texlive or system, q.v.)

- DejaVu Sans (texlive or system, *DejaVuSans.ttf*)

Note: neither *nazli* nor *FreeFarsi* provide the apostrophe and fancy double-quotes used by the English text in my example.

6 reducing what will be tested

At times (e.g. when I have known that a binary is broken) I find it useful to restrict which things will be tested. But mostly I have used these to check the logic of my changes to the configure script. Normally, these switches should not be required.

If a –without switch is used, it should report that each affected test has been dese-lected.

- –without-allpdflatex : do not run any tests which use pdflatex (as well as my pdflatex test these include biblatex-biber, biblatex-bibtex, biblatex-unibiber, bib-texidx, blended-circles, cylinder, mosfet, paren, pdflatex, periodictab, triangles, xindynonlua and the two remote tests.

- –without-asy : do not test for asy, disables the cylinder and triangles tests.

- –without-biber : do not test for biber, disables the biblatex-biber, biblatex-unibiber and bibxelatex tests.

- –without-bibtex : do not test for bibtex, disables the biblatex-bibtex and bibtexidx tests.

- –without-bibxelatex : disable the bibxelatex test.

- –without-bidipoem : disable the bidipoem test and do not test for poem font.

- –without-chinese : do not test for a simplified chinese font, removes simplified chinese text from the lualatex and xelatex tests and disables the bibxelatex test.

- –without-cylinder : disable the cylinder (3D asymptote) test - use this if asy exists but has not been linked against freeglut or if system dynamic libgs is not installed

- –without-dvisvgm : disable the dvisvgm and periodicsvg tests.

- –without-fonts : do not test for OTF/TTF fonts, disables the bibxelatex, bidipoem, jalua, ligatures, lualatex, oldjap, xelatex and xindy tests.

- –without-japanese : do not test for a japanese font, removes japanese text from the lualatex and xelatex tests and disables the bibxelatex test.

- –without-korean : do not test for a korean font, removes korean text from the bibxelatex, lualatex and xelatex tests.

- –without-ligatures : disable the ligatures test.

- –without-lualatex : do not test for lualatex, disables the jalua, ligatures, lualatex and xindy tests.

- –without-makeindex : this disables the bibtexidx test, makeindex should always be present.

- –without-pdflatex : disable my pdflatex test.

- –without-periodicsvg : disable the test to create an svg file of the periodic table of the elements.

- –without-periodictab : disable the test to create a PDF of the periodic table of the elements.

- –without-ruby : do not test for ruby or match_parens, disables the parens test.

- –without-tikz : do not test for tikz, disables anything using collection-pictures: blended-circles, dvisvgm, mosfet, oldjap, pdflatex, periodicsvg and and period-ictab.

- –without-triangles : disable the triangles (2D asymptote) test. Use this if asy exists but has not been linked against freeglut or if system dynamic libgs is not installed.

- –without-xelatex : do not test for xelatex, disables the bibxelatex, bidipoem and xelatex tests.

- –without-xindy : do not test for xindy, disables the xindy and xindynonlua tests.

7 specifying OTF/TTF fonts

For the tests which use fontspec (i.e. any of the tests which use lualatex or xelatex), I look for some known fonts. It is possible to override these tests, either to test a font which is normally ignored because a "better" font has been found amongst the texlive fonts or on the system, or to use a completely different font. However, the ligatures test relies on a specific font which is part of texlive.

If configure cannot find suitable fonts, it will deselect some, or all, of the test. Thus people who do not have any CJK fonts can nevertheless see *some* results from the lualatex and xelatex testsi (but not from the bibxelatex test) if they have a *suitable* font for the latin, cyrillic, and monotonic greek alphabets.

To use any of these switches you need to specify the filename of tfe font, e.g. Sillyfont-Regular.otf. Both system fonts and texlive fonts should work. You *must* use the (full) filename, e.g. Sillyfont-Regular.otf, If the filename contains spaces you might be able to use it by putting it in quotes.

- –with-main-font=SomeFont.ttf : specify the font to use for text in the latin, monotonic greek, and cyrillic alphabets. This font is not used for bidipoem, but for the other uses it needs to have good coverage – as well as the DejaVu and Free Sans/Serif fonts, Liberation fonts also work.

- –with-chinese-font="SomeFont.ttf" : my Chinese text is all in Simplified Chinese, so a Traditional font is probably not a good idea. All of my chinese text is simplified.

- –with-japanese-font="SomeFont.ttf" : use this for Japanese text - I assume that my examples may include both Kana and Kanji, so most Chinese fonts are unlikely to include all the required codepoints, quite apart from whether or not they look "strange" to someone who reads Japanese.

- –with-korean-font="SomeFont.ttf" : Modern Korean hardly-ever uses Unified Han ideograms, so a separate Korean font is often required to show Korean text.

- –with-poem-font="SomeFont.ttf" : Use this font for bidipoem which contains English and Persian text. Experiments with inadequate fonts indicate that one or other language may be omitted, and some fonts show the English text with empty boxes where the Persian should be, or vice versa.

- –prefer=japanese or –prefer=chinese : This can be used for bibxelatex, to determine which language is treated as the main one (the default is –prefer=japanese), the other will be used as the fallback. Everything using Han or Kanji glyphs in the preferred language should look reasonable to a reader, anything in the other language might appear in a mixture of Chinese and Japanese style. However, if a pan-CJK font was selected, everything in Kanji will probably look Chinese.

8 Examples using different fonts

People who know me will recall that I am interested in being able to render as much as possible of what I encounter on the net, even if I cannot read it. I am also interested in the different additions to the latin and cyrillic alphabets. The following examples are, if my luck has held, the first article of the Universal Declaration Of Human Rights, sometimes using different fonts to get the coverage.

For some of these languages I am forcing new lines in the text (with double backslashes at the end of the line) to get it to fit within the normal printable width of the page. I am sure that there are better ways, at least for *some* of these languages. Except where noted, these are all in the DejaVu Sans font.

For attribution, see the References. Most were found at http://www.ohchr.org/EN/UDHR/Pages/Introduction.aspx

N.B. for languages such as Latvian, Maltese and Romanian it is so much easier to do this in lualatex or xelatex with UTF-8 : No additional packages needed.

8.1 Abkhasian [3]

Дарбанзаалак ауаҩы дшоуп ихы даҟәиҭны. Ауаа зегь зинлеи патулеи еиҟароуп. Урҭ ирымоуп ахшыҩи аламыси, дара дарагь аешьеи аешьеи реиҧш еизыҟазароуп.

8.2 Adja [2]

Agbetɔwo pleŋu vanɔ gbɛmɛ ko vovoɖeka gbeswɛgbeswɛ, sɔto amɛnyinyi ko acɛwo gomɛ; wo xɔnɔ susunywin ko jimɛnywi so esexwe. Wo ɖo a wɛ nɔvi ɖaɖa wowo nɔnɔwo gbɔ.

8.3 Armenian [9]

Բոլոր մարդիկ ծնվում են ազատ ու հավասար իրենց արժանապատվությամբ ու իրավունքներով։ Նրանք օժտված են բանականությամբ ու խղճով և միմյանց պետք է եղբայրաբար վերաբերվեն

8.4 Cantonese *using Noto Sans CJK TC* [17]

人人生出嚟就係自由嘅，喺尊嚴同權利上一律平等。佢哋具有理性同良心，而且應該用兄弟間嘅關係嚟互相對待

8.5 Catalan [15]

Tots els éssers humans neixen lliures i iguals en dignitat i en drets. Són dotats de raó i de consciència, i han de comportar-se fraternalment els uns amb els altres.

8.6   Chinese (Simplified) *using Fandol Hei* [9]

人人生而自由，在尊严和权利上一律平等。他们赋有理性和良心，并应以兄弟关系的精神互相对待

8.7   Chinese (Traditional) *using Noto Sans CJK TC* [18]

人人生而自由，在尊嚴和權利上一律平等。他們賦有理性和良心，並應以兄弟關係的精神互相對待

8.8   Croatian [6]

Sva ljudska bića rađaju se slobodna i jednaka u dostojanstvu i pravima. Ona su obdarena razumom i sviješću i treba da jedno prema drugome postupaju u duhu bratstva.

8.9   Franco-Provençal or Arpitan [1]

Tsecoun poú chè prèvâli dè tui lè drouê è dè tóte lè libèrtâ proclamâ in sta Déclarachion, chin ócóna distinchion, dè rache, dè cólóou, dè sèxe, dè linvoua, dè relijion, d'opinion poleteca ou bïn dè tóte j'âtre j'opinion d'orejena nachionala ou sociala, dè fórtóna, dè nèchance ou bïn dè tóte j'âtre setoachion.

8.10   Hungarian [9]

Minden. emberi lény szabadon születik és egyenlő méltósága és joga van. Az emberek, ésszel és lelkiismerettel bírván, egymással szemben testvéri szellemben kell hogy viseltessenek.

8.11   Icelandic [12]

Hver maður er borinn frjáls og jafn öðrum að virðingu og réttindum. Menn eru gæddir vitsmunum og samvizku, og ber þeim að breyta bróðurlega hverjum við annan.

8.12   Japanese *using VL Gothic* [13]

すべての人間は、生まれながらにして自由であり、かつ、尊厳と権利とについて平等である。人間は、理性と良心とを授けられており、互いに同胞の精神をもって行動しなければならない。

8.13 Korean *using UnDotum* [11]

모든 인간은 태어날 때부터 자유로우며 그 존엄과 권리에 있어 동등하다. 인간은
천부적으로 이성과 양심을 부여받았으며 서로 형제애의 정신으로 행동하여야 한다.

8.14 Maltese [4]

Il-bnedmin kollha jitwieldu ħielsa u ugwali fid-dinjità u d-drittijiet. Huma mogħnija
bir-raġuni u bil-kuxjenza u għandhom iġibu ruħhom ma' xulxin bi spirtu ta' aħwa.

8.15 Polish [9]

Wszyscy ludzie rodzą się wolni i równi pod względem swej godności i swych praw.
Są oni obdarzeni rozumem i sumieniem i powinni postępować wobec innych w duchu
braterstwa.

8.16 Portuguese [14]

Todos os seres humanos nascem livres e iguais em dignidade e em direitos. Dota-
dos de razão e de consciência, devem agir uns para com os outros em espírito de
fraternidade.

8.17 Romanian [9]

This uses below-commas on s and t, not cedillas

Toate ființele umane se nasc libere și egale în demnitate și în drepturi. Ele înzestrațe cu
rațiune și conștiință și trebuie să se comporte unii față de altele în spiritul fraternității.

8.18 Russian [9]

Все люди рождаются свободными и равными в своем
достоинстве и правах. Они наделены разумом и совестью и должны поступать
в отношении друг друга в духе братства.

8.19 Serbian (Cyrillic) [5]

Сва људска бића рађају се слободна и једнака у достојанству и правима. Она
су обдарена разумом и свешћу и требаједни према другима да поступају у духу
братства.

8.20 Turkish [7]

Bütün insanlar hür, haysiyet ve haklar bakımından eşit doğarlar. Akıl ve vicdana sahip-tirler ve birbirlerine karşı kardeşlik zihniyeti ile hareket etmelidirler.

8.21 Ukrainian [8]

Всі люди народжуються вільними і рівними у своїй гідності та правах. Вони наділені розумом і совістю і повинні діяти у відношенні один до одного в дусі братерства.

8.22 Vietnamese [10]

Tất cả mọi người sinh ra đều được tự do và bình đẳng về nhân phẩm và quyền lợi. Mọi con người đều được tạo hóa ban cho lý trí và lương tâm và cần phải đối xử với nhau trong tình anh em.

8.23 Walloon [16]

Tos lès-omes vinèt-st-å monde lîbes, èt so-l'minme pîd po çou qu'ènn'èst d'leu dignité èt d'leus dreûts. I n'sont nin foû rêzon èt-z-ont-i leû consyince po zèls, çou qu'èlzès deût miner a s'kidûre onk' po l'ôte tot come dès frés.

References

[1] Alphonse Dayer

[2] Commission béninoise des Droits de l'Homme, Benin

[3] Office of the High Commissioner for Human Rights - Georgia Field Office

[4] Maltese University, Malta

[5] Permanent Mission of the Federal Republic of Yugoslavia to the United Nations Office in Geneva

[6] Permanent Mission of the Republic of Croatia to the United Nations Office in Geneva

[7] Secretariat of the Human Rights Coordinating Committee, Turkey

[8] Ukrainian Union of Jurists, Ukraine

[9] United Nations Department of Public Information, NY

[10] United Nations Development Programme, Viet Nam

[11] United Nations Educational, Scientific and Cultural Organization, Paris

[12] United Nations Information Centre, Denmark

[13] United Nations Information Centre, Japan, and the Ministry of Justice and Human Rights NGOs

[14] United Nations Information Centre, Portugal

[15] United Nations Information Centre, Spain

[16] Waremme - Ville de la Paix, Wallonie, Belgique

[17] http://lingwiki.com/index.php?title=Cantonese

[18] http://www.omniglot.com/udhr/sinotibetan.htm