Chinese

in ConTEXt

1 Introduction

Most TeX distributions come with at least the Computer Modern Roman fonts, which means that you can start typesetting right away. However, in order to use TeX for typesetting Chinese, you should setup Chinese font properly.

In the process of typesetting text, TeX only has to know the dimensions of glyphs and the way they are kerned. These are collected in so called `tfm` files, TeX Font Metrics. It is up to the DVI postprocessor to produce a format suitable for printing. When using DVIPS or similar postprocessors, you (preferable) need so called Type 1 fonts, stored in files with the suffix `pfb`.

Today most Chinese outline fonts are available in the TrueType format, that can be recognized by their suffix `ttf`. These `ttf` files can be converted into `pfb` files by programs like `chpfb`, formerly known as `gbpfb`, written by Chang Xiangyang. Werner Lermberg modified this program to `ttf2pfb`, that is part of `freetype` package. His version converts `ttf` files into a collection of fonts suitable for use in TeX, for which he wrote his CJK package.

In October 1999, CONTEXT was extended to support Chinese. Since Chinese support was written from scratch, and since PDFTeX was used while testing, Wei Lang and Hans Hagen explored the possibilities to use `pfb` as well as `ttf` fonts. Since PDFTeX needs an encoding vector, Hàn Thế Thành, the author of PDFTeX, patched `ttf2pfb` to print the font index to `enc` files so that we can now directly use the TrueType font in PDFTeX. In a similar fashion, Hàn Thế Thành adapted `ttf2afm`.

This means that we can use `ttf` as well as `pfb` fonts. The big advantage if directly using `ttf` files, is that each font is packaged in one file, while in `pfb` format the font is split in many small files. Both alternatives need `tfm` files with at most 255 characters, which for Chinese fonts with thousands of glyphs, means that we have many `tfm` files per font. There is no way around this, since the `tfm` format is frozen and limited.

When you print PDF files with TrueType fonts on a POSTSCRIPT printer from Acrobat, you can best print them as Level 1 files, otherwise characters may not show up.

We expect Chinese fonts to become part of the major TeX distributions, in which case users don't have to bother about installation details. However, in this manual we will describe how to install the fonts manually. Although testing Chinese TrueType support in PDFTeX took place with CONTEXT, the installation process described here also applies to LaTeX.

君不見，黃河之水天上來，奔流到海不復回。君不見，高堂明鏡悲白髮，朝如青絲暮成雪。
人生得意須盡歡，莫使金樽空對月。天生我材必有用，千金散盡還復來。烹羊宰牛且為樂，
會須一飲三百杯。岑夫子，丹丘生，將進酒，君莫停。與君歌一曲，請君為我側耳聽。鐘鼓饌
玉不足貴，但願長醉不願醒。古來聖賢皆寂寞，惟有飲者留其名。陳王昔時宴平樂，鬥酒十
千恣歡謔。主人何為言少錢，徑須沽取對君酌。五花馬，千金裘，呼兒將出換美酒，與爾同
銷萬古愁。

Intermezzo 1   李白：將進酒

2 Preparations

Before we can prepare the fonts, we first need to obtain the `freetype` package. The lastest sources of this package can be downloaded from:

`ftp://ftp.freetypw.org/pub/devel/`

The `gbpfb` program, or better its successor `chpfb` which also supports traditional Chinese Big5 encoded fonts, is distributed as part of `TurboLinux`. You can get the `chpfb-*.src.rpm` archive from:

`ftp://ftp.turbolinux.com.cn/pub/turbolinux/TurboLinuxC-*/SRPMS/`

Note that this is a rpm package. The `ttf2pfb` program part of this archive cannot produce the `enc` files needed for `ttf2afm`. The full function version can be fetched from:

`..... package site ...`

This archive also has two two unix shell scripts written by Wang Lei: `gbenc` and `b5enc`. These can be used to generate the encoding files.

Next we assume that all the archives mentioned are fetched and unpacked. Now the programs need to be compiled according to their installation guides located in the top of their source tree. The MS WINDOWS version of `chpfb` is contributed by He Qiang and contains precompiled binaries. Other MS WINDOWS binaries can be found in Fabrice Popineau's FPTEX.

If you are a Chinese MS WINDOWS user, the fonts needed are usually present on your system and can be found in the font directory. Some Linux distributions also include Chinese TrueType fonts. For example, the `TurboLinux` distribution has four huge Chinese TrueType fonts, `htsong.ttf`, `hthei.ttf`, `htkai.ttf` and `htfs.ttf`, each

in GBK simplified Chinese and Big5 traditional Chinese encoding. In the following section, we will use the file `htsong.ttf` to generate the `Song` face simplified Chinese fonts and the `Ming` face traditional Chinese fonts.

3 Making fonts

Because the Chinese font have a large number glyphs, for TeX these big fonts must be split it many subfonts. Each subfont must have at most 256 glyphs. When we use the following method to generate those subfonts, we can choose between two methods.

The first method suits the so called 'CJK standard encoding', in which each subfont has 256 glyphs and named

`<subfont name><01-94>`

for GBK encoding, as used for simplified Chinese, or

`<subfont name><01-55>`

for Big5 encoding that is used for traditional Chinese. Yet another range is used when we use the GB2312 encoding for simplified Chinese:

`<subfont name><01-35>`

The second method can be classified as `Poorman's Chinese Encoding`. Here, each subfont contains the glyphs that have the same first byte —keep in mind that Chinese fonts needs two bytes to identify a glyph—, and are named

`<subfont name><the hex number of first byte>`

Each subfont contains 190 glyphs for GBK simplified Chinese and 160 glyphs for Big5 traditional Chinese. This means that fonts organized this way way will waste many slots. On the other hand, glyphs in each subfont have the same first byte, which makes them easy to handle, especially because the first byte falls outside the ASCII range. This means that we can mix for instance English and Chinese quite comfortably. We will therefore use the second method.

4 Generating Chinese Type 1 fonts

In order to generate the `pfb` fonts from the `ttf` fonts, go to the `chpfb` directory. Since we will convert the Song face, used for simplified Chinese, we now say:

> 晉太原中，武陵人，捕魚為業，緣溪行，忘路之遠近。忽逢桃花林，夾岸數百步，中無雜樹，芳草鮮美，落英繽紛，漁人甚異之；復前行，欲窮其林。林盡水源，便得一山，山有良田美池桑竹之屬，阡陌交通，雞犬相聞。其中往來種作，男女衣著，悉如外人；黃髮垂髫，並怡然自樂。見漁人，乃大驚，問所從來，具答之，便要還家，設洒殺雞作食，村中聞有此人，咸來問訊。自雲先世避秦時亂，率妻子邑人，來此絕境，不復出焉；遂與外人間隔。問今是何世，乃不知有漢，無論魏、晉。此人一一為具言所聞，皆嘆惋。余人各復延至其家，皆出洒食。停數日辭去，此中人語雲：“不足為外人道也！”既出，得其船，便扶向路，處處志之。及郡下，詣太守說此。太守即遣人隨其往，尋向所志，遂迷不復得路南陽劉子驥，高士也，聞之，欣然規往，未果，尋病終。後遂無問津者。

Intermezzo 2　陶淵明（南北朝）：《桃花源記》

```
gbpfb [-c] htsong.ttf song
```

Here the `-c` is optional and used to select between 'CJK standard encoding' or 'Poorman's Chinese Encoding'. When the font generation is finished, you will have 126 `pfb` fonts which names like `gbsong81.pfb`, `gbsong82.pfb` upto `gbsongfe.pfb` and 252 `tfm` files. If you do not provide the `-c` option, you will get both the normal and the slanted alternatives.

In a similar way, for generating the Ming face, as used for traditional Chinese fonts, we say:

```
b5pfb [-c] htsong.ttf ming
```

Here you will get 89 `pfb` and 178 `tfm` files. When font generation is finished, you should move the files to their location in the `texmf` tree, preferable in local tree. The `pfb` and `tfm` files will for instance go into:

```
texmf-local/fonts/type1/chinese/
texmf-local/fonts/tfm/chinese/
```

We also need to create a map file (for instance) named `chinese.map`. To this map file, the following lines need to be added:

```
gbsong81    GB-Song81                          <gbsong81.pfb
gbsong82    GB-Song82                          <gbsong82.pfb
........
gbsongfe    GB-Songfe                          <gbsongfe.pfb

gbsongsl81  GB-Song81    ".167 SlantFont"  <gbsong81.pfb
gbsongsl82  GB-Song82    ".167 SlantFont"  <gbsong82.pfb
```

```
..........
gbsongslfe  GB-Songfe    ".167 SlantFont"  <gbsongfe.pfb

b5minga1    BIG5-Minga1                     <b5minga1.pfb
b5minga2    BIG5-Minga2                     <b5minga2.pfb
.........
b5mingf9    BIG5-Mingf9                     <b5mingf9.pfb

b5mingsla1  BIG5-Minga1  ".167 SlantFont"  <b5minga1.pfb
b5mingsla2  BIG5-Minga2  ".167 SlantFont"  <b5minga2.pfb
..........
b5mingslf9  BIG5-Mingf9  ".167 SlantFont"  <b5mingf9.pfb
..........
```

This map file also goes into you local texmf tree:

`texmf.local/dvips/chinese/`

Don't forget to add the following line to your DVIPS configuration file `config.ps`:

`p+chinese.map`

水陸草木之花，可愛者甚蕃。晉陶淵明獨愛菊；自李唐來，世人盛愛牡丹；予獨愛蓮之出淤泥而不染，濯清漣而不妖，中通外直，不蔓不枝，香遠益清，亭亭靜植，可遠觀而不可褻玩焉。予謂菊，花之隱逸者也；牡丹，花之富貴者也；蓮，花之君子者也。噫！菊之愛，陶後鮮有聞；蓮之愛，同予者何人；牡丹之愛，宜乎眾矣。

Intermezzo 3 （宋）周敦頤：愛蓮說

Of course you will want to use those fonts in PDFTEX too, so add

`map +chinese.map`

to your PDFTEX configuation file `pdftex.cfg`. Now you can use these Chinese Type 1 fonts in TEX.

5 Using TrueType fonts

Although we can use Chinese TrueType fonts in traditional TEX by using `ttf2tfm` and `ttf2pk`, next we will concentrate on using `ttf` fonts in PDFTEX. Instead of using

> 水陆草木之花，可爱者甚蕃。晋陶渊明独爱菊；自李唐来，世人盛爱牡丹；予独爱莲之出淤泥而不染，濯清涟而不妖，中通外直，不蔓不枝，香远益清，亭亭静植，可远观而不可亵玩焉。予谓菊，花之隐逸者也；牡丹，花之富贵者也；莲，花之君子者也。噫！菊之爱，陶后鲜有闻；莲之爱，同予者何人；牡丹之爱，宜乎众矣。

Intermezzo 4 （宋） 周敦颐 ： 爱莲说

`ttf2pk` to generate bitmap fonts from the `ttf` outline fonts and live with ugly screen previews in PDF document, we will include the information from the `ttf` files directly. In order to do so, we first we need to generate the encoding index files. For simplified Chinese, use the following command:

```
gbenc -gbk htsong.ttf song
```

Here the option `-c` has the same meaning as in the `gbpfb` command. When `gbenc` returns the prompt, we will have 126 `tfm` files and 126 `enc` files, unless we provide the `-c` option. The main difference with creating Type 1 fonts using `gbpfb` is, that now we get `enc` files instead of `pfb` files. This saves many files and much disk space.

After dealing with the `ttf` fonts, you need to move the `tfm` and `enc` files:

```
texmf-local/fonts/tfm/chinese/
texmf-local/dvips/chinese/enc/
```

Again, we have to make a map file named `ttf-ch.map`, that contains the following lines:

```
gbsong81 GBSong81.enc <htsong.ttf
gbsong82 GBSong82.enc <htsong.ttf
........
gbsongfe GBSongfe.enc <htsong.ttf
........
b5minga1 B5Minga1.enc <htsong.ttf
b5minga2 B5Minga2.enc <htsong.ttf
........
b5mingf9 B5Mingf9.enc <htsong.ttf
........
```

This file goes into:

```
texmf.local/dvips/chinese/
```

To the PDFTEX configuration file `pdftex.cfg` you need to add the line:

```
map +ttf-ch.map
```

Now your PDFTEX can take advantage of the Chinese `ttf` fonts. Of course these font files must be somewhere in your font path.

君不见，黄河之水天上来，奔流到海不复回。君不见，高堂明镜悲白发，朝如青丝暮成雪。人生得意须尽欢，莫使金樽空对月。天生我材必有用，千金散尽还复来。烹羊宰牛且为乐，会须一饮三百杯。岑夫子，丹丘生，将进酒，君莫停。与君歌一曲，请君为我侧耳听。钟鼓馔玉不足贵，但愿长醉不愿醒。古来圣贤皆寂寞，惟有饮者留其名。陈王昔时宴平乐，斗酒十千恣欢谑。主人何为言少钱，径须沽取对君酌。五花马，千金裘，呼儿将出换美酒，与尔同销万古愁。

Intermezzo 5　李 白 ： 将进酒

6 Chinese in CONTEXT

Typesetting Chinese is complicated by the fact that, since the glyphs have a pictorial nature, label texts as well as numbers depend on the encoding and style (simplified or traditional) used. Another complication is that the default style is to be adapted to the (at least in scientific documents) used mixed Chinese and Western numbering scheme. CONTEXT tries to take care of this as natural as possible.

A Chinese document starts with (we assume you use the english interface):

```
\usemodule[chinese]
```

This zero module initializes the language and the fonts and sets up a basic style. After that, you have to choose the kind of Chinese. More convenient is to use one of:

```
\usemodule[chi-traditional]
\usemodule[chi-simplified]
```

After loading one of those modules, you can typeset Chinese in the main area between `\starttext` and `\stoptext`. For example,

```
\midaligned{\bfb 老子} \blank
```

道可道，非常道。名可名，非常名。无名天地之始；有名万物
之母。故常无，欲以观其妙；常有，欲以观其徼。此两者，同
出而异名，同谓之玄。玄之又玄，众妙之门。

will be produce following text:

<div align="center">

老子

</div>

道可道，非常道。名可名，非常名。无名天地之始；有名万物之母。故常无，欲以观其妙；常有，欲以观其徼。此两者，同出而异名，同谓之玄。玄之又玄，众妙之门。

Typesetting Chinese in traditional TEX is not trivial, since we cannot fall back on low level support for multi–byte characters and chinese paragraph breaking. Therefore, the process of glyph placement is coded in macros. You can visualize this process by enabling the trace option with `\tracechinesetrue`, as we did in the following example.

<div align="center">

老子

道可道，非常道。名可名，非常名。无名天地之始；有名万物之母。故常无，欲以观其妙；常有，
欲以观其徼。此两者，同出而异名，同谓之玄。玄之又玄，众妙之门。

</div>

When tracing is turned on, you will see the bounding boxes and some characters indicating how the glyph is treated. The next table is typeset with `\showchinesetracelegend`.

key	meaning	glyph	keys
u	unknown character type	current	u l r
l	left boundary character	previous	u l r c n
r	right boundary character	next	u l r s
c	chinese character		
s	following space		
n	no preceding space		

When we typeset this text verbatim, we get:

`\midaligned{\bfb 老子} \blank`

道可道，非常道。名可名，非常名。无名天地之始；有名万物
之母。故常无，欲以观其妙；常有，欲以观其徼。此两者，同
出而异名，同谓之玄。玄之又玄，众妙之门。

6.1 Chinese font definition

In the Chinese module, the SongTi (宋体) is chosen as the default normal typeface and HeiTi (黑体) as bold typeface, for both Simplified and Traditional Chinese. These fonts are defined as follows:

```
\defineunicodefont [SimChi] [SimplifiedChinese]  [chinese]

\definefontsynonym [SimplifiedChineseRegular]      [gbsong]   [encoding=gbk]
\definefontsynonym [SimplifiedChineseSlanted]      [gbsongsl] [encoding=gbk]
\definefontsynonym [SimplifiedChineseItalic]       [gbsongsl] [encoding=gbk]
\definefontsynonym [SimplifiedChineseBold]         [gbhei]    [encoding=gbk]
\definefontsynonym [SimplifiedChineseBoldSlanted]  [gbheisl]  [encoding=gbk]
\definefontsynonym [SimplifiedChineseBoldItalic]   [gbheisl]  [encoding=gbk]
```

In a similar fashion we define a Traditional Chinese set of fonts.

```
\defineunicodefont [TraChi] [TraditionalChinese] [chinese]

\definefontsynonym [TraditionalChineseRegular]      [b5ming]  [encoding=big5]
\definefontsynonym [TraditionalChineseSlanted]      [b5mingsl] [encoding=big5]
\definefontsynonym [TraditionalChineseItalic]       [b5mingsl] [encoding=big5]
\definefontsynonym [TraditionalChineseBold]         [b5hei]   [encoding=big5]
\definefontsynonym [TraditionalChineseBoldSlanted] [b5heisl]  [encoding=big5]
\definefontsynonym [TraditionalChineseBoldItalic]  [b5heisl]  [encoding=big5]
```

As you can see, defining you own Chinese fonts environments in CONTEXT is not that hard, given that you have the desired Chinese font available in a suitable format like Type 1 or TrueType. If for instance you want to use a LiShu font, the following commands define the command \LiShu as well as switching to the font, you can write the following definition in the appropriant bold, slanted and whatever alternatives, for as far as they make sense in Chinese typesetting.

```
\defineunicodefont [LiShu] [LiShu] [chinese]

\definefontsynonym [LiShuRegular]     [gbls] [encoding=gbk]
\definefontsynonym [LiShuSlanted]     [gbls] [encoding=gbk]
\definefontsynonym [LiShuItalic]      [gbls] [encoding=gbk]
\definefontsynonym [LiShuBold]        [gbls] [encoding=gbk]
\definefontsynonym [LiShuBoldSlanted] [gbls] [encoding=gbk]
\definefontsynonym [LiShuBoldItalic]  [gbls] [encoding=gbk]
```

Here all alternatives map onto the same font, but you may fill in other names. The previous definitions also assume that Simplified Chinese is used. After the following definitions \SimLiShu will switch to Simplified Chinese using LiShu fonts.

```
\defineunicodefont [SimLiShu] [SimChiLiShu] [chinese]

\definefontsynonym [SimChiLiShuRegular]     [gbls] [encoding=gbk]
\definefontsynonym [SimChiLiShuSlanted]     [gbls] [encoding=gbk]
\definefontsynonym [SimChiLiShuItalic]      [gbls] [encoding=gbk]
\definefontsynonym [SimChiLiShuBold]        [gbls] [encoding=gbk]
```

```
\definefontsynonym [SimChiLiShuBoldSlanted] [gbls] [encoding=gbk]
\definefontsynonym [SimChiLiShuBoldItalic]  [gbls] [encoding=gbk]
```

The next definitions set up the command `\TraLiShu` to typeset Traditional Chinese in LiChu fonts.

```
\defineunicodefont [TraLiShu] [TraChiLiShu] [chinese]

\definefontsynonym [TraChiLiShuRegular]     [gbls] [encoding=big5]
\definefontsynonym [TraChiLiShuSlanted]     [gbls] [encoding=big5]
\definefontsynonym [TraChiLiShuItalic]      [gbls] [encoding=big5]
\definefontsynonym [TraChiLiShuBold]        [gbls] [encoding=big5]
\definefontsynonym [TraChiLiShuBoldSlanted] [gbls] [encoding=big5]
\definefontsynonym [TraChiLiShuBoldItalic]  [gbls] [encoding=big5]
```

6.2 Chinese number

Although Western typesetting styles more and more influences Chinese typesetting, there remain some unique features in Chinese documents, especially in handling numbers, section labels, indexes, etc. And, if you typeset Traditional Chinese documents, you may want to use typeset the Chinese texts vertically.

In CONTEXT, Chinese numbers are fully supported. When you use the Chinese module, the chapter numbers will be in Chinese while other numbers, such as the number of a section, subsection, figure, page, etc. will be in Arabic numbers.

You can set yourself if and where numbers should be handled in Chinese. For instance, if you want the section numbers to be in Chinese too, you can force CONTEXT in doing as such by the command:

```
\setuphead [section] [conversion=chinese]
```

All commands that accept a conversion key can be set up to handle Chinese numbers. You can set back the numbering by either providing no conversion or map into numbers:

```
\setuphead [chapter] [conversion=]
\setuphead [section] [conversion=numbers]
```

Note, that when you use the Chinese numbering scheme in section, subsection, table, figure, and other headers, you should make sure that the number that prefixes the number, like chapter numbers, are turned off. Otherwise you may get a strange results,

like '第一.一节' or '图二.一'. In the base Chinese typesetting module `s-chi-00.tex` you can find some examples.

===================== ADD ==============

```
%% Hans, some number styles I talk about below need implement
%% In ConTeXt. The first and second style have supported.
%% The different between the first style and third style is
%% that the third one use one Chinese character instead
%% the two characters used in the first style if chinese number
%% between 20 - 39. For example, the chinese number 20 - 29 will
%% be the follows in the first style:
%%
%% \uchar{182}{254}\uchar{202}{174},
%% \uchar{182}{254}\uchar{202}{174}\uchar{210}{187}
%% ....
%% \uchar{182}{254}\uchar{202}{174}\uchar{190}{197}
%%
%% In the third style, the first two characters
%% \uchar{182}{254}\uchar{202}{174}
%% will be replaced with one character \uchar{216}{165}.
%% So does the number 30 - 39. using \uchar{216}{166}
%% instead of \uchar{200}{253}\uchar{202}{174}.
%%
%% The fourth style is like the Arabic style, all numbers
%% will be just represented by the digitials which they have.
%% for example, the number 10 will have two digitial, 1 and 0.
%% then fourth style Chinese number will be
%% \uchar{210}{187}\uchar{\uchar{193}{227}, that is
%% chinesenumber{1}chinesenumber{0}, two characters. another example,
%% number 129 will be chinesenumber{1}chinesenumber{2}chinesenumber{9},
%% three characters. This style Chinese number will be mostly used
%% in page number, year.
%%
%% In fact, there are other styles which combined the two of
%% the above styles. For example, when the number samller than
%% 100, use the first style and use the fourth style when the
%% number larger than 100. another example, when the number samller
%% than 40, use the third style, and when the number larger than 40,
%% use the fourth style. So complicated is the chinese number system!
```

```
%% I can not make very clear about it even if I am Chinese.
%% But I hope you can add more style support in ConTeXt.
%%
%% Some coomand and options I used both above and below
%% may not exist. You may find best solution to implement
%% the function I wanted.

\CONTEXT\ supports four Chinese numbering styles. The first
one is the formal style:
一、二、十三、三十六、一百二十五、一千零四十八. The second
style looks like the first one, but uses capitical Chinese
numbers. In the third style, the number 二十 and 三十 will
be replaced with 廿 and 卅 respectively. The fourth style
may be useful in page numbers or when mentioning years. This
method mimics Arabic numbers, by translating the Arabic
numbers into Chinese.

The first numbering style is the default ... setup ...

\subsection{Labels}

Chinese documents usually use split labels for chapter,
section, and similar headings. For example, they use \quote
{第1章} instead of the \quote{Chapter 1} in an English
document. In \CONTEXT, these split lables can be setup using
\type {\setuplabeltext}. The Chinese chapter label was
defined as:

\starttyping
\setuplabeltext [cn] [chapter={第,章}]
```

After you define the split label text, the `\in{chapter}[ref]` will give the split label too. Of course you can define the label of section, subsection, table, figure, etc. yourself. The Chinese module predefines a few labels.

6.3 Referencing and indexing

```
% About the Reference or Biolography, I found the Taco's
% bib module have something need improvement to sort
```

```
% Chinese bib entries which need add Chinese punctuation,
% symbols, quote, etc. When I prepare my paper, I can not
% let context auto determine the Chinese and non-Chinese
% so that use the different way to sort them. Finally I
% use the itemize instead.

% As for Index, I have talk about it with De-Wei Yin. I also
% attached the mail to you. To sort Chinese index is not
% a easy job. As I know, there are some software may be
% handle it but no free one. Does context allow user to edit
% and sort the index entries manually? or just only by texutil?
% write a perl script to sort Chinese Index refer to a pre-build
% index database may be not a good way because the database
% of Chinese word is much big. I will thinking about it and
% ask someone who major in Chinese. May be find a better way.
```

6.4 Vertical typesetting

In Taiwan and HongKong, a large deal of Chinese is typeset vertical. CONTEXT support this mode by implementing on top of the multi–column routines.

6.5 Tips and tricks

- how to encode the document: emacs, etc

- how to code tex files to suit the encoding

- running context: `%convert=...` like the xml one: uc2tex

7 More information

This manual is written by Wang Lei and Hans Hagen. More information on CONTEXT can be found at the www.pragma-ade.nl or www.ntg.nl. Additional information on processing Chinese in CONTEXT can be found there.