Linux Foundation Training
Guide to Translation of Courses
Version 6.0

Jerry Cooperstein

04/05/2021

Note: This document is internal to the **Linux Foundation** and those working with us on courseware. It is not to be distributed outside that arena, and is considered private.

This documentation is not meant to be fully comprehensive, or to teach the use of either **git** or LaTeX. It is really more of an FAQ.

Contents

1 Introduction

Growing demand for translation of **Linux Foundation** courses into other languages has necessitated articulating proper methods, drawing on previous experience in doing so, and producing this documentation.

To date, translations of some material have been successfully done in Japanese, Spanish, Portuguese, Chinese, and Korean.

To make this process easier, we have modularized the methods as much as possible and restricted the number of places that require modification. In practice this means the metafiles needed for production are already language-aware and need be added to only once per language no matter how many courses we translate.

Further Documentation

Besides this document, we provide a more detailed instruction manual, the **Linux Foundation Style Manual and Preparation Guide for Developer and Enterprise System Administration Courses**, which you should have obtained with this document. Please keep in mind that long document is intended for **course authors** and contain many details a translator has no need to know, but reading it will give a good background that can only help.

There is also a short *Cheatsheet*, the **Linux Foundation LaTeX Formatting and Style CheatSheet**, which you should also have received.

2 Git and Communication with the Linux Foundation Course Developers

All **Linux Foundation** course material is kept in **Git private** repositories at github.com.

Once things are mature, we give access to a (limited) number of translating collaborators and open up a new branch (such as Spanish, French, Chinese etc.) within the repo.

However, since we do not "merge" the foreign language translations, but keep them in parallel concurrent **branches** this is not necessarily a particularly good way to do things. When beginning, we will open up a separate repository for the specific course and the language, based on a clone of the English version.

Besides email, we will likely maintain a Google drive area we have mutual access to. This area should also contain any other supplementary files such as figures, change logs etc.

Keeping up to date

All **Linux Foundation** classes are under constant revision. This is because of keeping up with upstream versions, as well as fixing weaknesses and including more or enhanced material.

When a course has been translated this adds additional burdens in that the main course maintainer has to become very careful about keeping track of changes so they can be incorporated in the next translation cycle. Since translation takes time, it is often the case that translation updates are less frequent and thus not as up-to-date than their English language counterparts.

3 Types of Courses

It is important to know that in terms of **delivery methods**, there are two distinct types of **Linux Foundation** courses; this impacts how translations may be done. These are:

1. **Instructor-led**

   These may be delivered either in a physical classroom or over the Internet. In almost all cases content is the same in either format (The **Linux Foundation** does offer classroom-only courses, but these are developer courses which require hardware access. However, the course manuals are of the usual format.)

2. **Self-Paced On-Line E-learning**

   These do not have human instructors, and getting help requires posting to forums, which are populated by other students as well as **Linux Foundation** moderators.

There are differences here in both content coverage and in production methods. For example, self-paced courses need to be more verbose as a real-time teacher is not available.

Instructor-led courses have two essential documents; a slide deck for the instructor to use, and a book for students to receive. Both are released as PDF files.

For the most part, self-paced courses are produced using an elearning content preparation system, and not in LaTeX, except for the exceptions (labs) discussed here.

Very Important

There is a further bifurcation; self-paced courses that have an instructor-led sibling (e.g., **LFS258** and **LFS458**) and those which do not (e.g., **LFS253**).

Those which have a sibling relationship have essentially the same content, but the lecture part of the content is formatted very differently. However, the lab exercises are still produced in LaTeX and differ only in the course number identifiers and version numbers in the footers of each page.

Thus:

- If there is no sibling relationship **no translation** needs to done in LaTeX for the e-learning course. You can stop reading this document!
- If there is a sibling relationship, **only the labs** need to be translated in LaTeX.
- If there needs to be produced an instructor-led version, **everything** needs to be translated into LaTeX.

We will not discuss the translation process here for the non-LaTeX parts as they are documented elsewhere by the **Linux Foundation** instructional design team.

Furthermore, the **Linux Foundation** produces two different kinds of classes:

1. **LFS Courses (Linux Enterprise System Administration Training)**
2. **LFD Courses (Linux Developer Training)**

To date we have only embarked on translation of **LFS** courses so we will concentrate on that here.

It is important to note that the actual format of these two course categories is distinct; this is a result of audience variation as well as somewhat accidental parallel evolution within the **Linux Foundation** training department.

Until recently, **LFS** and **LFD** courses used different document preparation systems; even though both used LaTeX, the sources looked quite different. Now they have been unified into one system, although end product looks quite different.

LFS classes adopt a format where there is a slide deck which the instructor uses to present (which is a pdf file) and a book file which contains every slide on the top half of the page, and on the bottom half of the page there are additional notes which do not appear in the slides deck. Furthermore, there are laboratory exercises at the end of each chapter that appear in each pdf file.

4 LaTeX

Linux Foundation instructor-led courseware is all produced using LaTeX . At the source level, self-paced courses use the same underlying materials, although in the end LaTeX is only used to prepare laboratory exercises and solutions which are available as downloads for students.

Note that one important reason to prefer using LaTeX for translation is that the courseware will retain the look and feel and branding for the **Linux Foundation**, including covers, logos, page styles, tables of contents and figures and tables, etc. This is important for branding issues.

The preparation document has an extended explanation (perhaps defense) of the reasons for using LaTeX. These reasons include:

- It is a non-proprietary system, in fact one of the oldest open source projects dating back to the 1970s at least.
- It is essentially bug-free and has a very active contributing community and is very well maintained.
- It is universally available on all operating systems and distributions.
- It produces highly sophisticated output, particularly with equations and other difficult to format content; it is actually a typesetting system, not a word processor or slide preparation system.

However, for people used to working in a WYSIWYG type system such as an office suite, it can be intimidating. But once you learn the basics and develop some good ergonomic habits it is extremely efficient to use and extremely fast at producing even very long and complex documents.

LaTeX has been ported to virtually every language on the planet, and most **Linux** distributions have the appropriate language-specific packages available as a simple installation within the normal package management system. However, each language may have its own specific quirks that may require some additional investigation, especially for non-western character sets. This was true for Korean for example, but it was not hard with a little googling to figure out what was needed.

LaTeX Installation

The longer document gives some information about how to install LaTeX on any **Linux** system. We will be happy to assist in details and problems which you encounter. On most distributions simply installing all related packages is enough.

5 What needs to be translated

First we will explain what needs to be translated **within LaTeX** for a complete translation for instructor-led courses. If it is self-paced, there are some other options; we will come back to that.

Generally note that LaTeX is a markup language and one does not have to translate LaTeX commands. Basically that means anything that starts with a backslash \, such as `\newcommand{}` or `\item` etc., or beginning and ending environments such as `\begin{itemize}` and `\end{itemize}`.

Also anything in a **verbatim** environment (generally script and code, commands and output) should not be translated.

In the following we will assume the course is **LFS301**. The source is all under the directory `LFS301`.

5.0.1 Main Directory

In `LFS301` one needs to do some translation in only the following file:

- `LFS301.tex`:

```
... \coursetitle{Linux System Administration} ....
```

That line is the only one that requires direct translation, but there are some other lines that should be added to the file before the `\documentclass{}` statement. For example, in the Spanish version we have:

```
\usepackage[utf8x]{inputenc}
\renewcommand{\contentsname}{Contenidos}
\renewcommand{\chaptername}{Cap\'itulo}
```

The first line gives the correct character encoding set, the second replaces the word "Contents" and the third replaces the word "Chapter". There may be other names that need translation, such as `\listfigurename{}`, `\listtablename{}`, `\figurename{}`, `\tablename{}`, etc. If there is anything missing in the final document I can help figure it out. (You will see this in the form of English words you would rather have in your language.)

5.0.2 Style files in common/texmf/tex/latex/LFCW

Only needs to be done once per language!

If there has already been a translation into your language this has already been done and nothing in the texmf directory tree needs to be touched again. You can skip this section except to learn.

One also has to customize the LaTeX class files which are found in LFS301/common/texmf/tex/latex/LFCW. It is only a few things which require translation here. Anything you miss will be pretty obvious in the output. Particulars:

- longcopyright.tex

  This one is pretty obvious. It is the long copyright that appears after the title page in the manual, and the whole thing requires translation.

- LFcover.cls

  The Back Cover text which starts with:

```
....
        {
           \textbf{\color{LFDeepBlue}Train with
              the Linux Experts}\par\medskip The
           Linux Foundation has a mandate to
           train the next generation of Linux
           .....
         }
```

- LFD.cls

  This is the main class (style) file for the courseware.

```
\SetWatermarkText{Confidential Copy: Do Not Distribute}

......

\textcopyright\  Copyright the Linux Foundation 2020.  All rights reserved.}

......

      \course: Version \version

.....

\newcommand{\optionalfootnotetext}{
```

```
  \textbf{**} These sections may be considered in part or
  in whole as optional.  They contain either background reference
  material, specialized topics, or advanced subjects. The
  instructor may choose to cover or not cover them depending on
  classroom experience and time constraints.
}

.....

   \section*{Lab \thechapter}

.....

   \normalsize{\textbf{Solution \theex}}}

.....
```

5.0.3 Main Course Content

The text for the course sections is contained under `LFS301/CHAPS` with each section having its own directory. For example:

```
c8:/teaching/LFCW/LFS301/CHAPS>ls -l xfs_btrfs/
total 32
-rw-rw-r--  1 coop coop 2440 Nov  7 09:02 btrfs.tex
-rw-rw-r--  1 coop coop  136 Nov  7 09:02 index.tex
-rw-rw-r--  1 coop coop   91 Nov  7 09:02 labs.tex
-rw-rw-r--  1 coop coop 2531 Nov  7 09:02 xfs_btrfs1.tex
-rw-rw-r--  1 coop coop 2555 Nov  7 09:02 xfs_btrfs2.tex
-rw-rw-r--  1 coop coop 3558 Mar 25 18:10 xfs.tex
```

Each of these files will have to be translated. This is the main body of work.

Very Important

Remember to translate the arguments of `\chapter{}`, `\section{}` and `\subsection{}` statements.

Sometimes only the lab exercises are needed

In some cases it is much easier; only the exercises need to be translated. This means only `labs.tex` and any files included from therein, as well as any titles that are in `index.tex`.

Be careful of files which are included as raw text (usually anything with a `.inc` extension) such as:

```
$ cat LFS301/CHAPS>cat kernel/kern.inc

kernel /vmlinuz-2.6.32-279.14.1.el6.x86_64 ro root=UUID=<UUIDVALUE>
rd_NO_LUKS rd_NO_LVM LANG=en_US.UTF-8 rd_NO_MD
SYSFONT=latarcyrheb-sun16 crashkernel=auto  KEYBOARDTYPE=pc
KEYTABLE=us rd_NO_DM rhgb quiet
```

These will generally be included **verbatim** and do not need to be translated.

Note that in **LFS301** (an uncommon case) there are inclusions in `labs.tex`:

```
$ cat LFS301/CHAPS/xfs_btrfs/labs.tex
```

```
\clearpage\section{Labs}\begin{Lab}
   \input{xfs_btrfs1}
   \input{xfs_btrfs2}
\end{Lab}
```

Producing lab files for the e-learning LMS

As noted earlier, self-paced classes have been done in two different ways:

1. Do everything in LaTeX and the produce PDF's which will then be imported into the e-learning content management system.
2. Translate directly into the LMS everything except the labs, which are still produced with LaTeX by doing

   ```
   $ make labs
   $ make splitlabs
   ```

   which produces each lab as a separate file, and produces also one long file with all labs; these are directly inserted into the LMS.

If one were not to produce labs in LaTeX one would have to find a way to produce PDF's with the right formatting. (All labs are provided as downloaded PDF's, not as direct input in **Lectora**.)