Future Format of Data Safety Guidance

# Introduction

As we discussed in our last meeting, I’ve had requests for the data safety guidance document to be available as ePub (or similar eReader format) and as a HTML document.

I think ePub is only useful if it’s flowable. If the format is fixed then it offers no significant advantage over PDF. Any route to that is likely to need manual fixing up of tables and so on.

HTML is tricky because for it to be useful *as HTML* it should be non-linear, unlike the existing book. It might be possible to satisfy Divya’s wish to be able to link to points in the book by setting destinations in the PDF rather than moving to HTML.

I’ve looked at the suggested ways suggested to manage the guidance document to support multiple output formats.

# LaTeX

The current format for the document. We have solid experience with it, and have configured it to produce good-quality documents. It’s free, and accurately reflects the input (although that could be because we are reasonably adept at tweaking it). However, producing ePub and HTML is likely to be laborious. It is also not very robust, because we depend on many independent packages that can be incompatible – a lot of the effort of producing version 3.7 of the guidance was fixing up the document because an update to one package caused another, seemingly independent, package to give incorrect output.

# Pandoc/Calibre

These are document conversion tools. Pandoc will take LaTeX directly, but could not process the guidance because it does not support some of the packages we use. Converting from PDF gave superficially *reasonable* (not good) results, but because PDF (and DVI) lose much of the document’s structural information, the resulting HTML has serious deficiencies such as every *line* of text being marked up as a new paragraph, whether it’s a new paragraph or not (so no easy way to automate fixing up the resulting document)

# Dita

Dita imposes a structure on the text based strongly around steps to accomplish tasks. This would need a complete restructuring and rewrite of the guidance document around tasks. That might arguably lead to a more useful document, but it would be hard to incorporate much of the existing content in a Dita structure, and it would be so different from the existing one that it couldn’t really be regarded as just an update; it would be a new document. It doesn’t seem to support multiple glossaries, so we’d need to find a different way to distinguish normative and informative glossary entries.

# DocBook

DocBook is a mature and comprehensive XML structure, and as such it could certainly *define* the content we want. But turning it into an actual document will depend on the renderer we use. Most of them use XSLT to perform the transform, so for ePub and HTML we can probably tailor existing XSLT to give the appearance we want, as we have done with the LaTeX for the existing document. Tailoring PDF could be more difficult. Supports glossaries and abbreviation lists, but no automation so they would have to be generated manually (though because it’s XML, it shouldn’t be difficult to write scripts in a suitable scripting language to automate that).

# Affinity

Affinity is Serif’s DTP package, and seems to be highly regarded at the budget end of the DTP market. Because it’s a DTP package it should be able to produce a highly presentable book. It doesn’t seem to have any support for automatic generation of glossaries or abbreviation lists, so that would all have to be done manually. It doesn’t provide any straightforward way to produce ePub or HTML, though they’re long-requested features, so it offers no significant advantage over the LaTeX we’re currently using unless we want to make substantial changes to the presentation.

# InDesign

Adobe’s DTP, which seems to be the industry standard for book production. Unquestionably powerful, but unfortunately expensive. Still has limited HTML capability, probably because HTML structure is so different to the linear structure of a book. Also very limited automation of glossaries and acronym lists.

# Scribus

An open-source DTP package that seems well-regarded. As a DTP package it should be able to give good results. Claims to support ePub and HTML as well as PDF. Unfortunately, I’ve not been able to make much progress with it because the documentation has fallen behind the current version of the software and I have had difficulty registering on the support forum (not unusual issues with open source software, and comparable to the problems I’ve had with LaTeX packages).

# Word

A word processor rather than DTP. Doesn’t have automation for acronym lists or glossaries. Moderately expensive, but most of us probably have it anyway (free alternatives such as LibreOffice Writer might be contenders too). For presentation, we might be working somewhat against its word processing nature, but there’s a well-established path from Word to ePub. There is the same issue that HTML needs a very different structure.

# Summary

The most professional-looking results would probably come from InDesign, but unless the DSIWG has the budget for a couple of InDesign seats (and considers this a worthwhile use of funds) that’s not feasible. Scribus and Docbook might be worth further investigation, but otherwise there does not seem to be a feasible route to PDF, HTML and ePub from a common source text.