Writing I-Ds and RFCs using Pandoc and a bit of XML
draft-gieben-pandoc2rfc-02

Abstract

   This document presents a technique for using a Markdown syntax
   variant, called Pandoc, and a (bit) of XML [RFC2629] as a source
   format for documents in the Internet-Drafts (I-Ds) and Request for
   Comments (RFC) series.

   The goal of this technique is to let an author of an I-D focus on the
   main body of text without being distracted too much by XML tags, it
   does however not alleviate the need to typeset some files in XML.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on January 06, 2014.

Copyright Notice

Table of Contents

1. Introduction

This document presents a technique for using a Markdown [Markdown]
syntax variant, called Pandoci [Pandoc], and a bit of XML [RFC2629]
as a source format for documents in the Internet-Drafts (I-Ds) and
Request for Comments (RFC) series.

The goal of this technique is to let an author of an I-D focus on the
main body of text without being distracted too much by XML tags, it
does however not alleviate the need to typeset some files in XML.

Pandoc is an almost plain text format and therefor particularly well
suited for editing RFC-like documents. The syntax itself is a super
set of the syntax championed by Markdown.

2. Pandoc to RFC

Pandoc's syntax is easy to learn and write and it can be translated
to numerous output formats, including, but not limited to: HTML,
EPUB, (plain) Markdown and DocBook XML.

Pandoc2rfc allows authors to write in Pandoc syntax which is then
transformed to XML and given to xml2rfc. The conversions are, in a
way amusing, as we start off with (almost) plain text, use elaborate
XML and end up with plain text again.

```
         +-------------------+   pandoc   +---------+
         | ALMOST PLAIN TEXT |   ------>   | DOCBOOK |
         +-------------------+      1      +---------+
                   |                            |
         non-existent |                    2 | xsltproc
           faster way |                        |
                   v                            v
            +------------+   xml2rfc   +---------+
            | PLAIN TEXT |   <--------  |   XML   |
            +------------+      3      +---------+
```
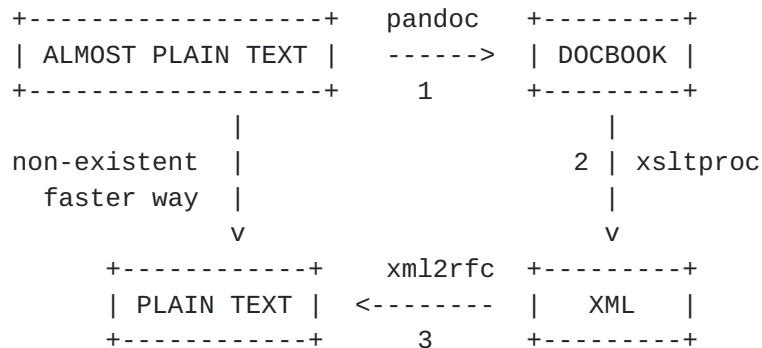
                  Figure 1: Attempt to justify Pandoc2rfc.

The output of step 2 in Figure 1 is XML which is suitable for
inclusion in either the "middle" or "back" section of an RFC.

Even though Pandoc2rfc abstracts away a lot of XML details, there are
still places left where XML files needs to be edited. Most notably
in the "front" section of an RFC.

The simplest way to start using Pandoc2rfc is to create a template
XML file and include the appropriate XML for the "front", "middle"
and "back" section:

```
<?xml version='1.0' ?>
<!DOCTYPE rfc SYSTEM 'rfc2629.dtd' [
<!ENTITY pandocAbstract PUBLIC '' 'abstract.xml'>
<!ENTITY pandocMiddle   PUBLIC '' 'middle.xml'>
<!ENTITY pandocBack     PUBLIC '' 'back.xml'>
<!ENTITY rfc.2629       PUBLIC '' 'reference.RFC.2629.xml'>
]>

<rfc ipr='trust200902' docName='draft-gieben-pandoc2rfc-01'>
 <front>
    <title>Writing I-Ds and RFCs using Pandoc</title>
    <author>
       <organization/>
       <address><uri>http://www.example.com</uri></address>
    </author>
    <date/>
    <abstract>
        &pandocAbstract;
    </abstract>
```

```
    </front>
    <middle>
        &pandocMiddle;
    </middle>
    <back>
        <references title="Normative References">
            &rfc.2629;
        </references>
        &pandocBack;
    </back>
    </rfc>
```

Figure 2: A minimal template.xml.

In this case you will need to edit four documents:

1. "abstract.mkd" - contains the abstract;

2. "middle.mkd" - contains the main body of text;

3. "back.mkd" - holds the appendices (if any);

4. And this "template.xml" - probably a fairly static file, among
    other things, it holds the author(s) and the references.

Up to date source code for Pandoc2rfc can be found at [Pandoc2rfc],
this includes the style sheet "transform.xsl" which used for the XML
transformation (also see Section 3).

2.1. Dependencies

Pandoc2rfc needs "xsltproc" [XSLT] and "pandoc" [Pandoc] to be
installed. The conversion to xml2rfc XML is done with a style sheet
based on XSLT version 1.0 [W3C.REC-xslt-19991116].

When using the template from Figure 2 xml2rfc version 2 (or higher)
must be used.

3. Building an Internet-Draft

Assuming the setup from Section 2, we can build an I-D as follows (in
a Unix-like environment):

```
for i in abstract middle back; do
pandoc -st docbook $i.mkd | xsltproc --nonet transform.xsl - > $i.xml
done

xml2rfc template.xml -f draft.txt --text  # create text output
```

```
xml2rfc template.xml -f draft.html --html # or create HTML output
xml2rfc template.xml -f draft.xml --exp   # or create XML output
```

                    Figure 3: Building an I-D.

Note that the output file names (abstract.xml, middle.xml and
back.xml) must match the names used as the XML entities in
"template.xml" (See the "!ENTITY" lines in Figure 2). The Pandoc2rfc
source repository includes a shell script that incorporates the above
transformations. Creating a "draft.txt" or a "draft.xml" can be done
with "pandoc2rfc *.mkd" and "pandoc2rfc -X *.mkd" respectively.

4. Supported Features

The full description of Pandoc's syntax can be found in
[PandocGuide]. The following features of xml2rfc are supported by
Pandoc2rfc (also see Table 1 in Appendix B for a "cheat sheet"):

o Sections with an anchor and title attributes;

o Several lists styles:

   * style="symbols", use "* " for each item;

   * style="numbers", use digits: "1. " for each item;

   * style="empty", use "#. " for each item;

   * style="format %i", use roman lowercase numerals: "ii. ";

   * style="format (%d)", use roman uppercase numerals "II. ";

   * style="letters", use lower- or uppercase letters: "a. " and
      "A. " (note: two spaces as mandated by Pandoc);

   * style="hanging", use the Pandoc definition list syntax:

                        Term 1

                        :   Definition 1

        Figure 4: Pandoc syntax used for a hanging paragraph.

o Spanx style="verb", style="emph" and style="strong", respectively
   use: "`text`", "_text_" or "**text**";

o Block quote is converted to "<list style="empty">" paragraph;

   o Figures with an anchor and title (Section 6.1);

   o Tables with an anchor and title (Section 6.2);

   o References (Section 6.3)

      * external ("<eref>");

      * internal ("<xref>"), to:

         + sections (handled by Pandoc);

         + figures (handled by XSLT);

         + tables (handled by XSLT).

   o Citations, by using internal references;

   o Processing Instructions ("PI"s: "<?rfc?>"), may be used after a
      section header, they are carried over to the generated XML;

   o The "<vspace>"-tag is supported and carried over to the generated
      XML.

5. Unsupported Features and Limitations

   With Pandoc2rfc an author of an I-D can get a long way without
   needing to input XML, but it is not a 100% solution. The initial
   setup and the reference library still forces the author to edit XML
   files. The meta data feature (Pandoc's "Title Block" extension) is
   not used in Pandoc2rfc. This information (authors, date, keyword and
   URLs) should be put in the "template.xml".

   Some other quirks:

   o An index is not supported;

   o Comments are supported via HTML comments in the Pandoc source
      files;

   o Authors still need to know how to deal with possible errors from
      xml2rfc.

6. Pandoc Style

   The following sections detail how to use the Pandoc syntax for
   figures, tables and references to get the desired output.

6.1. Figures

   Indent the paragraph with 4 spaces as mandated by Pandoc. If you add
   an inline footnote _directly_ after the figure, the artwork gets a
   title attribute with the text of that footnote (and a possible
   anchor).

6.2. Tables

   A table can be entered by using Pandoc's table syntax. You can
   choose multiple styles as input, but they all are converted to the
   same style (plain "<texttable>") table in xml2rfc. If you add an
   inline footnote _directly_ after the table, it will get a title
   attribute with the text of that footnote (and a possible anchor).
   The built in syntax of Pandoc to create a caption with "Table:"
   should not be used.

6.3. References

   Pandoc provides a syntax that can be used for references. Its syntax
   is repeated in this paragraph. Any reference like: "[Click
   here](URI)", is an external reference. An internal (i.e. see
   Section X) reference is typeset with: "[](#localid)".

   For referencing RFCs (and other citations), you will need add the
   reference source in the template, as an external XML entity, Figure 2
   provides an example. After that you can use an internal reference:
   "[](#RFC2629)" to reference RFC 2629.

   There is no direct support for referencing tables, figures and
   artworks, but pandoc2rfc employs the following "hack". If an inline
   footnote is added after the figure or table, the text of the footnote
   is used as the title. The first word up until a double colon "::"
   will be used as the anchor. If a figure has an anchor it will be
   centered on the page.

   Figure 2 for instance, is followed by this inline footnote:

   ^[fig:minimal::A minimal template.xml.]

7. Acknowledgements

The following people have helped shape Pandoc2rfc: Benno Overeinder,
Erlend Hamnaberg, Matthijs Mekking and Trygve Laugstoel.

8. Security Considerations

This document raises no security issues.

9. IANA Considerations

This document has no actions for IANA.

10. Normative References

[Markdown]
          Gruber, J., "Markdown", 2004,
          <http://daringfireball.net/projects/markdown/>.

[Pandoc2rfc]
          Gieben, R., "Pandoc2rfc git repository", October 2012,
          <http://github.com/miekg/pandoc2rfc>.

[PandocGuide]
          MacFarlane, J., "Pandoc User's Guide", 2006,
          <http://johnmacfarlane.net/pandoc/README.html>.

[Pandoc]   MacFarlane, J., "Pandoc, a universal document converter",
          2006, <http://johnmacfarlane.net/pandoc/>.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629,
          June 1999.

[W3C.REC-xslt-19991116]
          Clark, J., "XSL Transformations (XSLT) Version 1.0", World
          Wide Web Consortium Recommendation REC-xslt-19991116,
          November 1999,
          <http://www.w3.org/TR/1999/REC-xslt-19991116>.

[XSLT]   Veillard, D., "The XSLT C library for GNOME", 2006,
          <http://xmlsoft.org/XSLT/xsltproc2.html>.

Appendix A. Changelog

[This section should be removed by the RFC editor before publishing]

A.1. -00

1. Initial document.

A.2. -01

   1. Lots of updates;

   2. Added the style sheet use in an appendix.

A.3. -02

   1. Make "template.xml" actually valid XML;

   2. Removed the style sheet from the appendix;

   3. Make more explicit that typesetting some XML files is still
       needed;

   4. Fix blockquote text and conversion;

   5. Overhauled the way references to figures and tables work;

   6. Cleaned up and removed duplicate text.

Appendix B. Cheat Sheet

```
   +---------------------+----------------+--------------+
   | Textual construct   | Pandoc syntax  | Text output  |
   +---------------------+----------------+--------------+
   | Section Header      | "# Section"    | 1. Section   |
   | Unordered List      | "* item"       | o  item      |
   | Unordered List      | "#. item"      | item         |
   | Ordered List        | "1. item"      | 1.  item     |
   | Ordered List        | "a.  item"     | a.  item     |
   | Ordered List        | "ii. item"     | i.  item     |
   | Ordered List        | "II. item"     | (1)  item    |
   | Ordered List        | "A.  item"     | A.  item     |
   | Emphasis            | "_text_"       | _text_       |
   | Strong Emphasis     | "**text**"     | *text*       |
   | Verbatim            | "`text`"       | "text"       |
   | Block Quote         | "> quote"      | quote        |
   | External Reference  | "[Click](URI)" | Click [1]    |
   | Internal Reference  | "[](#id)"      | Section 1    |
   | Figure Anchor       | "^[fid::text]" | N/A          |
   | Figure Reference    | "[](#fid)"     | Figure 1     |
   | Table Anchor        | "^[tid::text]" | N/A          |
   | Table Reference     | "[](#tid)"     | Table 1      |
   | Citations           | "[](#RFC2119)" | [RFC2199]    |
   | Table               | Tables         |              |
   | Figures             | Code Blocks    |              |
   | Definition List     | Definition     |              |
```

```
   +--------------------+----------------+-------------+
```

Table 1: The most important textual constructs that can be used in Pandoc2rfc. The bottom three create output to voluminous to show in this table.

Author's Address

   R. (Miek) Gieben
   Google

   Email: miek@google.com