Overview

The RFC Editor plans to produce a canonical RFC document in XML using the xml2rfc v3 grammar, and to publish the RFC in several Publication Formats as defined in RFC6949.

This project will result in one or more applications to produce the Text, HTML, and PDF publication formats from an xml2rfc v3 source document. It will also produce a mechanism to create an xml2rfc v3 document from an xml2rfc v2 document, and a mechanism to conditioning an xml2rfc v3 document entry into the RFC archives.

Deliverables/Tasks

- Design the command line interface(s) and API(s)
- Develop the conversion application(s)
- Demonstrate the conversion of a specified set of test document in a standalone environment
- Demonstrate the conversion of a specified set of test documents using a web service
- Provide an extensible test suite for the application(s)
- Provide documentation, and training for the RFC Production Center staff

Detailed Description and Requirements

Each application must run standalone in Linux, OS X, and Windows environments. The application will also be made available as part of a web-service offering the translation, similar to what is available now at http://xml2rfc.tools.ietf.org/. The application will also be invoked, preferably directly through a software API, by the draft submission tool in the IETF datatracker.

The developer will work with the Program Manager to agree on how each application is invoked before beginning development work.

The project will use the following document(s) as the primary development and test targets:

<http://hildjj.github.io/draft-iab-html-rfc/example/test.3.xml>

<http://trac.tools.ietf.org/tools/xml2rfc/trac/browser/trunk/cli/tests/input/draft -miek-test.xml>

<http://trac.tools.ietf.org/tools/xml2rfc/trac/browser/trunk/cli/tests/input/draft -miek-test-v3.xml>

<http://datatracker.ietf.org/doc/draft-hoffman-rfcexamples/>

The application suite will accept input using either the xml2rfc v3 grammar (as described in draft-iab-xml2rfc or its successor) or in the xml2rfc v2 grammar (as described in RFC 7749). When processing an xml2rfc v2 document, the suite will produce a version converted into the xml2rfc v3 grammar. We expect the implementation of that conversion to be informed by the substantially complete grammar converter developed by the volunteer community at
<https://durif.tools.ietf.org/~tonyh/convertv2v3/convertv2v3.txt>).

The application(s) will produce output as specified in the following documents:

Format	Specifications
Text	draft-iab-rfc-plaintext
	RFC 7322 (section 4)
HTML	draft-iab-html-rfc
PDF	draft-iab-rfc-use-of-pdf

It is acceptable (as xml2rfc does now) to produce variations on those output formats, and other output formats, controlled by the application invocation or instructions in the document source.

There is a hand-crafted example of intended html output at http://hildjj.github.io/draft-iab-html-rfc/example/test.3.html. Where it conflicts with the definitions in draft-iab-html-rfc, draft-iab-html-rfc takes precedence.

The application(s) will correctly process any provided SVG (as defined by draft-iab-svg-rfc, according to the each output specification.

When producing an xml2rfc v3 output document the application will translate any deprecated elements in the v3 vocabulary using the constructs described in draft-hoffman-xml2rfc. This processing will occur both when the input document is v2 and when it is v3.

When creating an xml2rfc v3 document conditioned for entry into the RFC archives, the application will perform the additional processing described in draft-iab-rfcv3-preptool. It is anticipated that the draft submission tool will also perform these steps as part of accepting a draft for submission.

When presented with an input document that contains errors, the diagnostic output of the application(s) must help an author or the RFC Editor quickly locate and correct the source of each error. When possible, the application(s) should suggest corrective actions.

The requirements on the formats defined above are expected to change over time. The application must be easy to modify and maintain. An application written in a language already familiar to most of the volunteer support community is preferred.

Expected Development Processes and Practices

The contractor will adhere to the requirements at
http://trac.tools.ietf.org/tools/ietfdb/wiki/ContractorInstructions?version=23