

XML files

How to make raw files interoperable for SDR

Clemens Schmidt, Jan van Katwijk

April 2020

Table of Contents

1	INTRODUCTION.....	3
2	REQUIREMENTS.....	4
3	EXAMPLE.....	5
4	DETAILED DESCRIPTION.....	6
4.1	The Element tagged <i>Sample</i>	6
4.2	The Element tagged <i>Datablocks</i>	6
4.3	The Element tagged <i>Datablock</i>	7
5	OPTIONAL SUB ELEMENTS <i>RECORDER, DEVICE, TIME</i>	8
6	SUPPORTED DEVICES.....	9

1 INTRODUCTION

When working with software for Software Defined Radio, we often work with pre-recorded files. Pre-recorded files create possibilities of exchanging recordings, and - important when developing - they provide a benchmark for testing changes in the program(s).

Working on QIRX (Clemens Schmidt) and Qt-DAB (Jan van Katwijk), we noted that there was no real opportunity to exchange recordings from different devices, as devices tend to differ in the format of data their handlers deliver.

As an example, while some SDR devices provide 12 (14) bit integer values, the well known RT2832 based DABsticks provide 8 bit samples, as e.g. the Hackrf device does. However, the encoding of the 8 bit samples from the DABsticks is different from that of the Hackrf.

In order to exchange recordings of data from these - and other - devices, we needed a format to store and describe the data that allows us to process raw data from different devices, using a single interpreter.

The XML file format aims at providing a single format for the exchange of raw data, generated by different devices. The idea is simple: such an xml-file is a file containing in its first X bytes a description - UTF8 encoded, with the raw data following. Following state-of-the-art practice, the description will be written in xml. ¹

In section 2 we specify the requirements, in section 3 we give an example description and in section 4 we give a detailed description of the elements.

¹ Qt-Dab files containing the described format have the .uff extension, QIRX files have their usual .raw extension.

2 REQUIREMENTS

What we want is a format that allows the description of the contents of any foreseeable raw file, in a way that a single interpreting input handler in our software is able to process it. Therefore, the format should provide the possibility of specifying:

- An optional proper identification of the creating environment;
- The type and structure of samples to such a detail that an interpreter is able to rebuild the samples and interpret them properly;
- The amount of samples that can be processed as well as - optional - information on the recording purpose.

We have chosen UTF-8 encoded xml as the format.

3 EXAMPLE

```
<?xml version="1.0" encoding="utf-8"?>
<SDR>
  <Recorder Name="Qt-DAB" Version="3.2-Beta"/>
  <Device Name="AIRspy" Model="I"/>
  <Time Value="Wed Dec 18 12:39:34 2019" Unit="UTC"/>
  <!--The Sample information holds for the whole recording-->
  <Sample>
    <Samplerate Value="2500000" Unit="Hz"/>
    <Channels Bits="12" Container="int16" Ordering="LSB">
      <Channel Value="I"/>
      <Channel Value="Q"/>
    </Channels>
  </Sample>
  <!--Here follow one or more data blocks-->
  <Datablocks>
    <Datablock Number="1" Count="375783424" Unit="Channel">
      <Frequency Value="227360" Unit="KHz"/>
      <Modulation Value="DAB"/>
    </Datablock>
  </Datablocks>
</SDR>
```

4 DETAILED DESCRIPTION

A valid XML file is tagged with *SDR* as its root element. The valid XML file contains at least sub elements tagged with *Sample* and *Datablocks*. A valid XML file may further contain sub elements from the root tagged with *Recorder*, *Device*, and *Time*.

4.1 The Element tagged *Sample*

The element tagged *Sample* describes the sample in detail. It contains two sub elements, tagged *Samplerate* and *Channels*.

The element tagged *Samplerate* has the following two attributes:

- Value, i.e. the numerical value of the sample rate,
- Unit, i.e. the physical unit of the value. Valid values are Hz, KHz and MHz.

The element tagged *Channels* describes the individual sample, it contains three attributes and one or two sub elements. The attributes are

- *Bits*, i.e. the number of bits,
- *Container*, i.e. an indicator of the number of bytes used to for the bits as well as an indication of the interpretation of the bits. Valid values are:
 - o *int8*, the bits are to be interpreted as a signed 8 bit value, e.g. data from the Hackrf;
 - o *uint8*, the bits are to be interpreted as an unsigned 8 bit value, e.g. data from DAB sticks;
 - o *int16*, the bits are to be interpreted as a signed 16 bit value, e.g. used for SDRplay, AIRspy and LimeSDR data;
 - o *int24*, the bits are to be interpreted as a signed 24 bit value;
 - o *int32*, the bits are to be interpreted as a signed 32 bit value;
 - o *float*, the bits are to be interpreted as a 32 bit float number.
- The sub elements with a tag *Channel*, tell whether or not a sample has an *I* element, a *Q* element, both, and in case of both, the order.

4.2 The Element tagged *Datablocks*

While the sub element tagged *Sample* describes the individual samples, the data to be interpreted is described here. The sub element contains one or more sub elements tagged *Datablock*. A valid XML file may contain more than one data segment.²

4.3 The Element tagged *Datablock*

A sub element tagged *Datablock* describes the data segment to be interpreted. Attributes are:

² At the time of writing the interpreter in the Qt-DAB software supports is just processing a single sub element tagged *Datablock*.

- *Number*, useful if the file contains more than one data segment;
- *Count*, a number telling the number of elements, as specified by the *Unit* attribute;
- *Unit*, either *Channel*, i.e. either the *I* or the *Q* elements of the sample, or *Sample*, an element as described by the sub element tagged *Sample*.

The “Datablock” sub element itself has two further optional sub elements:

- A sub element tagged *Frequency*, with the attribute *Value* telling the frequency of the receiver expressed as indicated by the *Unit* attribute. The *Unit* attribute takes as values *Hz*, *KHz* and *MHz*.
- A sub element tagged *Modulation*, with the *Value* attribute as an indicator what the recording is about.

5 OPTIONAL SUB ELEMENTS *RECORDER*, *DEVICE*, *TIME*

To identify the recording, the recording software, the device used and the time of recording can be made part of the specification.

- The sub element tagged *Recorder* has as attributes *Value*, a free format value naming the software creating the file, e.g. QIRX, and an optional attribute *Version*, a free format value, e.g. 3.2-Beta.
- The sub element tagged *Device* has as attribute *Name* with a free format string naming the device as value and an optional attribute *Model* with as value a free format string telling the model of the device.
- The sub element *Time* has as attribute *Value* with as value a string denoting the time of recording, and an attribute *Unit* with as value an indication of the time zone.

6 SUPPORTED DEVICES

The QIRX software is now capable of generating XML files (.raw file extension) for

- Common DAB Sticks,
- sdrplay devices,
- Airspy devices.

The Qt-DAB software now supports the generation of raw XML files (.xml file extension) for

- SDRplay devices (separate drivers with a generator for 2.13 and 3.06 libraries);
- AIRspy devices;
- Common DABsticks;
- Hackrf devices;
- LimeSDR devices;