CxF – Color Exchange Format

The universal language to communicate colors digitally

Abstract: In a global world, communicating electronically the color data of spot colors is a hot (and not yet solved) issue. Global workflows and value chains, whether they are found in a B2B or B2C process, as well as an in-house digital workflow from designer to press, are demanding adequate and effective means to communicate (spot) colors. CxF is a new standard allowing seamless, worldwidedigital communication of all commercially significant aspects of spot colors. Furthermore, CxF is defined in a completely open way so that all aspects of a color can be communicated, even when the application and the color communication features required are unknown. For example, every software vendor implementing / supporting CxF is able to extend the basic feature set to the needs of a new application without affecting the general usability. Wherever color communication is mission critical, CxF should be considered to be the solution to the problem!

1 How to communicate color?

1.1 Background and history of color communication

The best way to do standardized color communications has been in discussion for years and many different approaches have been tried. Typically, color communication is done today using colorimetric measurement values such as CIE-Lab, XYZ, RGB, density, CMYK or spectral measurement values. Today, the new economy is based on a worldwide digital workflow. The evolving use of Internet technology in distribution and logistics creates an even greater demand for a standardized color communication method. A new universal language to communicate colors would simplify the process to communicate colors greatly.

1.2 Device dependent versus device independent color spaces

An important aspect in color communications is to differentiate whether device dependent or device independent color spaces are used. For a long time, the way to communicate colors has been to exchange CMYK, (Scanner-) RGB values or to use named colors like Pantone®, Toyo or HKS.

When device dependent color spaces are used to communicate, color profiles (ex. ICC-profiles) assigned to those color coordinates must be used, to ensure accurate color communications. The appropriate algorithms, procedures and data exchange standards therefore have been defined by the ICC (International Color Consortium; http://www.color.org). Device dependent color spaces are specified therein by so called device profiles. The detailed specifications can be found in the ICC document "Spec ICC.1: 1998-09".

1.3 Named color spaces

Other approaches to communicate colors are “named color spaces”. This approach is used by companies and organizations like Pantone®[1], RAL, NCS, Toyo or HKS. In this approach, colorimetric values / spectral values are assigned to color names. The assigned color name is then used to communicate the color.

1.4 Appearance effects

Human perception of colors is not defined ultimately by a colorimetric measurement value of the sample. As important as the colorimetric value itself, are surrounding colors, or the absolute brightness (light level). One possible approach to that problem can be found in the CIE publication CIE 131-1998 (“The CIE 1997 interim colour appearance model (simple version) CIECAM 97s”). Other mathematical models are known as well.

Further effects affecting the appearance of colors, are angular dependencies of the emission or reflectance, as typically seen on metallic surfaces. This effect is measured using a gonio spectrophotometer. The color has to be defined in that situation by a set of, angular dependent, reflectance curves.

In many situations the appearance / perception of the color is affected not only by the color itself. As important as the color is the substrate the color is printed / applied on. Fluorescence effects, implied by the use of optical brighteners affect the color significantly. Appropriate measures to quantify those effects are white and yellow indexes or as a preferred alternative substrate measurements done with a double monochromatic spectrophotometer in a wavelength range from 360 nm to 730 nm. This data must be communicated as well, to communicate those colors.

In many applications, homogeneousity and structure of the sample are important in color communications. A possible way to solve that problem is to communicate pictures of spot colors in combination with the colorimetric / spectral information.

Other effects affecting the perception of colors are surface effects of the color sample (gloss effects of the color and or the substrate). These are effects typically measured by a gloss meter built according to DIN 16357. A universal color communication language should support such appearance effects.

Other important aspects in color communication are size, position and shape of color patches. There are several papers published defining appearance effects based on the simultaneous contrast of color patches.

---

[1] Pantone®, is a registered Trademark of Pantone Inc., Carlstadt, New Jersey

1.5 Commercial aspects

Commercially used colors have to be within a certain colorimetric tolerance field. This tolerance is often defined as a dE tolerance using the CIE-Lab, CMC or FMCII color space. To communicate the color, dE implies the communication of the acceptable color tolerance field as well.

Other factors, such as light resistance, resistance against chemicals, or other physical, chemical, biological aspects of a color matter may also need to be communicated, depending on the application.

1.6 Mathematical and optical conditions

Absolute colorimetric values depend further on the physical / optical configuration used to do the measurement. Measurements done on the same sample using a 45/0° optical system versus a sphere will not match. Other well-known optical setups affecting the measured results include polarization filters and the physical light source used to illuminate the sample (ie. D65-flash or A-Tungsten halogen). In general it is not possible to convert spectrophotometer readings, done with different optical setups. Therefore, to compare and communicate those spectrophotometer readings, the physical optical setup conditions used to do the measurements must be communicated as well.

Another important aspect, as far as colorimetric values are used to communicate, are the mathematical conditions used to do the calculations (observer 2°/10°, light sources D65, D50, A, C, F1...F11). Depending on the illuminating light source, the colorimetric values will differ. Therefore all commercially known color measurement devices, define these conditions. It may be useful in some conditions, to do the above-mentioned calculations on the spot. A universal data communication standard must therefore be able to include physical light sources or emissions standards as well.

1.7 ANSI standard to communicate colors

One possible way to communicate colors (especially useful to communicate colorimetric values of device profiling charts) using absolute colorimetric values is explained in detail in the ANSI Standard IT 8.7/2-1993. ("Graphic technology - Color reflection target for input scanner calibration"). Therein especially, the aspect is explained how colorimetric and spectral measurement values, can be serialized into an ASCII data stream.

1.8 Summary

In general, it is not sufficient to communicate a reflectance curve or a CIE-Lab value. Depending on the application, there are specific needs in the way a color should be communicated. A universal color communication language must be open – to describe and communicate such known and even new, not yet defined effects.

Depending on the application, further attributes need to be assigned to spot colors. Among the infinite list of possible attributes assigned to spot colors are, serial numbers, part numbers, color mixtures, price of pigments, light resistance of the color, descriptions, application notes, comments and many more.

To define and communicate a color, dependent on the application, an open set of selected attributes of a color has to be communicated. Today's existing color exchange formats, such as the IT8 format, use a table-based approach to store colorimetric and or reflectance values. Every record / line in the file will therefore contain a sample name and the colorimetric or spectral values to be communicated in columns. This approach to store the attributes of a sample has severe limitations. As soon as you store different samples with different attributes, a table-based approach will lead to large tables containing a column for every attribute to be stored. And many of these cells will end up being left blank. Every application that required a new attribute to be stored would require the table to be extended with a new column and therefore an established color data communication standard would have to be changed.

2 The CxF solution to color communications

2.1 Aim of CxF

The Aim of CxF is to define a procedure to communicate all aspects of spot colors in a global value chain by electronic means. To achieve that goal, all relevant attributes affecting the appearance or any other important aspect of the colors needs to be stored (serialized) using a suitable form. Furthermore it must be possible to specify every color in a different way. The specifications have to be open to all software applications dealing with colors. The user or the software application package has to have the choice, to choose the attribute of interest to be used in the application. In general, depending on the application, CxF must support the communication of all further aspects as color recipes, price, weight or any other information that can be helpful. CxF must be a container to store any kind of information assigned to a spot color as well.

CxF must be built in a way, allowing soft- and hardware vendors, to extend the definition, to their specific needs. These extensions, must not affect the backward compatibility, i.e. the new extended attributes should simply be ignored by applications that don't recognize them.

A modern color communication standard has to be Internet aware. The data format chosen should follow standards used today in Internet applications and must therefore seamlessly integrate into existing standards.

Support of color management is a must. Therefore embedding of device profiles must be supported, to allow true color visualization / print out of colors on the remote site and therefore guarantee a seamless workflow.

2.2 The CxF design approach

2.2.1 XML based design

The design approach chosen to implement CxF is to define an XML based language featuring a basic set of attributes used in color communications. This XML based data format must be chosen for the following reasons:

- CxF can be seamlessly integrated into an Internet-based workflow.
- A color communication language must be defined platform and programming language independent.
- XML is a widely accepted standard in the Internet world. There are already many tools available supporting the XML standard.
- Using an XML based approach provides for an ASCII data stream. An ASCII stream will simplify the communication.
- It is easy to embed an XML data stream into objects.
- The ASCII stream can be read and understood by human beings as well. An ASCII capable editor is therefore the only tool required to do the first steps in composing / editing CxF files.
- XML is perfectly suited to define and serialize a set of hierarchically structured data objects – each data object having an open set of attributes.
- XML is a perfect tool, to achieve persistency of objects found in OO programming languages.

2.2.2 Support of ICC profiles

Today color communication software packages must be aware of the ICC approach. CxF allows a set of device ICC profiles to be assigned to every sample. These profiles can be used to either transform a device dependent color value into a device independent one, or to simulate a color on an output device.

2.2.3 Links

Often there is a need to assign the same information to a set of samples. Typical examples are measuring conditions or ICC profiles. To avoid encoding recurrent / redundant information CxF allows a link to an object inside the CxF compatible file to be defined. In this way, ICC profiles or measuring conditions only need to be stored once in a file. The profile or measuring conditions can then be referenced by this link – thus greatly reducing the required size to store the information.

2.3 A typical CxF compatible system

A CxF compatible system is built of the following components:

- A spectrophotometer or any other device capable of doing colorimetric or spectral readings.
- An application software, (CxF composer) that allows spot colors to be measured or entered (using reflectance data, RGB, CIE-Lab...) and defines or adds additional attributes (pictures, text strings...) to the sample. This software is capable of serializing this data with the help of a CxF library.
- A CxF library. The CxF library is capable of reading and writing spot colors according to the CxF standard. In addition the CxF library will give the application access to the CxF type of data structures and interfaces to colorimetric and color management functions. The data stream produced by the library can be embedded in other data formats.
- A viewer software (CxF viewer) that can load, display and print spot colors found in CxF files in true color. In addition, the CxF viewer can export CxF color patches to different application specific formats, such as Adobe® Photoshop palette files or similar palettes. (**i1Share** is a **free** CxF composer / viewer.
  The application can be downloaded at http://www.i1color.com/freeware/).
- An ICC based color management system, that can synchronize all involved input / output devices. In addition ICC profiles allow device dependent color definitions to be translated into device independent color definitions.
- An optional print device, to get a hard copy of the color patches in true colors as well as to get a printout of the text and picture attributes assigned to colors.
- An appearance modeling system, to visualize the appearance effects of the samples.

The system components interact as follows:

A user will enter the colors to communicate with help of the graphical interface of the CxF composer. Typically the user will do a measurement with an attached spectrophotometer. The CxF composer will then add the new sample to the existing sample set. An alternative way to add a color is to enter the color manually. After the color is defined by colorimetric or spectral data, the user can add additional attributes to the color, such as ICC profiles, pictures or any other attribute required by the application. Typically these attributes will be added using either a drag-and-drop style user interface (specifically suited to add profiles and pictures) or the user will add additional attributes in text form using a ASCII editor or a specialized input mask. The colors added will be immediately visualized in true color using ICC compatible color management. Once the full sample set is defined, the information is serialized into an ASCII compatible data stream. This stream can be stored into a file, or as an alternative solution, can be embedded into a parent object. The file is transmitted afterwards to the recipient. The recipient will then use a CxF compatible viewer, to open and visualize the file in true colors. In addition the user may choose to print out the sample set in true colors. Because of the capability of the CxF viewer to write color data files compatible with the most important applications as Photoshop, Quark Express or similar applications, the user may want to export the color data into a data format compatible with those applications.

2.4 Structure of a CxF compatible data stream

As mentioned, the information describing the color samples are encoded in an XML based form. These XML based streams, may be stored in a file or as an alternative embedded into a third party data object. A CxF data stream is based on the ASCII character set.

The following example CxF file contains the following information:

The first item we find in the sample set "PANTONE®" including the samples "Yellow C", "Warm Red" and "Process Yellow C". The first sample is defined as a CIE-Lab value, a spectral curve, as well as a "Named Color" (Pantone® name of the color). In addition a reference to the corresponding measurement conditions is stored.

The second sample found in the file is defined as a spectral curve.

The last sample is defined as a CIE-XYZ as well as a RGB value. Assigned to the sample is an ICC compatible device profile.

Embedded data objects (ICC compatible profiles, pictures) must be converted into an ASCII data stream. A MIME compatible data format is used to achieve this goal.

2.5 CxF example

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE CXF SYSTEM "file:/G:/ColorLab/ColorLab/CXF/cxf.dtd" >
<CXF>
 <Name>PANTONE</Name>
 <Description>Color Formula Guide 1000</Description>
 <SampleSet>
  <Name>Basic Colors</Name>
  <Description>Basic Colors of this swatchbook</Description>
  <Sample>
   <Name>Yellow C</Name>
   <ColorVector Conditions="UniqueID1">
    <ColorSpace>CIE-Lab</ColorSpace>
    <Value Name="L*">89.62</Value>
    <Value Name="a*">-9.35</Value>
    <Value Name="b*">110.77</Value>
   </ColorVector>
   <NamedColor Conditions="UniqueID2">PANTONE Yellow C</NamedColor>
   <Spectrum Conditions="UniqueID1">
    <SpectrumData>0.0333 0.0303 0.0268 0.0238 0.0223 0.0210 0.0204 0.0207 0.0219 0.0235 0.0289 0.0563 0.1814 0.4662
0.7146 0.8075 0.8385 0.8509 0.8585 0.8663 0.8707 0.8740 0.8762 0.8790 0.8822 0.8834 0.8865 0.8902 0.8923 0.8906 0.8920
0.8931 0.8947 0.8955 0.8937 0.8929</SpectrumData>
   </Spectrum>
   <BinaryData UniqueID="Pic1"/>
  </Sample>
  <Sample>
   <Name>Warm Red C</Name>
   <Spectrum Conditions="UniqueID1">0.0810 0.0748 0.0689 0.0713 0.0749 0.0740 0.0712 0.0622 0.0486 0.0375 0.0309 0.0269
0.0254 0.0258 0.0269 0.0279 0.0304 0.0371 0.0611 0.1522 0.3522 0.5776 0.7340 0.8111 0.8452 0.8598 0.8690 0.8767 0.8810
0.8811 0.8828 0.8846 0.8874 0.8892 0.8880 0.8879</Spectrum>
  </Sample>
  <Sample>
```

```
      <Name>Process Yellow C</Name>
      <ColorVector Conditions="UniqueID3">
        <ColorSpace>XYZ</ColorSpace>
        <Value Name="X">20</Value>
        <Value Name="Y">20</Value>
        <Value Name="Z">30</Value>
      </ColorVector>
      <DeviceColor Conditions="UniqueID4">
        <ColorSpace>RGB</ColorSpace>
        <Value Name="R">55</Value>
        <Value Name="G">88</Value>
        <Value Name="B">145</Value>
        <ICC-ProfileLink UniqueID="A Profile 1"/>
      </DeviceColor>
    </Sample>
  </SampleSet>
  <BinaryObject Name="Pic1" MIME-Type="TIFF">Binary data in the MIME format goes here.</BinaryObject>
  <ICC-Profile Name="A Profile 1">
    <ICC-Data>Store the ICC-File MIME-encoded here.</ICC-Data>
  </ICC-Profile>
  <Conditions>
    <ID>UniqueID1</ID>
    <Attribute Name="Filter">No</Attribute>
    <Attribute Name="Geometry">45/0</Attribute>
    <Attribute Name="Illumination">D65</Attribute>
    <Attribute Name="LambdaMin">360</Attribute>
    <Attribute Name="LambdaMax">720</Attribute>
    <Attribute Name="NrOfDataPoints">36</Attribute>
  </Conditions>
  <Conditions>
    <ID>UniqueID2</ID>
    <Attribute Name="Filter">D65</Attribute>
    <Attribute Name="Illumination">2°</Attribute>
    <Attribute Name="LambdaMin">360</Attribute>
    <Attribute Name="LambdaMax">750</Attribute>
    <Attribute Name="NrOfDataPoints">40</Attribute>
  </Conditions>
  <Conditions>
    <ID>UniqueID3</ID>
    <Attribute Name="Geometry">45/0</Attribute>
  </Conditions>
  <Conditions>
    <ID>UniqueID4</ID>
    <Attribute Name="RGB-Rang">0-255</Attribute>
  </Conditions>
</CXF>
```

2.6 CxF standard tags

The definition of typical tags (objects, attributes) found in a CxF compatible file, can be done using the DTD (document type definition) standard. The next paragraph lists the most important CxF tags:

```
<!--Color eXchange Format-->
<!ELEMENT CXF  (Name,Description?,SampleSet*,Conditions*,ICC-Profile*,Bi naryObject*) >

<!ELEMENT Name  (#PCDATA) >

<!ELEMENT Description  (#PCDATA) >
```

```
<!ELEMENT SampleSet  (Name,Description?,Sample+) >

<!ELEMENT Conditions  (ID,Attribute+) >

<!ELEMENT ICC-Profile  (Name,Description?,ICC-Data) >

<!ATTLIST ICC-Profile Name  CDATA    #IMPLIED  >

<!ELEMENT BinaryObject  (#PCDATA) >

<!ATTLIST BinaryObject
          Name  CDATA    #IMPLIED
          MIME-Type  CDATA    #IMPLIED  >

<!ELEMENT Sample Name,Description?,BinaryData*,Spectrum*, ColorVector*,DeviceColor*,
          NamedColor*,Density*) >

<!ELEMENT ID  (#PCDATA) >

<!ELEMENT Attribute  (#PCDATA) >

<!ATTLIST Attribute Name  CDATA    #IMPLIED  >

<!ELEMENT ICC-Data EMPTY  >

<!ELEMENT BinaryData (Name?,Description?,BinaryDataLink) >

<!ATTLIST BinaryData UniqueID  CDATA    #IMPLIED  >

<!ELEMENT Spectrum(Name?,Description?,SpectrumData) >

<!ATTLIST Spectrum Conditions  CDATA    #IMPLIED  >

<!ELEMENT ColorVector  (Name?,Description?,ColorSpace,Value+) >

<!ATTLIST ColorVector Conditions  CDATA    #IMPLIED  >

<!ELEMENT DeviceColor  (Name?,Description?,ColorSpace,Value+,ICC-Profile Link?) >

<!ATTLIST DeviceColor Conditions  CDATA    #IMPLIED  >

<!ELEMENT NamedColor  (Name,Description?) >

<!ATTLIST NamedColor Conditions  CDATA    #IMPLIED  >

<!ELEMENT Density  (Name?,Description?,DensityData+) >

<!ATTLIST Density Conditions  CDATA    #IMPLIED  >

<!ELEMENT BinaryDataLink   EMPTY  >

<!ELEMENT SpectrumData  (#PCDATA) >

<!ELEMENT ColorSpace  (#PCDATA) >

<!ELEMENT Value  (#PCDATA) >

<!ATTLIST Value Name  CDATA    #IMPLIED  >

<!ELEMENT ICC-ProfileLink   EMPTY  >

<!ATTLIST ICC-ProfileLink UniqueID  CDATA    #IMPLIED  >
```

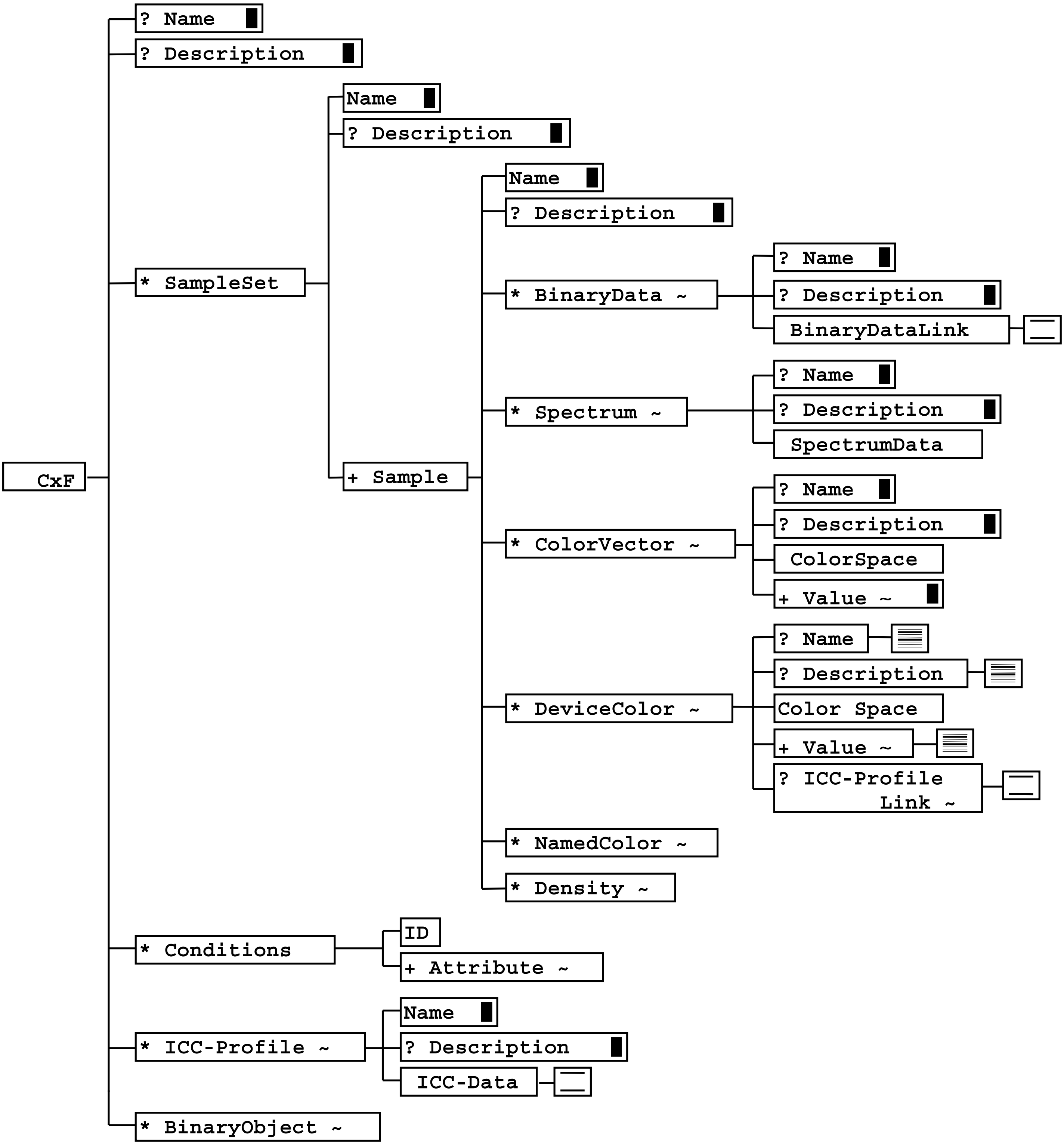
<!ELEMENT DensityData (#PCDATA) >

<!ATTLIST DensityData Filter CDATA   #IMPLIED >

GretagMacbeth

2.7 CxF tree

An alternative way to visualize the defintions above, is to use a tree like structure:



2.8 CxF SDK

To support the use of CxF compatible data files, GretagMacbeth has developed a CxF software development kit (SDK). The functionality of a CxF compatible application may be structured into 5 layers. On top is the CxF application layer. A CxF application will access layer 1 to 3 for colorimetric or spectral sample definitions. Layer 4 serves to apply color management or appearance modeling functions to this information.

<table>
<tr><td colspan="3" align="center">CxF compatible application</td></tr>
<tr><td align="center">color management</td><td align="center">appearance modeling</td><td align="center">…
…</td></tr>
<tr><td colspan="3" align="center">layer 3</td></tr>
<tr><td colspan="3" align="center">layer 2</td></tr>
<tr><td colspan="3" align="center">layer 1</td></tr>
</table>

Going a little bit more into the details of the CxF SDK we can assign the following functionality to the main CxF layers:

Layer 4: CxF – appearance / perception / color management layer C++ Software Development Kit (SDK) Access to color management, appearance information
Layer 3: CxF – colorimetric data layer C++ Software Development Kit (SDK) high level C++ access to colorimetric data
Layer 2: CxF – data structure layer C++ Software Development Kit (SDK) low level C++ access to CxF data structures
Layer 1: CxF – file layer File-format: XML

Layer 1: CxF file layer. Layer 1 is a representation of the tools / methods and data structures used to load and store CxF compatible data files to / from disk or to / from a memory stream.

Layer 2: CxF data structure layer. Layer 2 will parse the basic IO / memory stream defined in layer 1 and transform the attributes and objects found in the stream into a tree like data structure in memory. In addition layer 2 will give the application access to this data structures. All objects and attributes are stored as ASCII strings on layer 2. No conversion to binary data structures is done on layer 2.

Layer 3: CxF – colorimetric layer: Layer 3 does provide access to all kinds of colorimetric information found in a CxF compatible file. Tools are provided to iterate through the tree to find the appropriate colorimetric values. Output of layer 3 is binary formatted data.

Layer 4: CxF – appearance / perception / color management layer: Once colorimetric values are extracted using layer 3 functionality, depending on the application, the colorimetric values must be adapted / transformed into a visual representation of the (spot) colors. To achieve this goal, layer 4 functionality of CxF is used to apply the rules of color management and appearance management to the sample.

2.9 The Future of CxF

In former times, to communicate a Pantone® number or a CMYK value was enough. Today we know that we need to communicate much more information to get the color right the first time.

CxF can be the base that can be added to any digital workflow to provide the ability to communicate spot colors electronically. In a digital workflow from prepress to press, no standardized ways to communicate all aspects of spot colors in the whole value chain is known. CxF will open new opportunities to automate and simplify the color communication of spot colors in a digital workflow.

In a world moving from analog proofs to electronic proofs, we need to extend the capabilities of our workflow tools. They must be extended to allow proofing of all appearance effects of the color. The more electronic the workflow becomes, the more significant the aspect to control color and appearance of a spot color becomes.

PDF and PostScript allow electronic proofing of pictures – to a certain extent. However, the exact specification and communication of spot color has not been implemented so far. CxF can do in the future for spot colors, what PDF and PostScript has done for documents in the past.

2.10 The CxF committee

As a matter of fact, because of the open definition of the CxF standard, CxF will never be finally defined. All attributes stored in a CxF compatible file, used to communicate between software packages of different vendors, needs, of course, to be identified. To simplify and organize this task, the CxF committee will be set up. Every organization or company interested in color communications can become a member of the CxF committee.

2.11 Questions and Answers about CxF Technology

- What do I need to build my own CxF compatible application?
  We recommend licensing the free CxF SDK from GretagMacbeth to write CxF compatible applications.

- Where do I get the free CxF SDK (software development kit)?
  You can get the CxF SDK from GretagMacbeth AG. Send an email to:
  mailto:cxf@gretagmacbeth.ch

- How do I get the free CxF viewer / composer?
  A free CxF compatible viewer / composer "(i1Share") can be downloaded from the GretagMacbeth Web site. http://www.i1color.com/freeware/. A CxF compatible composer will also be bundled with select GretagMacbeth instruments. In addition, all Gretag-Macbeth soft- and hardware products will support CxF as a color communication standard. Check http://www.gretagmacbeth.com for an update of your application and/or firmware.

- How do I embed a CxF compatible data stream into my application?
  We recommend using the free GretagMacbeth CxF SDK. The SDK has functionality to redirect a CxF compatible data stream into a memory stream.

- How to participate in future CxF development?
  Any organization interested in CxF can become a member of the CxF committee. Contact mailto:cxf@gretagmacbeth.ch to get additional information.

- How do I become member of the CxF consortium?
  To become a member of the CxF consortium send an email to mailto:cxf@gretagmacbeth.ch