A HUMAN READABLE FORM OF THE DCC

Muhammed-Ali Demir* [a], Moritz Jordan [a], Thomas Krah [a], Shanna Schönhals [a], Siegfried Hackel [a],
Frank Härtig [a], Thorsten Schrader [a], Jan Loewe [a], Lutz Doering [a], Benjamin Gloger [a],
Justin Jagieniak [a], Gamze Söylev-Öktem [a]

[a] Physikalisch-Technische Bundesanstalt (PTB), Bundesallee 100, 38116 Braunschweig, Germany
Email address: muhammed.demir@ptb.de, moritz.jordan@ptb.de, thomas.krah@ptb.de,
shanna.schoenhals@ptb.de, siegfried.hackel@ptb.de, frank.haertig@ptb.de, thorsten.schrader@ptb.de,
jan.loewe@ptb.de, lutz.doering@ptb.de, benjamin.gloger@ptb.de, justin.jagieniak@ptb.de,
gamze.soeylev-oektem@ptb.de
*Corresponding author: muhammed.demir@ptb.de

Abstract - One of the benefits of the Digital Calibration Certificate (DCC, [1] and [2]) as shown by PTB is to have flexible ways to transform the resulting DCC document since it is a digital document. In order to better visualise the content for human beings, or for a transitional period to print an analogue copy of the DCC XML [3] document with sign and seal, a feature for transforming the DCC to a human readable form is needed. We will call this form human readable (HR) (see Fig. 1), and it is an optional part of the DCC. This paper will show how to generate the human readable form of the DCC by using the eXtensible Stylesheet Language (XSLT [4]).

In section 1 we will give a short review of the DCC schema (XSD) and describe the structure of an exemplary DCC (XML). In section 2 we will suggest how to generate the human readable of a DCC document. First, we will discuss for which purpose the human readable can be used. Then, we show how the human readable can be generated by using XSLT. Finally, we will give an outlook on future steps and topics to be solved.

***Keywords*: DCC, Human Readable, XML, XSLT**

1. THE DIGITAL CALIBRATION CERTIFICATE

It is not the primary objective of this section to cover every aspect of the digital calibration certificate (DCC) but to give a short summary. The purpose of the DCC is to electronically store and transmit authenticated, encrypted and signed calibration results. Furthermore, it enables them to be uniformly interpreted (see [1, p. 1]).

To give a visual example, the schema of the DCC is the foundation, on which the house, the DCC document, is built upon (Fig. 1). If we could add a carport to the existing building on the same foundation, without having to damage neither the foundation nor the structure of the existing building, we would have a home for the HR and could generate it in a flexible, nonintrusive manner.
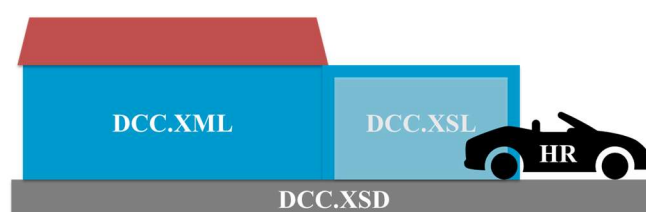

Fig. 1: The relation between XML, XSLT and Human Readable

The DCC is written in eXtensible Markup Language (XML). The corresponding schema (XSD) regulates the structure of the DCC. It defines the possible elements and their data types, the number of child elements they can hold, and whether they are mandatory or optional [5]. A DCC can be validated against the schema using external software tools, e.g., Notepad++.

A DCC consists of the four areas administrative data, measurements results, comments, and document. Only the first two areas are mandatory (Fig. 2) We will describe them in more detail below.
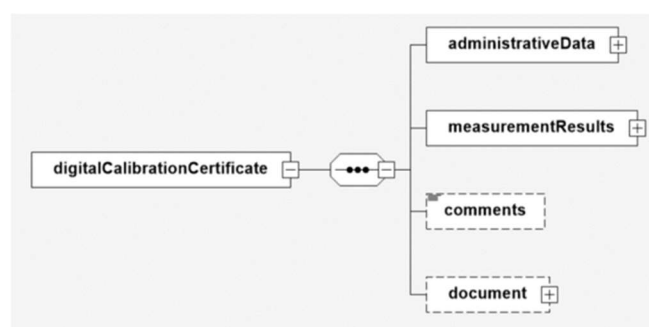

Fig. 2: The content of the DCC schema (XSD) (from [1])

The PTB provides Good Practise examples (GP) for the temperature DCC with various complexity. These show what a valid DCC could look like. For this paper, we take a look at two sections from the Good Practice (GP) example for Temperature Simplified [6]. We will use them as a showcase in the upcoming sections to explain how the human readable form can be generated from the XML data.

In Fig. 3, the *coreData* block in the *administrativeData* area is shown. It contains elements such as *uniqueIdentifier* which hold a simple alphanumeric value, but also elements such as *identifications* that contain a list of child elements.

```
<dcc:administrativeData>
  <dcc:dccSoftware>…
  </dcc:dccSoftware>
  <dcc:coreData>
  <dcc:countryCodeISO3166_1>DE</dcc:countryCodeISO3166_1>
  <dcc:usedLangCodeISO639_1>de</dcc:usedLangCodeISO639_1>
  <dcc:usedLangCodeISO639_1>en</dcc:usedLangCodeISO639_1>
  <dcc:mandatoryLangCodeISO639_1>de</dcc:mandatoryLangCodeISO639_1>
  <dcc:uniqueIdentifier>GP_DCC_temperature_simplified_1.2
    </dcc:uniqueIdentifier>
  <dcc:identifications>
    <dcc:identification>
      <dcc:issuer>calibrationLaboratory</dcc:issuer>
      <dcc:value>string-calibrationLaboratory-coreData
        </dcc:value>
      <dcc:name>
        <dcc:content lang="de">Auftrags Nr.</dcc:content>
        <dcc:content lang="en">Orde no.</dcc:content>
      </dcc:name>
    </dcc:identification>
  </dcc:identifications>
  <dcc:beginPerformanceDate>1957-08-13</dcc:beginPerformanceDate>
  <dcc:endPerformanceDate>1957-08-13</dcc:endPerformanceDate>
  <dcc:performanceLocation>laboratory</dcc:performanceLocation>
```

Fig. 3: The *coreData* block in the administrative data of the temperature simplified GP

In Fig. 4 the quantity for the measurement error from the block of *measurementResults* is shown. It contains complex data structures, e.g., lists and tables.

The aim of the DCC is to be machine readable/ executable, machine interpretable and even machine controllable [7, p. 7]. Since a digital document is optimised for machines, it may not be easy to read for human beings. However, it is possible to transform XML documents. Therefore, in the next section we will have a detailed look at the human readable form of the DCC and how we can transform the DCC XML.

2. THE HUMAN READABLE FORM OF THE DCC

So, the DCC XML can be difficult to read and interpret by human beings. This is one of the reasons a human readable might be useful. It could also be helpful in order to convince people of the DCC by offering a human readable form similar to an analogue calibration certificate. It could be the case that an analogue document must be generated sporadically for verification or other reasons. In the future it could even be printed and signed, so we would have an exact digital copy of the analogue calibration certificate.

```
<dcc:measurementResults>
  <dcc:measurementResult>
    <dcc:name>
      <dcc:content lang="de">Messergebnisse</dcc:content>
      <dcc:content lang="en">Measurement results</dcc:content>
    </dcc:name>
    <dcc:usedMethods> … </dcc:usedMethods>
    <dcc:measuringEquipments> … <dcc:measuringEquipments>
    <dcc:influenceConditions> … </dcc:influenceConditions>
    <dcc:results>
      <dcc:result refType="gp_measuringResult1">
        <dcc:name>
          <dcc:content lang="de">Messergebnisse</dcc:content>
          <dcc:content lang="en">Measuring results</dcc:content>
        </dcc:name>
        <dcc:data>
          <dcc:list refType="gp_table1">
            <dcc:quantity refType="basic_referenceValue">
              <dcc:name>
                <dcc:content lang="de">Bezugswert</dcc:content>
                <dcc:content lang="en">Reference value</dcc:content>
              </dcc:name>
              <si:hybrid>
                <si:realListXMLList>
                  <si:valueXMLList>
                    306.248 373.121 448.253 523.319 593.154
                  </si:valueXMLList>
                  <si:unitXMLList>\kelvin</si:unitXMLList>
                </si:realListXMLList>
                <si:realListXMLList>
                  <si:valueXMLList>
                    33.098 99.971 175.103 250.169 320.004
                  </si:valueXMLList>
                  <si:unitXMLList>\degreecelsius</si:unitXMLList>
                </si:realListXMLList>
              </si:hybrid>
            </dcc:quantity>
```

Fig. 4: The quantity for the reference value in the measurement results of the temperature simplified GP

The good news is that it is possible to transform XML documents. We will continue this section with the goal to make all the data contained in a DCC XML visible in one or more languages if available and show it with examples.

2.1 Introduction

From the perspective of the human readable, the DCC XML is simply a data container we want to extract the data from. There are many ways to extract and transform the data, for example by using programming languages such as Java or C# or languages such as XPath [8] or XSLT, which were specifically designed for addressing different parts of an XML document or transforming it into other formats. Since we are already using XML and XSLT is XML-like syntax, we will use it in order to generate a human readable form as an HTML. This will also enable the user to print the document to PDF via the browser if necessary. By using XSLT, we will also try to keep the programming overhead minimal, and if the user understands XML and XPath, the understanding of XSLT is simple. Besides that, we will be independent of the operating system (OS) and a lot of other dependencies. In the next subsection we will describe how to achieve this based on our examples from the previous sections.

2.2 Use of XSLT to generate the human readable form of the DCC

In order to obtain a complete XSLT, we must cover every possibility given by the DCC schema (dcc.xsd). Since the DCC consists of blocks, e.g., administrative data, we will cover each block one after another in the order defined in the schema and try to keep the flexibility of the schema in the XSLT by using template definitions. This will also allow us to avoid redundancies. In the end, we will have a possibility to visualise DCCs being valid regarding the given DCC schema. Since the DCC schema will be adapted over time, the XSLT for the human readable will have to be updated as well.

Before we go into the details, there are a few types of data that could be in the DCC XML (see [6]):

Data type	Example tags
plain text	e.g., <dcc:mandatoryLangCodeISO639_1>
pictures *not part of the GP*	e.g., <dcc:descriptionData>
multilingual text	e.g., <dcc:name>

Value of plain text can be selected as shown in Fig. 5. Base64 data of images can be visualised as shown in Fig. 6 Multilingual text can be selected as shown in Fig. 7 to Fig. 9.

```
<xsl:value-of select="dcc:mandatoryLangCodeISO639_1"/>
```

Fig. 5: Selecting value of plain text in XSLT

```
<xsl:variable name="fileName"
  select="dcc:descriptionData/dcc:fileName" />
<xsl:variable name="mimeType"
  select="dcc:descriptionData/dcc:mimeType" />
<xsl:variable name="data"
  select="dcc:descriptionData/dcc:dataBase64" />
<xsl:if test="starts-with($mimeType, 'image')">
  <p>
   <img width="250">
     <xsl:attribute name="src">
      data:<xsl:value-of select="$mimeType"/>;base64,
       <xsl:value-of select="$data"/>
     </xsl:attribute>
     <xsl:attribute name="alt">
       <xsl:value-of select="$fileName"/>
     </xsl:attribute>
   </img>
  </p>
</xsl:if>
```

Fig. 6. Visualising Base64 data of images

```
<xsl:template match="dcc:name">
  <xsl:call-template name="richContent">
    <xsl:with-param name="element" select="."/>
  </xsl:call-template>
</xsl:template>
```

Fig. 7: Selecting value of multilingual text in XSLT with sub template "*richContent*"

```
<xsl:template name="richContent">
  <xsl:param name="element" select="."/>
  <div class="avoid-break mb-default">
    <xsl:for-each select="$element/dcc:*">
      <xsl:if
 test="self::dcc:content[@lang=$primaryLang]">
        <p class="text-primary">
          <xsl:value-of select="."/>
        </p>
      </xsl:if>
    </xsl:for-each>
    <xsl:for-each select="$element/dcc:*">
      <xsl:if
 test="self::dcc:content[@lang=$secondaryLang]">
        <p class="text-secondary">
          <xsl:value-of select="."/>
        </p>
      </xsl:if>
    </xsl:for-each>
    <xsl:for-each select="$element/dcc:*">
      <xsl:if test="self::dcc:content[not(@lang)]">
        <p class="text-default">
          <xsl:value-of select="."/>
        </p>
      </xsl:if>
      …
    </xsl:for-each>
  </div>
</xsl:template>
```

Fig. 8: sub template „*richContent*"

```
<xsl:variable name="primaryLang"
select="dcc:digitalCalibrationCertificate/dcc:administ
rativeData/dcc:coreData/dcc:mandatoryLangCodeISO639_1"
/>

<xsl:variable name="secondaryLang"
select="dcc:digitalCalibrationCertificate/dcc:administ
rativeData/dcc:coreData/dcc:usedLangCodeISO639_1[text(
) != $primaryLang]"/>
```

Fig. 9: Variables for primary and secondary language of the DCC XML

The corresponding template in the XSLT is structured in the following way: first, we define templates from outer to inner tag that we want to extract the data from. Then, we match the template with the tags of the DCC XML, e.g., administrativeData (see Fig. 10). The results with the XSLT applied to the mentioned good practice temperature DCC will give the following results. The output for the *administrativeData* block is quite simple, one by one we have the data from the corresponding good practice example (see Fig. 11).

```
<xsl:template match="dcc:administrativeData">
  <table class="administrative-data table">
    <xsl:apply-templates select="dcc:dccSoftware"/>

    <xsl:apply-templates select="dcc:coreData"/>

    <xsl:apply-templates select="dcc:items"/>

    <xsl:apply-templates
select="dcc:calibrationLaboratory"/>

    <xsl:apply-templates select="dcc:respPersons"/>

    <xsl:apply-templates select="dcc:customer"/>
  </table>
</xsl:template>
```

Fig. 10: XSLT template for *administrativeData* block

Country code ISO3166_1 *Ländercode ISO3166_1*	DE
Country code ISO639_1 (used) *Ländercode ISO639_1 (benutzt)*	de en
Country code ISO3166_1 (mandatory) *Ländercode ISO639_1 (verpflichtend)*	en
Unique Identdifier *Kalibrierzeichen*	GP_DCC_temperature_simplified_1.2
Issuer *Herausgeber*	calibrationLaboratory
Order no. *Auftrags Nr.*	string-calibrationLaboratory-coreData
Date of Calibration *Datum der Kalibrierung*	1957-08-13
Place of Calibration *Ort der Kalibrierung*	laboratory

Fig. 11: The output of the human readable for the specified part of the *administrativeData* block

Measuring results
Messergebnisse

Reference value Bezugswert \kelvin	306.248 373.121 448.253 523.319 593.154
Reference value Bezugswert \degreecelsius	33.098 99.971 175.103 250.169 320.004

Fig. 12: The output of the human readable for *administrativeData* block

The output for the first result of the *measurementResults* block will result in a header and a table (see Fig. 12).

3. CONCLUSIONS

In this paper, we have shown that it is possible to generate a human readable form of the DCC via XSLT and have given examples based on the available good practice of the temperature DCC of the PTB.

A topic to cover in the future could be the enhancement of the human readable to be dynamic and multilingual. Another topic could be to develop a concept of how to visualize different forms of tables, e.g., when to use tables with vertical column headers or horizontal row headers. Finally, formulas and their visualisation would have to be covered by the human readable. Updates of the DCC schema will require updates of the XSL for the human readable.

Please have a look at https://www.ptb.de/dcc for more information and feel free to contribute to the project.

ACKNOWLEDGMENTS

REFERENCES

[1] Hackel, Siegfried Gustav, Härtig, Frank, Hornig, Julia, and Wiedenhöfer, Thomas, 'The Digital Calibration Certificate', *PTB-Mitteilungen Vol. 127 2017*, vol. Issue 4, p. 7, Dec. 2017, doi: 10.7795/310.20170403. [Online]. Available: https://oar.ptb.de/resources/show/10.7795/310.20170403. [Accessed: Mar. 16, 2021], doi:10.7795/310.20170403

[2] 'Digital Calibration Certificate - DCC', Apr. 14, 2021. [Online]. Available: https://www.ptb.de/cms/en/research-development/into-the-future-with-metrology/the-challenges-of-digital-transformation/kernziel1einheitlichkeitim/digital-calibration-certificate-dcc.html. [Accessed: Apr. 23, 2021]

[3] 'Introduction to XML'. [Online]. Available: https://www.w3schools.com/xml/xml_whatis.asp

[4] 'XML and XSLT'. [Online]. Available: https://www.w3schools.com/xml/xml_xslt.asp. [Accessed: Aug. 09, 2021]

[5] 'XML Schema Tutorial'. [Online]. Available: https://www.w3schools.com/xml/schema_intro.asp. [Accessed: Jan. 23, 2021]

[6] 'Good Practice', *Digital Calibration Certificate - Wiki*. [Online]. Available: https://dccwiki.ptb.de/de/gp_home. [Accessed: Apr. 13, 2022]

[7] Damian Czarny *et al.*, 'Szenarien zur Digitalisierung der Normung und Normen', 2021 [Online]. Available: https://www.dke.de/de/normen-standards/normung-strategie/deutsche-normungsstrategie/idis

[8] 'XPath Tutorial'. [Online]. Available: https://www.w3schools.com/xml/xpath_intro.asp. [Accessed: May 18, 2021]