Deliverable D5

Title

Good practice guides SmartCom validation

(1) Test for communication interfaces used for the exchange of metrological data
(2) Conformity test for unified DCCs
(3) TraCIM system

EMPIR Grant Agreement number

17IND02

The EMPIR initiative is co-funded by the European Union's Horizon 2020 research and innovation programme and the EMPIR Participating States

Project short name

SmartCom

Leading partner

PTB (Physikalisch-Technische Bundesanstalt)

Due date	Submission date
2020-05-31	2020-07-02

DOI

10.5281/zenodo.3816696

SmartCom

Good Practice Guides SmartCom Validation

(1) Test for communication interfaces used for the exchange of metrological data

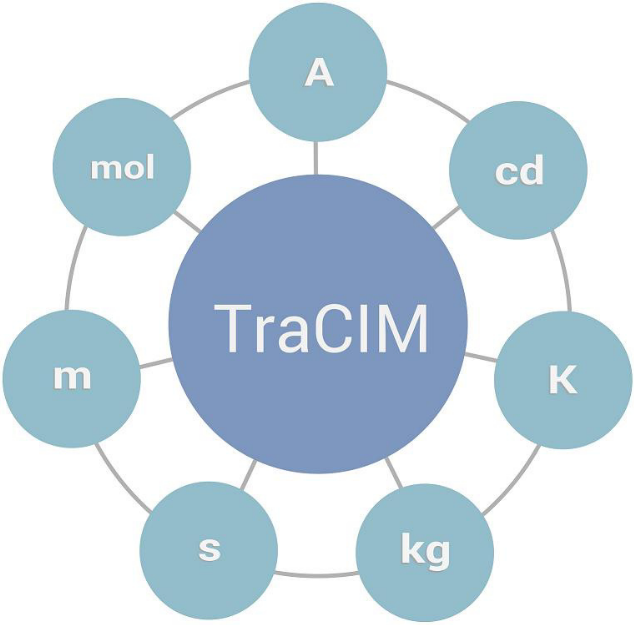
(2) Conformity test for unified DCCs

(3) TraCIM system

TraCIM D-SI

EN

DOI: 10.5281/zenodo.3816696



Good practice guide

Test for communication interfaces used for the exchange of metrological data

Version 1.0

DOI: 10.5281/zenodo.3816696

Editors

National Physical Laboratory, United Kingdom:
I. Smith, Y. Luo

Physikalisch Technische Bundesanstalt, Germany:
D. Hutzschenreuter, S. Schönhals

Comprising the results from our research and the fruitful and intensive discussions with all our other project partners and stakeholders worldwide.

Contact: smartcom@ptb.de

Teddington June 2020

DOI: 10.5281/zenodo.3816696

Table of Contents

DOI: 10.5281/zenodo.3816696

1 Introduction

The European Metrology Programme for Innovation and Research (EMPIR) Joint Research Project (JRP) “Communication and validation of smart data in IoT-networks” (short name “SmartCom”) [1] is concerned with establishing a means to exchange metrological data in communication networks.

A key output of the SmartCom JRP is the “Digital System of Units” (referred to subsequently as the “D-SI”) [2]. The D-SI provides a framework that allows metrological data, including associated uncertainty information, to be stored and transmitted in digital form. The D-SI forms the basis for the harmonised, clear, secure and economical exchange of measurement results in the case that those results are stored and transmitted as digital data. The D-SI allows real and complex quantities, and univariate and multivariate quantities to be specified. In addition to a general framework for metrological data, an implementation of the framework in Extensible Markup Language (XML) [3] has been developed within the SmartCom JRP.

For communication interfaces that transfer metrological data, it is imperative that the data conforms to all conditions deemed to be applicable. Examples include the requirements that only units identified within the BIPM SI Brochure [4] are used, that an expanded uncertainty must be a positive decimal number, and that a coverage probability must lie in the closed interval [0, 1]. It is therefore highly recommended that testing of metrological data is undertaken prior to transmission.

For the XML implementation of the D-SI, it is generally possible to undertake a subset of the necessary tests by checking that data provided in XML format is both well-formed and adheres to the XML schema for metrological data. However, since these checks are unable to cover all the required tests, it is necessary to develop a means to implement additional testing.

This document describes the framework conditions for the online validation of communication interfaces that are used for the exchange of metrological data.

DOI: 10.5281/zenodo.3816696

2 General XML structure of metrological data in SmartCom

This guide considers version 1.3.1 of the D-SI XML structure which was developed within the SmartCom project. The XML Schema Definition (XSD) of the D-SI can be obtained from [5]. The XML structure allows metrological data to be stored in a machine-readable form.

For example, consider the following human-readable information relating to a measurement:

“The measured temperature is 20.10 °C with an expanded uncertainty of 0.50 °C corresponding to a coverage probability of $p = 0.95$ using a coverage factor of $k = 2$, where it is assumed that the probability distribution for the temperature is normal.”

The same information is represented in machine-readable form as:

```
<si:real>
  <si:label>temperature<\si:label>
  <si:value>20.10<\si:value>
  <si:unit>\degreecelsius<\si:unit>
  <si:expandedUnc>
    <si:uncertainty>0.50<\si:uncertainty>
    <si:coverageFactor>2<\si:coverageFactor>
    <si:coverageProbability>0.95<\si:coverageProbability>
    <si:distribution>normal<\si:distribution>
  <\si:expandedUnc>
<\si:real>
```

The D-SI XML structure allows real quantities, complex quantities, multivariate (real and complex) quantities to be represented.

3 Testing the syntax of metrological data

For data provided within an XML file, the tests that can be undertaken fall into three main categories: checking for well-formedness of the XML, validation against the XSD of the D-SI, and additional validation of the XML (that cannot be undertaken against the XSD of the D-SI). Each category is described in more detail below.

3.1 Test for well-formedness

Many editors or integrated development environments (IDEs) that can be used to draft XML can automatically test that an XML file is well-formed. Such a test considers the following aspects:

- Does the file have a root element?
- Do all XML elements have a closing tag?
- Is the case sensitivity of XML tags respected?
- Are all XML elements properly nested?
- Are all XML attribute values quoted?

Example 1: The XML is well-formed.

```
<si:real>
  <si:label>temperature<\si:label>
  <si:value>20.10<\si:value>
  <si:unit>\degreecelsius<\si:unit>
  <si:expandedUnc>
    <si:uncertainty>0.50<\si:uncertainty>
    <si:coverageFactor>2<\si:coverageFactor>
    <si:coverageProbability>0.95<\si:coverageProbability>
    <si:distribution>normal<\si:distribution>
  <\si:expandedUnc>
<\si:real>
```

DOI: 10.5281/zenodo.3816696

Example 2: There is a mismatch between the opening and closing tags for the coverage probability element, and therefore the XML is not well-formed.

```
<si:real>
  <si:label>temperature<\si:label>
  <si:value>20.10<\si:value>
  <si:unit>\degreecelsius<\si:unit>
  <si:expandedUnc>
    <si:uncertainty>0.50<\si:uncertainty>
    <si:coverageFactor>2<\si:coverageFactor>
    <si:coverageProbability>0.95<\si:coverageProb>
    <si:distribution>normal<\si:distribution>
  <\si:expandedUnc>
<\si:real>
```

3.2 Validation against D-SI XML Schema

Validation of the metrological data against the associated XML Schema Definition (XSD) file considers the following aspects:

- Is the file containing the metrological data a valid XML file?
- Are all mandatory elements defined?
- Are elements presented in the correct order?
- Are correct element tags used?
- Are correct data types used?

Many integrated development environments (IDEs) that can be used to draft XML also allow for the validation of XML against an XSD file. Numerous free web-based XML validation services are also available.

Example 3: The mandatory element for the coverage factor has been omitted, and therefore the XML is not validated against the D-SI XML schema.

```
<si:real>
  <si:label>temperature<\si:label>
  <si:value>20.10<\si:value>
  <si:unit>\degreecelsius<\si:unit>
  <si:expandedUnc>
    <si:uncertainty>0.50<\si:uncertainty>
    <si:coverageProbability>0.95<\si:coverageProbability>
    <si:distribution>normal<\si:distribution>
  <\si:expandedUnc>
<\si:real>
```

It is recommended to ensure that XML is well formed and is validated against the D-SI XML schema before undertaking the further validation described in section 3.3.

3.3 Additional validation

The reference implementation of the D-SI data model for XML users [5] requires additional tests to be undertaken. These tests include, for example:

- The provision of SI identifiers for names of units.
- The provision of SI prefixes for units.
- The correct formatting of time stamps.
- Universal properties of uncertainty statements such as a symmetric covariance matrix.

For anyone wishing to store and exchange metrological data using the XML D-SI format, it is unreasonable to expect them to undertake this additional validation, for example, by developing bespoke software. Partners in the SmartCom project are therefore developing a system that will allow users to submit their data to an online server that undertakes, and informs them of the outcome of, the additional validation. This system will also be able to provide the validation of the well-formed XML and validation against the D-SI XML schema. The requirements for such a system are described in section 4.

4 Requirements for, and implementation of, an online validation system

4.1 Requirements

The requirements for the online validation system can be summarised as follows:

- Each application of the validation system must provide one unambiguous and correct result.

- A set of rules must be developed that list all tests that the validation system must undertake. The rules will relate to a specific edition of the D-SI booklet [2].

- Prior to its being made available to users, the validation system must be tested, using a suitable number of test XML files, to ensure confidence in the results it returns. The test XML files should be reflective of actual cases that arise within various metrology domains.

- Users must be provided with a user manual for the validation system that gives a clear description of the test aim and the test procedure.

- The validation system must allow users to submit their XML data in a simplistic manner, e.g., using a GUI to select an XML file and pressing a button to submit it to the system or allowing automated testing through a suitable API.

- The validation system must undertake validation and return to the user, in an appropriate format, information on the quality of the submitted XML file.

- There should be a plan for maintenance of the validation system. For example, if a new edition of the BIPM SI Brochure is released that allows for additional, or deprecates, units.

- The validation system must allow an unambiguous management of tests, e.g. by allowing registration of users on the system and subsequently log in to the system to request and trace instances of validation.

4.2 Implementation

The online validation system being developed by SmartCom partners leans heavily on technology developed within the European Metrology Research Programme (EMRP) JRP "Traceability for computationally-intensive metrology" (short name "TraCIM") [6]. Within the TraCIM JRP, an information and communications technology (ICT) infrastructure, referred to as the "TraCIM system" was developed that allows the verification of mathematical software to be undertaken online – see, for example, [7]. The TraCIM system requires users to submit and receive information in a similar way to that proposed for the online validation system. Therefore, it is appropriate to adopt the underlying software of the TraCIM system to satisfy the requirements of the online validation system.

The online validation system will classify submitted metrological data as belonging to one of the following quality classes:

- Platinum (or Next Generation).
- Gold.
- Silver.

DOI: 10.5281/zenodo.3816696

- Bronze.
- Improvable.

The criteria used to assess the quality are described in detail in [2]. A comprehensive set of example D-SI XML data to test the online validation system is available at [8].

5 References

[1] EMPIR project 17IND02 SmartCom, webpage: https://www.ptb.de/empir2018/smartcom (accessed May 2020).

[2] Hutzschenreuter D., Härtig F., Heeren W., et al.: SmartCom Digital System of Units (D-SI) Guide for the use of the metadata-format used in metrology for the easy-to-use, safe, harmonised and unambiguous digital transfer of metrological data – Second Edition, doi: 10.5281/zenodo.3816686 (accessed May 2020).

[3] World Wide Web Consortium *Extensible Markup Language (XML)*; version 1.0; fifth edition: https://www.w3.org/TR/xml/ (accessed May 2020).

[4] BIPM Brochure, *The International System of Units (SI) 9th edition 2019*, https://www.bipm.org/en/publications/si-brochure (accessed May 2020).

[5] Hutzschenreuter D., Härtig F., Wiedenhöfer T., et al.: SmartCom Digital-SI (D-SI) XML exchange format for metrological data version 1.3.1: doi: 10.5281/zenodo.3826517 (accessed May 2020).

[6] EMRP project NEW06 TraCIM, project webpage: https://www.ptb.de/emrp/tcim.html (accessed May 2020).

[7] PTB TraCIM system 1.0 service, webpage: https://tracim.ptb.de/tracim/index.jsf (accessed May 2020).

[8] GitLab repository of the D-SI XML reference implementation, webpage: https://gitlab1.ptb.de/d-ptb/d-si/xsd-d-si (accessed May 2020).

DOI: 10.5281/zenodo.3816696

DOI: 10.5281/zenodo.3816696

DOI: 10.5281/zenodo.3816696

The content presented was developed within the framework of the EU-funded project SmartCom *"Communication and validation of smart data in IoT-networks"* with the support of international partners from science and industry.

🏠

https://www.ptb.de/empir2018/smartcom
(accessed June 2020)

The EMPIR initiative is co-funded by the European Union's Horizon 2020
research and innovation programme and the EMPIR Participating States

DOI: 10.5281/zenodo.3816696

SmartCom

Good Practice Guides SmartCom Validation

(1) Test for communication interfaces used for the exchange of metrological data

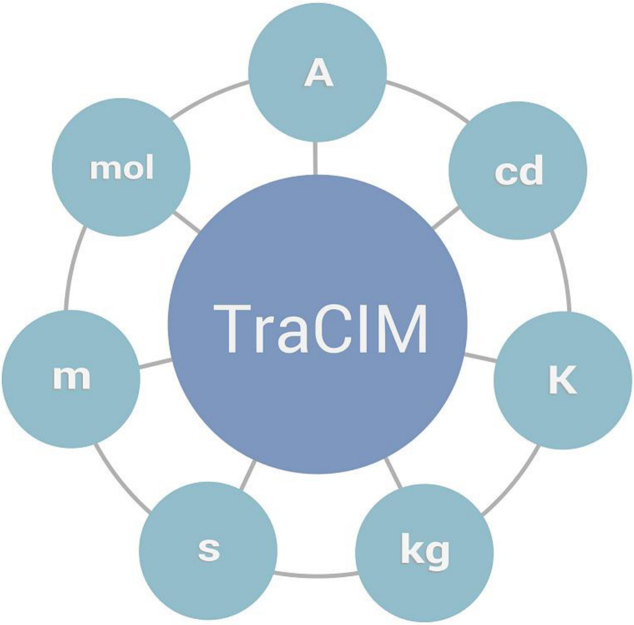
(2) Conformity test for unified DCCs

(3) TraCIM system

TraCIM D-SI

EN

DOI: 10.5281/zenodo.3816696



Good practice guide

Conformity test for unified DCCs

Version 1.0

DOI: 10.5281/zenodo.3816696

Editors

Physikalisch Technische Bundesanstalt, Germany:
D. Hutzschenreuter, S. Lin, S. Schönhals

National Physical Laboratory, United Kingdom:
I. Smith

Comprising the results from our research and the fruitful and intensive discussions with all our other project partners and stakeholders worldwide.

Contact: smartcom@ptb.de

Braunschweig June 2020

DOI: 10.5281/zenodo.3816696

Table of Contents

DOI: 10.5281/zenodo.3816696

1 Preliminary considerations

The SmartCom research project [1] collected and discussed minimum requirements for the content and use of unified Digital Calibration Certificates (DCCs) [2, 3]. The format for the storage and transmission of DCC data is not regulated and can be implemented in XML [4], JSON [5] or other well-established formats.

An XML implementation of the storage of DCC data was created based on the SmartCom project outcome [6]. It is currently hosted and maintained by the Physikalisch-Technische Bundesanstalt (PTB). This guide considers testing conformity against this XML structure for DCCs.

Testing the conformity of a DCC refers to a test of its XML syntax. It comprises testing the validity of the XML structure, the usage of all mandatory XML elements, and the compatibility of all information with the underlying data types. The conformity test is not intended to replace proper accreditation of the content of DCCs as it is typically undertaken for many paper-based calibration certificates through independent review. Furthermore, this test does not consider the conformity of information related to digital signatures and electronic seals to underlying cryptographic standards.

The tools for testing the conformity of the syntax of unified DCCs in XML format rely on third party software. It is possible to use these tools as a web-service or operate them locally.

2 General XML structure of DCCs in SmartCom

This guide considers version 2.3.0 of the DCC XML structure which was implemented according to the minimum requirements for DCCs from the SmartCom project [2, 3]. Documentation of this XML structure is available at [7]. In addition, the XML Schema Definition (XSD) of the DCC can be obtained from [8].

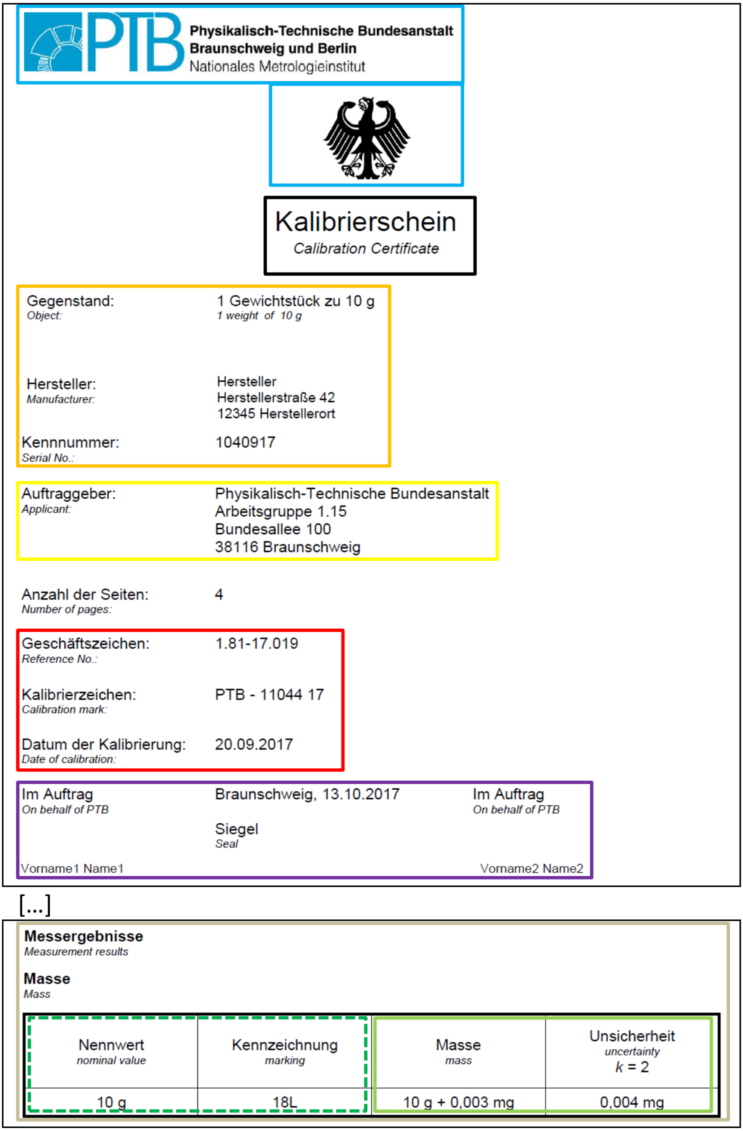
The XML structure of the DCC will be outlined for an extract from the calibration example at [6]. This example is based on the calibration of a weight of nominal mass 10 g that was performed at PTB. Figure 1 shows an extract from the human-readable document that is issued by PTB as a calibration certificate. The corresponding machine-readable version of this document in the form of a DCC in XML format is given in Figure 2.

To facilitate comparison between the human-readable and machine-readable documents, all data providing the same information is marked by rectangular boxes of the same colour in Figures 1 and 2, i.e., data from a box with green colour in Figure 1 provides the same information as the XML data in a green box in Figure 2.

All data not marked by a box is only required in the human-readable format and not relevant for the machine-readable representation, e.g., the number of pages in Figure 1.

Values of the mass measurement (in a green box) are provided in the D-SI metadata model [9]. XML elements of this metadata format start with the namespace prefix “si” in Figure 2. All other XML elements have the prefix “dcc”.

Finally, the XML element “respPersons” (from the purple box in Figure 2) shows who is responsible for signing the DCC and applying an electronic seal (digital seal). The electronic seal is applied to the XML DCC document as a whole and hence is not part of the structure in version 2.3.0.



Physikalisch-Technische Bundesanstalt
Braunschweig und Berlin
Nationales Metrologieinstitut

Kalibrierschein
Calibration Certificate

Gegenstand: *Object:*	1 Gewichtstück zu 10 g *1 weight of 10 g*
Hersteller: *Manufacturer:*	Hersteller Herstellerstraße 42 12345 Herstellerort
Kennnummer: *Serial No.:*	1040917
Auftraggeber: *Applicant:*	Physikalisch-Technische Bundesanstalt Arbeitsgruppe 1.15 Bundesallee 100 38116 Braunschweig
Anzahl der Seiten: *Number of pages:*	4
Geschäftszeichen: *Reference No.:*	1.81-17.019
Kalibrierzeichen: *Calibration mark:*	PTB - 11044 17
Datum der Kalibrierung: *Date of calibration:*	20.09.2017

Im Auftrag
On behalf of PTB

Braunschweig, 13.10.2017

Im Auftrag
On behalf of PTB

Siegel
Seal

Vorname1 Name1

Vorname2 Name2

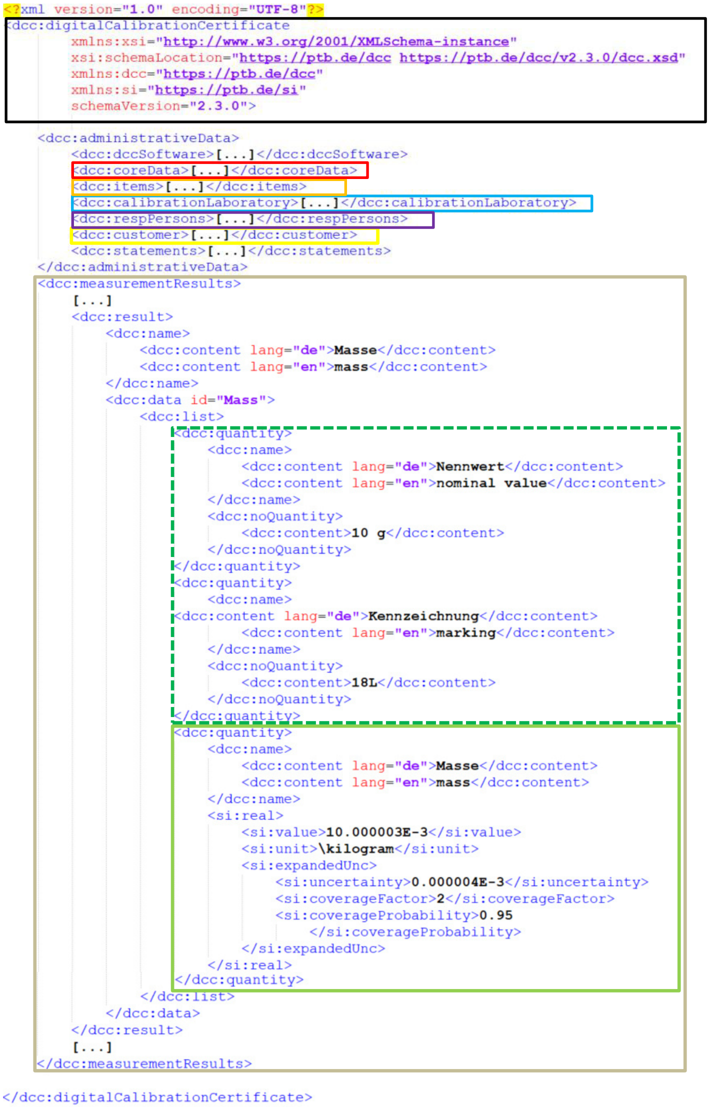
[...]

Messergebnisse
Measurement results

Masse
Mass

Nennwert *nominal value*	Kennzeichnung *marking*	Masse *mass*	Unsicherheit *uncertainty* *k* = 2
10 g	18L	10 g + 0,003 mg	0,004 mg

Figure 1: *Extract from a human-readable PTB certificate for a weight of nominal mass 10 g as it is typically issued at calibration.*



```
<?xml version="1.0" encoding="UTF-8"?>
<dcc:digitalCalibrationCertificate
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="https://ptb.de/dcc https://ptb.de/dcc/v2.3.0/dcc.xsd"
        xmlns:dcc="https://ptb.de/dcc"
        xmlns:si="https://ptb.de/si"
        schemaVersion="2.3.0">
    <dcc:administrativeData>
        <dcc:dccSoftware>[...]</dcc:dccSoftware>
        <dcc:coreData>[...]</dcc:coreData>
        <dcc:items>[...]</dcc:items>
        <dcc:calibrationLaboratory>[...]</dcc:calibrationLaboratory>
        <dcc:respPersons>[...]</dcc:respPersons>
        <dcc:customer>[...]</dcc:customer>
        <dcc:statements>[...]</dcc:statements>
    </dcc:administrativeData>
    <dcc:measurementResults>
        [...]
        <dcc:result>
            <dcc:name>
                <dcc:content lang="de">Masse</dcc:content>
                <dcc:content lang="en">mass</dcc:content>
            </dcc:name>
            <dcc:data id="Mass">
                <dcc:list>
                    <dcc:quantity>
                        <dcc:name>
                            <dcc:content lang="de">Nennwert</dcc:content>
                            <dcc:content lang="en">nominal value</dcc:content>
                        </dcc:name>
                        <dcc:noQuantity>
                            <dcc:content>10 g</dcc:content>
                        </dcc:noQuantity>
                    </dcc:quantity>
                    <dcc:quantity>
                        <dcc:name>
                    <dcc:content lang="de">Kennzeichnung</dcc:content>
                            <dcc:content lang="en">marking</dcc:content>
                        </dcc:name>
                        <dcc:noQuantity>
                            <dcc:content>18L</dcc:content>
                        </dcc:noQuantity>
                    </dcc:quantity>
                    <dcc:quantity>
                        <dcc:name>
                            <dcc:content lang="de">Masse</dcc:content>
                            <dcc:content lang="en">mass</dcc:content>
                        </dcc:name>
                        <si:real>
                            <si:value>10.000003E-3</si:value>
                            <si:unit>\kilogram</si:unit>
                            <si:expandedUnc>
                                <si:uncertainty>0.000004E-3</si:uncertainty>
                                <si:coverageFactor>2</si:coverageFactor>
                                <si:coverageProbability>0.95
                                    </si:coverageProbability>
                            </si:expandedUnc>
                        </si:real>
                    </dcc:quantity>
                </dcc:list>
            </dcc:data>
        </dcc:result>
        [...]
    </dcc:measurementResults>

</dcc:digitalCalibrationCertificate>
```

Figure 2: *Extract from the machine-readable XML DCC structure for the example calibration certificate given in Figure 1. Some elements are deliberately left empty.*

3 Testing the unified DCC syntax

3.1 Validation with DCC XML Scheme

Validation of the DCC against the underlying XSD file will test the following aspects of conformity with the DCC XML syntax:

- Valid XML file;
- All mandatory elements defined;
- Correct order of elements;
- Correct element tags (names);
- Correct data types used.

Many software development tools have plugins for validation of XML against an XSD file. Various free web-based services are also available.

3.2 Validation of D-SI Elements using TraCIM

The SmartCom TraCIM service is an additional tool for the validation of the correct usage of the D-SI metadata model in the XML DCC. Details are outlined in the first part of this series of guides. The SmartCom TraCIM service will be used to check all D-SI elements in a DCC for conformity to the following requirements:

- Compliance of the units used against SI units;
- Compliance with minimum required metrological information according to VIM [10] and GUM [11] as defined for the D-SI format;
- Compliance with D-SI XML Scheme Definition;
- All D-SI elements with correct tags, sequence of elements and data types.

It is recommended to test the DCC with the XSD before applying TraCIM testing.

4 References

[1] EMPIR project 17IND02 SmartCom, webpage: https://www.ptb.de/empir2018/smartcom (accessed May 2020).

[2] Wiedenhöfer T., Hutzschenreuter D., Smith I, Brown C.: Document describing a universal and flexible structure for digital calibration certificates (DCC), doi: 10.5281/ zenodo.3696567 (accessed May 2020).

[3] Nikander P., Elo T., Mustapää T., et al.: Document specifying rules for the secure use of DCC covering legal aspects of metrology, doi: 10.5281/zenodo.3664211, (accessed May 2020).

[4] World Wide Web Consortium *Extensible Markup Language (XML)*; version 1.0; fifth edition https://www.w3.org/TR/xml/ (accessed May 2020).

[5] Standard ECMA-404: The JSON Data Interchange Syntax, 2nd edition, December 2017.

[6] GitLab repository for the DCC XML implementation, operated by PTB, webpage: https://gitlab1.ptb.de/d-p tb/dcc/xsd-dcc (accessed May 2020).

[7] DCC XML Schema Definition (XSD) version 2.3.0, hosted by PTB, web page: https://www.ptb.de/dcc/v2.3.0/dcc.xsd (accessed May 2020).

[8] Online guide for using the DCC XML schema version 2.3.0 hosted by PTB, web page: https://www.ptb.de/dcc/v2.3.0/ (accessed May 2020).

[9] Hutzschenreuter D., Härtig F., Heeren W., et al.: SmartCom Digital System of Units (D-SI) Guide for the use of the metadata-format used in metrology for the easy-to-use, safe, harmonised and unambiguous digital transfer of metrological data – Second Edition, doi: 10.5281/zenodo.3816686 (accessed May 2020).

[10] JCGM 200:2012 *International Vocabulary of Metrology – Basic and general concepts and associated terms (VIM)*, 3rd edition, 2008 version with minor corrections, 2012.

[11] JCGM 100:2008 *Evaluation of Measurement Data – Guide to the Expression of uncertainty in Measurement*, GUM 1995 with minor corrections, first edition, September 2008.

DOI: 10.5281/zenodo.3816696

DOI: 10.5281/zenodo.3816696

The content presented was developed within the framework of the EU-funded project SmartCom *“Communication and validation of smart data in IoT-networks”* with the support of international partners from science and industry.

🏠

https://www.ptb.de/empir2018/smartcom
(accessed June 2020)

EMPIR EURAMET

The EMPIR initiative is co-funded by the European Union's Horizon 2020
research and innovation programme and the EMPIR Participating States

DOI: 10.5281/zenodo.3816696

SmartCom

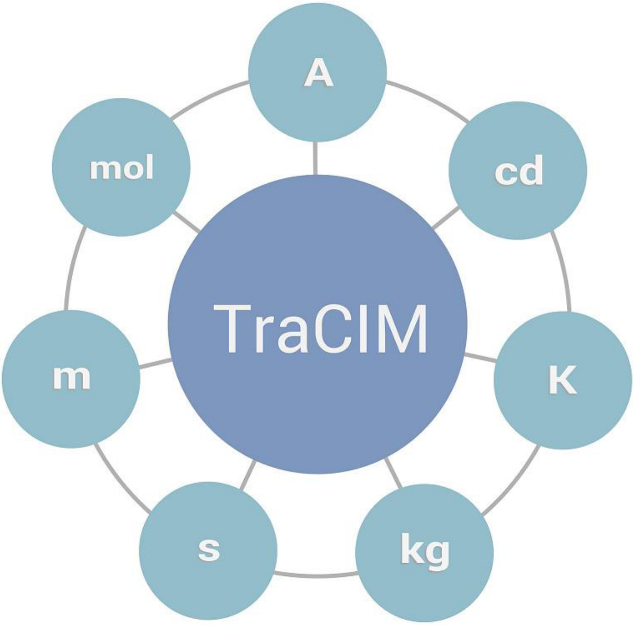
Good Practice Guides SmartCom Validation

(1) Test for communication interfaces used for the exchange of metrological data

(2) Conformity test for unified DCCs

(3) TraCIM system

TraCIM D-SI



EN

DOI: 10.5281/zenodo.3816696

Good practice guide

TraCIM System

Integration of the SmartCom online validation system into TraCIM

Version 1.0

DOI: 10.5281/zenodo.3816696

Editors

Ostfalia University of Applied Sciences, Germany:
L. Heindorf, B. Müller

Physikalisch Technische Bundesanstalt, Germany:
D. Hutzschenreuter, J. H. Loewe

Comprising the results from our research and the fruitful and intensive discussions with all our other project partners and stakeholders worldwide.

Contact: smartcom@ptb.de

Braunschweig June 2020

DOI: 10.5281/zenodo.3816696

Table of Contents

DOI: 10.5281/zenodo.3816696

1 Introduction

The SmartCom online validation system (SmartCom Expert Extension – or in short SE) was developed by Ostfalia University [1] in the scope of the European joint research project EMPIR 17IND02 SmartCom [2]. It establishes a service that can be integrated into a TraCIM validation system to test and certify XML documents for correct usage of metrological data based on the D-SI format, such as Digital Calibration Certificates [3,4]. It can be used with the TraCIM Server 1.0 [5,6,7] and the new TraCIM Server 2.0 (provided by Ostfalia University [1]).

Normal TraCIM Expert Extensions deliver test data that will be used by a system under test to calculate a result (e.g. [8]). Then this result will be validated by the specific Expert Extension.

The validation process of the SE slightly differs from this normal process. The SE validates any well-formed XML document and therefore doesn't require test data and test data generation. Nevertheless, the corresponding REST API of the TraCIM server must be called in order to get an individual test process key for using the validation service. However, the SE doesn't generate test data.

DOI: 10.5281/zenodo.3816696

2 Integration of SmartCom online validation into TraCIM

The SmartCom online validation system (SE) is provided as a jar module for the TraCIM server application. This section explains how to use Maven [9] to build the module and run it with the TraCIM system.

This version was developed, tested and prepared to use with the new TraCIM server 2.0 and the corresponding Expert Extension API. Parts of the validation methods for the SE are provided through the SiChecker API. In the scope of the SmartCom project this API was implemented by PTB.

Because the TraCIM server as well as the SE come with their own installation instructions these instructions here serve the purpose to give an overview. Eventually, this document will not be updated on a regular basis. In doubt, please take a look into the TraCIM and SmartCom documentation with your used versions.

2.1 Set up a TraCIM service

In order to use the SE, you need a running TraCIM service. Refer to [1,7] to obtain and set up a TraCIM service.

The `jboss-deployment-structure.xml` of the TraCIM server must be extended with the following entry.

```
  <module name="deployment.smartcom-expert-
extension.jar" services="import" meta-
inf="import" optional="true" />
```

2.2 Build the SmartCom Expert Extension

The SE uses the PTB SiChecker for validation. Before building the SE, you have to install the corresponding dependency to your local repository.

```
  mvn install:install-file -Dfile=SiChecker-
1.0.jar -DgroupId=de.ptb.si.tools -
DartifactId=si-checker -Dversion=1.0 -
Dpackaging=jar
```

If you choose to use the integrated `SiChecker` you can remove the PTB SiChecker dependency from the Project Object Model (POM) and can continue without the PTB SiChecker.

Build the project.

```
  mvn clean package
```

2.3 Deploy the SmartCom Expert Extension

The build generates a jar with and without dependencies. You are free to use either of those. If you use the jar without dependencies, you must deploy all dependencies separately and edit the `jboss-deployment-structure.xml` accordingly.

The simple and recommended way is to use the jar with dependencies.

Use your preferred method for deployment, for example the deployment scanner:

```
  cp target/smartcom-expert-extension.jar
<WILDFLY_HOME>/standalone/deployments
```

or

```
  cp target/smartcom-expert-extension-jar-
with-dependencies.jar
<WILDFLY_HOME>/standalone/deployments/smartc
om-expert-extension.jar
```

(Renaming the jar with dependencies is not necessarily required as long as the file name is the same as the file name in the `jboss-deployment-structure.xml` of the TraCIM server.)

If you have deployed the SE while the TraCIM server was running, restart Wildfly or deploy the TraCIM server again.

2.4 Run the Integration Tests

We are using the REST Interface of the TraCIM server for integration testing. Therefore, a TraCIM server with the SE under test must be running on localhost and port 8080. The Maven command to run the integration test is

```
mvn clean test failsafe:integration-test
```

2.5 Run the OWASP Dependency Check

The OWASP Dependency Check is used by our POM. Please make sure to use the most recent version. To run the OWASP dependency check, one of the following two commands can be used:

```
mvn verify
```

or

```
mvn dependency-check:check
```

2.6 Customization

2.6.1 PDF reports

Each SmartCom test provides a report of the test results in PDF format. It may be required to change the output PDF document for various reasons such as to use company specific logos, comply with different reporting policies, or incorporate updates to the validation scope.

The package `de.ostfalia.smartcom.pdf` in the SE contains the Java source files that generate a PTB-specific test report. These files can be changed to modify the report. The open source Java tool Apache PDFBox [10] is used to create the PDF documents.

It is also possible to integrate an alternative PDF creation library. In this case, the PDF generation must properly be linked and run in the `generateValidationResult` method in the main class `SmartComExtension.java` in package `de.ostfalia.smartcom` of the SE.

2.6.2 D-SI XML Schema changes

The D-SI XML schema (the path of the schema) that should be used for validating the SI elements is defined in the SmartComValidator class of the SE. In order to validate against a new schema, the schema must be added as a resource to the project and the path to the schema in the SmartComValidator class must be modified accordingly.

The package `de.ostfalia.smartcom.si` contains the Java representation for the elements defined by the schema. If the schema changes, make sure to update the Java objects accordingly. When adding new types, make sure to explicitly declare the name in `@XmlRootElement` annotation. Additionally, they must be added to the `jaxb.index` file in folder `resources/de/ostfalia/smartcom/si`. If the namespace changes (prefix or URI), update the namespace in `package-info.java`.

No further changes are required. Namespace and tag names are automatically determined by reflection from those annotations. The version information of the schema (i.e. as printed in pdf) is automatically determined from the version tag inside the schema.

After modifying the elements, it might be necessary to build and distribute the new API for the adaption of the external SiChecker (see the following subsection).

2.6.3 Build the SiChecker API

For the development of external implementations of the SiChecker (i.e. PTB SiChecker), the API is needed. Currently the API is included in this project. In future the API might be extracted into a separate project.

In order to extract and build the API, one can use the script `build_api.sh` from the SmartCom Expert Extension Maven project.

As described above, the validation of a single SI element is performed by an implementation of the interface `SiChecker`. The used implementation can be changed in the `SmartComValidator` class. As default, the `PTBSiChecker` implementation is used. Additionally, the project contains a second implementation called `OstfaliaSiChecker` which can be used as an alternative.

Caution: The `OstfaliaSiChecker` checks only the units of real elements based on a regular expression as defined in the D-SI brochure [11]. Additional checks of the latest regulations [3] are not implemented.

2.6.4 Other properties

When the SE is used with the TraCIM system 2.0, additional information can be customized that will be displayed by the TraCIM server to users of the SmartCom validation. The relevant information is in the project folder `src/main/resources/META-INF`. The file `webshop.properties` contains general web shop properties and `resources/service.xhtml` allows the creation of a web page that includes a detailed description of the service.

DOI: 10.5281/zenodo.3816696

2.7 Other TraCIM Systems

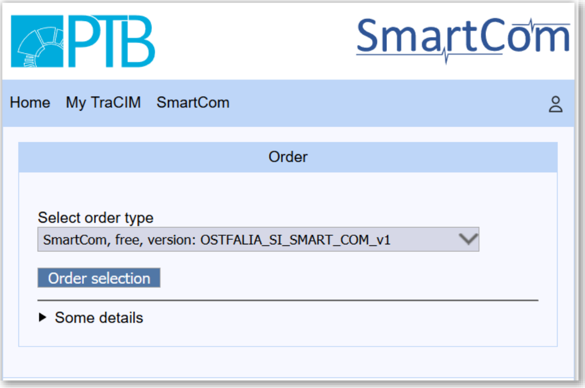
In order to use the SE with the old TraCIM server 1.0 (e.g. [6,7]) you have to change the version of the Expert Extension dependency in the POM.

Furthermore, the TraCIM server is used with PostgreSQL [12] and WildFly [13]. However, it should be possible to port to any RDBMS and Java EE application server. From WildFly, the `jboss-deployment-structure.xml` is used to tweak class loading.

3 Examples for the integration in TraCIM

This section gives a short outline on the integration of the SE into a TraCIM system 2.0 that was run as example by PTB within the SmartCom project [2].

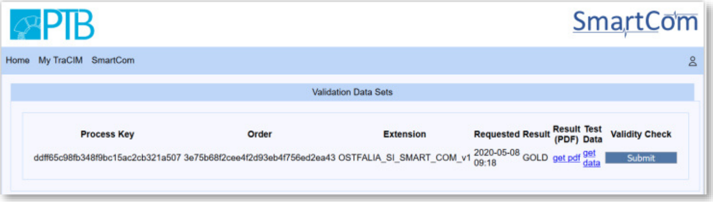
After registration and login, users can order the SmartCom validation in the order menu within “My TraCIM” as shown in Figure 1. There is the option to choose between a free test and a paid test in this example.



Select order type
SmartCom, free, version: OSTFALIA_SI_SMART_COM_v1

Figure 1: *Example of a SmartCom validation service selection in the TraCIM system 2.0.*

The user can also utilize the TraCIM web page to send D-SI XML data for validation and obtain the result by downloading the PDF result certificate (see Figure 2).



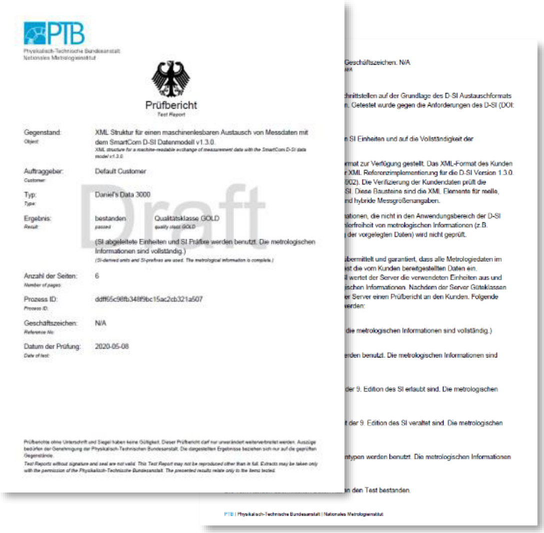
Validation Data Sets

Process Key	Order	Extension	Requested	Result	Result (PDF)	Test Data	Validity Check
ddff65c98fb348f9bc15ac2cb321a507	3e75b68f2cee4f2d93eb4f756ed2ea43	OSTFALIA_SI_SMART_COM_v1	2020-05-08 09:18	GOLD	get pdf	get data	Submit

Figure 2: *Example of information on SmartCom tests that is made available by the TraCIM system 2.0.*

It is not necessary to build a TraCIM client as it was the case with the TraCIM version 1.0.

Figure 3 shows an example of the first page of the PDF report that is formatted in compliance with the PTB policy for test report.


PTB
Physikalisch-Technische Bundesanstalt
Nationales Metrologieinstitut
Draft

Prüfbericht
Test Report

Gegenstand: *Object*	XML Struktur für einen maschinenlesbaren Austausch von Messdaten mit dem SmartCom D-SI Datenmodell v1.3.0. *XML structure for a machine-readable exchange of measurement data with the SmartCom D-SI data model v1.3.0*
Auftraggeber: *Customer*	Default Customer
Typ: *Type*	Daniel's Data 3000
Ergebnis: *Result*	bestanden *passed* Qualitätsklasse GOLD *quality class GOLD* (SI abgeleitete Einheiten und SI Präfixe werden benutzt. Die metrologischen Informationen sind vollständig.) *(SI-derived units and SI-prefixes are used. The metrological information is complete.)*
Anzahl der Seiten: *Number of pages*	6
Prozess ID: *Process ID*	ddff65c98fb348f9bc15ac2cb321a507
Geschäftszeichen: *Reference No*	N/A
Datum der Prüfung: *Date of test*	2020-05-08

Prüfberichte ohne Unterschrift und Siegel haben keine Gültigkeit. Dieser Prüfbericht darf nur unverändert weiterverbreitet werden. Auszüge bedürfen der Genehmigung der Physikalisch-Technischen Bundesanstalt. Die dargestellten Ergebnisse beziehen sich nur auf die geprüften Gegenstände.

Test Reports without signature and seal are not valid. This Test Report may not be reproduced other than in full. Extracts may be taken only with the permission of the Physikalisch-Technische Bundesanstalt. The presented results relate only to the items tested.

Geschäftszeichen: N/A
N/A

chnittstellen auf der Grundlage des D-SI Austauschformats
n. Getestet wurde gegen die Anforderungen des D-SI (DOI:

n SI Einheiten und auf die Vollständigkeit der

rmat zur Verfügung gestellt. Das XML-Format des Kunden
r XML Referenzimplementierung für die D-SI Version 1.3.0.
902). Die Verifizierung der Kundendaten prüft die
SI. Diese Bausteine sind die XML Elemente für reelle,
nd hybride Messgrößenangaben.

ationen, die nicht in den Anwendungsbereich der D-SI
hlerfreiheit von metrologischen Informationen (z.B.
der vorgelegten Daten) wird nicht geprüft.

übermittelt und garantiert, dass alle Metrologiedaten im
est die vom Kunden bereitgestellten Daten ein.
I wertet der Server die verwendeten Einheiten aus und
ischen Informationen. Nachdem der Server Güteklassen
er Server einen Prüfbericht an den Kunden. Folgende
werden:

die metrologischen Informationen sind vollständig.)

erden benutzt. Die metrologischen Informationen sind

der 9. Edition des SI erlaubt sind. Die metrologischen

t der 9. Edition des SI veraltet sind. Die metrologischen

ntypen werden benutzt. Die metrologischen Informationen

n den Test bestanden.

PTB | Physikalisch-Technische Bundesanstalt | Nationales Metrologieinstitut

Figure 3: *Example of the first page of a PTB Test Report for a free SmartCom test of XML data with D-SI elements.*

4 References

[1] Ostfalia University of Applied Science, webpage: https://www.ostfalia.de/cms/en/ (accessed May 2020).

[2] EMPIR project 17IND02 SmartCom, webpage: https://www.ptb.de/empir2018/smartcom (accessed May 2020).

[3] Hutzschenreuter D., Härtig F., Heeren W., et al.: SmartCom Digital System of Units (D-SI) Guide for the use of the metadata-format used in metrology for the easy-to-use, safe, harmonised and unambiguous digital transfer of metrological data – Second Edition, doi: 10.5281/zenodo.3816686 (accessed May 2020).

[4] Wiedenhöfer T., Hutzschenreuter D., Smith I, Brown C.: Document describing a universal and flexible structure for digital calibration certificates (DCC), doi: 10.5281/ zenodo.3696567 (accessed May 2020).

[5] EMRP project NEW06 TraCIM, project webpage: https://www.ptb.de/emrp/tcim.html (accessed May 2020).

[6] NPL TraCIM system 1.0 service, webpage: https://tracim.npl.co.uk/tracim/index.jsf (accessed May 2020).

[7] PTB TraCIM system 1.0 service, webpage: https://tracim.ptb.de/tracim/index.jsf (accessed May 2020).

[8] Wendt K., Brand U., Lunze U., Hutzschenreuter D.: Traceability for computationally intensive metrology user manual for Chebyshev algorithm testing, TraCIM e. V.,

https://tracim.ptb.de/tracim/resources/downloads/ptbwhz_math_chebyshev/ptbwhz_math_chebyshev_ma nual.pdf (accessed May 2020).

[9] Apache Maven project, web page: http://maven.apache.org/ (accessed May 2020).

[10] Apache PDFBox project, web page: https://pdfbox.apache.org/ (accessed May 2020).

[11] Hutzschenreuter D., Härtig F., Heeren W. et al.: SmartCom Digital System of Units (D-SI) Guide for the use of the metadata-format used in metrology for the easy-to-use, safe, harmonised and unambiguous digital transfer of metrological data, doi: 10.5281/zenodo.3522631 (accessed May 2020).

[12] PostgreSQL object-relational database system, web page: https://www.postgresql.org/ (accessed May 2020).

[13] Wildfly web application server, web page: https://wildfly.org/ (accessed May 2020).

DOI: 10.5281/zenodo.3816696

DOI: 10.5281/zenodo.3816696

DOI: 10.5281/zenodo.3816696

The content presented was developed within the framework of the EU-funded project SmartCom *“Communication and validation of smart data in IoT-networks*” with the support of international partners from science and industry.

🏠

https://www.ptb.de/empir2018/smartcom
(accessed June 2020)

EMPIR EURAMET

The EMPIR initiative is co-funded by the European Union's Horizon 2020
research and innovation programme and the EMPIR Participating States

DOI: 10.5281/zenodo.3816696