

New dynamics needed:
Standardisation and
Open source in a
successful ecosystem
of open collaboration

Overview

1 Open source
challenging
standardisation

2 The interplay of
standardisation
and open source

3 Regulating
software means
regulating open
source

4 The urgency for
a standardisation
ecosystem meeting
the needs of IT

Challenging

Several things that used to be done in standards bodies ...

... have increasingly been done in open source projects

Open Source has been highly disruptive to traditional standardisation

Open Source is running code – fast way from innovative development to deployment

Standards bodies are looking for ways how to include Open Source into their scope and work

A Standard ...

AI Management System ISO 42001

814 Responsibilities of the various roles should be defined to the level appropriate for the individual(s) to
815 perform their duties.

816 A.3.3 Reporting system

817 Control

818 The organization should define and put in place a process for reporting concerns about the
819 organization's role with respect to an AI system throughout its life cycle.

820 Implementation guidance

821 The reporting system should be designed so that it fulfils the following functions:

- 822 a) options for confidentiality or anonymity or both;
- 823 b) available and promoted to employed and contracted persons and others;
- 824 c) staffed with qualified persons;
- 825 d) stipulates appropriate investigation and resolution powers for the persons referred to in list
826 item b);
- 827 e) provides for mechanisms to report and to escalate to top management in a timely manner;
- 828 f) provides for effective protection from reprisals for both the persons concerned with reporting
829 and investigation (e.g. by allowing reports to be made anonymously and confidentially);
- 830 g) provides reports in accordance with Clause 7.4 and 7.5 and, if appropriate, item e);
- 831 h) provides feedback via an obligation to respond and follow up to the discloser within an
832 appropriate time frame.

833 In addition to the implementation guidance provided in this clause organizations should further
834 consider ISO 37002:2021 [5].

835 NOTE ISO 37002:2021 provides guidance on the use of reporting system reports specified in item g).

836 A.4 Resources for AI

837 A.4.1 General

838 Objective

839 To ensure that the organization accounts for the resources (including components and assets) of the AI
840 system in order to fully understand and address risks and impacts.

841 A.4.2 Resource documentation

842 Control

843 The organization should identify and document all relevant resources required for the AI system life
844 cycle stages and other AI-related activities relevant for the organization.

... also a Standard ...

3.3.5 Does *response* match *metadataList*?

1. Let *parsedMetadata* be the result of [parsing *metadataList*](#).
2. If *parsedMetadata* is **no metadata**, return **true**.
3. If [response is not eligible for integrity validation](#), return **false**.
4. If *parsedMetadata* is the empty set, return **true**.
5. Let *metadata* be the result of [getting the strongest metadata from *parsedMetadata*](#).
6. For each *item* in *metadata*:
 1. Let *algorithm* be the *alg* component of *item*.
 2. Let *expectedValue* be the *val* component of *item*.
 3. Let *actualValue* be the result of [applying *algorithm* to *response*](#).
 4. If *actualValue* is a case-sensitive match for *expectedValue*, return **true**.
7. Return **false**.

This algorithm allows the user agent to accept multiple, valid strong hash functions. For example, a developer might write a **script** element such as:

EXAMPLE 7

```
<script src="https://example.com/example-framework.js"
  integrity="sha384-Li9vy3DqF8tnTXuiaAJuML3ky+er10rcgNR/VqsVpcw+ThHmYcwiB1pb0xEI
  sha384-+/M6kredJcxdsqkczBUjMLvqyHb1K/JThDXwsBVxMEEzHEaMKEOect339VI1
  crossorigin="anonymous"></script>
```

which would allow the user agent to accept two different content payloads, one of which matches the first SHA384 hash value and the other matches the second SHA384 hash value.

Subresource Integrity

W3C Recommendation 23 June 2016

This version:

<http://www.w3.org/TR/2016/REC-SRI-20160623/>

Latest published version:

<http://www.w3.org/TR/SRI/>

... and also a Standard

DP-3T

Available on
<https://github.com/DP-3T>

[dp3t-sdk-backend/](#)
[dpppt-backend-sdk/](#)
[dpppt-backend-sdk-data/pom.xml](#)

67 lines (61 sloc) | 1.9 KB

```
1 <!--
2 ~ Copyright (c) 2020 Ubique Innovation AG <https://www.ubique.ch>
3 ~
4 ~ This Source Code Form is subject to the terms of the Mozilla Public
5 ~ License, v. 2.0. If a copy of the MPL was not distributed with this
6 ~ file, You can obtain one at https://mozilla.org/MPL/2.0/.
7 ~
8 ~ SPDX-License-Identifier: MPL-2.0
9 -->
10
11 <project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
12         xmlns="http://maven.apache.org/POM/4.0.0"
13         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
14     <modelVersion>4.0.0</modelVersion>
15     <parent>
16         <groupId>org.dpppt</groupId>
17         <artifactId>dpppt-backend-sdk</artifactId>
18         <version>1.0.0-SNAPSHOT</version>
19     </parent>
20     <artifactId>dpppt-backend-sdk-data</artifactId>
21     <name>DP3T Backend SDK Data</name>
22
23     <properties>
24         <org.testcontainers.version>1.14.2</org.testcontainers.version>
25         <sonar.projectKey>DP-3T_dp3t-sdk-data</sonar.projectKey>
26     </properties>
27
28     <dependencies>
29
30         <!-- dp3t models -->
31         <dependency>
32             <groupId>org.dpppt</groupId>
33             <artifactId>dpppt-backend-sdk-model</artifactId>
34             <version>1.0.0-SNAPSHOT</version>
35         </dependency>
36
```

The different types of a standard...

814 Responsibilities of the various roles should be defined to the level appropriate for the individual(s) to
815 perform their duties.

816 A.3.3 Reporting system

817 Control

818 The organization should define and put in place a process for reporting
819 organization's role with respect to an AI system throughout its life cycle.

820 Implementation guidance

821 The reporting system should be designed so that it fulfils the following functions:

- 822 a) options for confidentiality or anonymity or both;
- 823 b) available and promoted to employed and contracted persons and other persons causing or contributing to the risk;
- 824 c) staffed with qualified persons;
- 825 d) stipulates appropriate investigation and resolution powers for the reporting system;
- 826 e) provides for mechanisms to report and to escalate to top management in a timely manner;
- 827 f) provides for effective protection from reprisals for both the reporting persons causing or contributing to the risk and investigation (e.g. by allowing reports to be made anonymously and confidentially);
- 828 g) provides reports in accordance with Clause 7.4 and 7.5 and, if appropriate, to the relevant regulatory authorities;
- 829 h) provides feedback via an obligation to respond and follow up to the reporting person within an appropriate time frame.

833 In addition to the implementation guidance provided in this clause, organizations should further
834 consider ISO 37002:2021 [5].

835 NOTE ISO 37002:2021 provides guidance on the use of reporting systems.

836 A.4 Resources for AI

837 A.4.1 General

838 Objective

839 To ensure that the organization accounts for the resources (including
840 system in order to fully understand and address risks and impacts.

841 A.4.2 Resource documentation

842 Control

843 The organization should identify and document all relevant resources, requirements, and
844 cycle stages and other AI-related activities relevant for the organization.

3.3.5 Does response match metadataList?

1. Let *parsedMetadata* be the result of [parsing metadataList](#).
2. If *parsedMetadata* is **no metadata**, return **true**.
3. If *response* is **not eligible for integrity validation**, return **false**.
4. If *parsedMetadata* is the empty set, return **true**.
5. Let *metadata* be the result of [getting the strongest metadata from parsedMetadata](#).
6. For each *item* in *metadata*:
 1. Let *algorithm* be the *alg* component of *item*.
 2. Let *expectedValue* be the *val* component of *item*.
 3. Let *actualValue* be the result of [applying algorithm to response](#).
 4. If *actualValue* is a case-sensitive match for *expectedValue*, return **true**.
7. Return **false**.

This algorithm allows the user agent to accept multiple, valid strong hash functions. For example, a developer might write a `script` element such as:

EXAMPLE 7

```
<script src="https://example.com/example-framework.js"
  integrity="sha384-Li9vy3DqF8tnTXuiaAJuML3ky+erl0rcgNR/VqsVpcw+ThmYcwiB1pb0Xe
  sha384-+/M6kredJcxdsqkczBUjMLvqyHb1K/JThDXwBVxMEeZHEaMKE0Ect339VI
  crossorigin="anonymous"></script>
```

which would allow the user agent to accept two different content payloads, one of which matches the first SHA384 hash value and the other matches the second SHA384 hash value.

67 lines (61 sloc) | 1.9 KB

```
1 <!--
2 ~ Copyright (c) 2020 Ubiq Innovation AG <https://www.ubique.ch>
3 ~
4 ~ This Source Code Form is subject to the terms of the Mozilla Public
5 ~ License, v. 2.0. If a copy of the MPL was not distributed with this
6 ~ file, You can obtain one at https://mozilla.org/MPL/2.0/.
7 ~
8 ~ SPDX-License-Identifier: MPL-2.0
9 -->
10
11 <project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
12   xmlns="http://maven.apache.org/POM/4.0.0"
13   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
14   <modelVersion>4.0.0</modelVersion>
15   <parent>
16     <groupId>org.dpppt</groupId>
17     <artifactId>dpppt-backend-sdk</artifactId>
18     <version>1.0.0-SNAPSHOT</version>
19   </parent>
20   <artifactId>dpppt-backend-sdk-data</artifactId>
21   <name>DP3T Backend SDK Data</name>
22
23   <properties>
24     <org.testcontainers.version>1.14.2</org.testcontainers.version>
25     <sonar.projectKey>DP-3T_dp3t-sdk-data</sonar.projectKey>
26   </properties>
27
28   <dependencies>
29
30     <!-- dp3t models -->
31     <dependency>
32       <groupId>org.dpppt</groupId>
33       <artifactId>dpppt-backend-sdk-model</artifactId>
34       <version>1.0.0-SNAPSHOT</version>
35     </dependency>
36
```

The form(at) of a standard can be very different

Open Source Software can be a standard – often APIs, protocols, etc.

Agile standardisation taking place collaboratively, e.g. on github

Interplay of Standardisation and Open Source (1)

Standards are developed in open source

Open Source software (code) is the standard
May be part or entire standard
Available for fast implementation
E.g. APIs, interfaces, protocols

Interplay of Standardisation and Open Source (2)

Standards are **implemented** in open source

Standards are **maintained** in open source

Reference implementations, test implementations

Promulgation of standards

Dynamic relation between open source implementation and standard

Release cycles, fast feedback cycles, stable code

Interplay of Standardisation and Open Source (3)

Open Source implements standards

Like any other software open source software may implement standards

The standards need to be implementable in open source (IPRs; “restriction free”)

Important in areas like software interoperability

Interplay of Standardisation and Open Source (4)

**Open Source
complements
standards**

Non-normative parts
around standards

Platforms
E.g. service layer,
service delivery


Tools
E.g. test environment,
implementation
environment

AI Act, Cyber Resilience Act, Data Act:
Extension of EU safety objectives into the virtual world



So far: Safety of products
for the single market

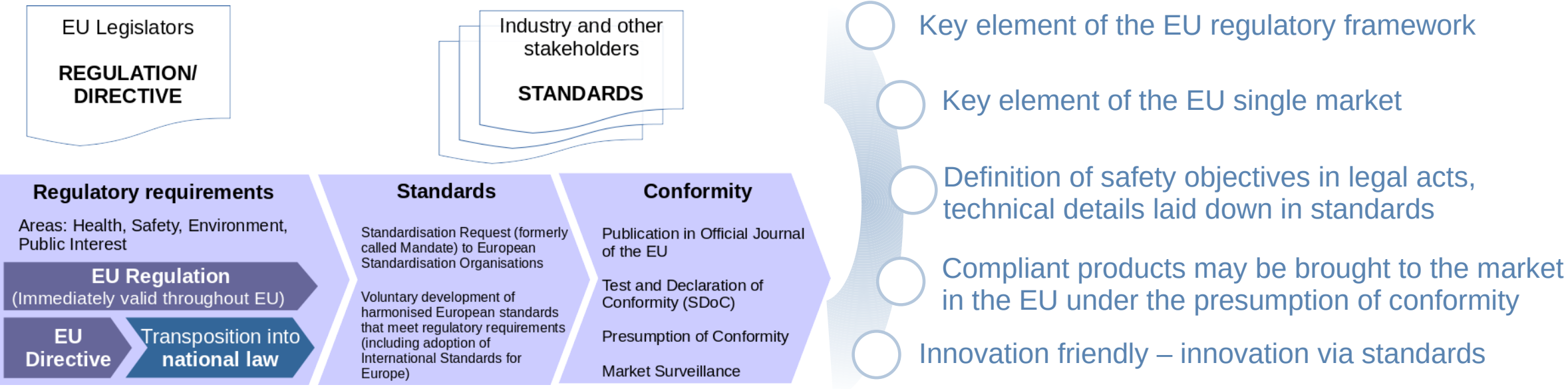
The image shows a close-up of a black power supply unit. It features several certification labels, including CE, FCC, and a 'SAFETY MARK' label with the number 072016-11. The unit is labeled 'POWER SUPPLY' and 'Output: 9.0V(9.0V) = 500mA'. It also includes the text 'Conforms to UL Standard 60950-1', 'Listed to UL Standard 2 No. 60950-1', '3091402', 'N136', 'R3514', 'Plantronics Japan Ltd', 'TUV Rheinland America S.A.', 'CAUTION: ITE use only For indoor use only For Plantronics product only Made in China by SIL', and '모델명: SSA-5W 090050' and '최적소비효율기준 만족 제품'.



In addition now: Safety for the
interaction in the virtual world

The image shows a person wearing a VR headset and gesturing with their hand. The person is smiling and appears to be engaged in a virtual environment. The background is a white wall with a blue and white pixelated pattern.

Success Story for Europe: New Approach – New Legislative Framework (NLF)

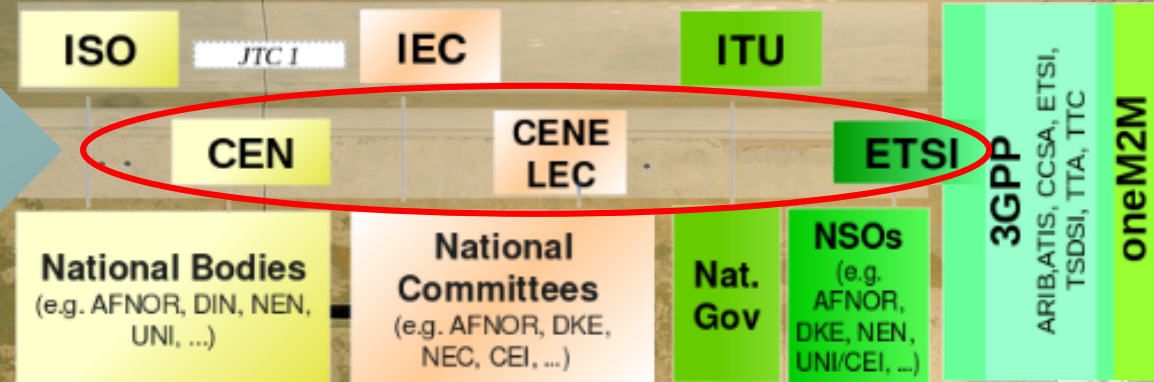


Highly successful instrument for EU technical regulation for decades.

NLF processes have now been applied for the new regulations, the AI Act, the Cyber Resilience Act, the Data Act

But the ESS is not “fit-for-purpose” for IT Standardisation

European Standardisation Organisations (ESO)
Development of Harmonised European Standards



Structures and processes don't meet the needs of the IT sector

Distance between the actual stakeholders and the decision makers in the organisations.

Lack of agility and direct open collaboration – in particular with open source communities.

Different structures and processes are needed for having an attractive and effective ecosystem in Europe for the development of IT standards that can be globally influential.

Do we need a fourth ESO?

European standardisation needs different and new processes for supporting a successful environment for IT standardisation.

CEN-CENELEC should play a key role in developing new structures and processes.

Europe should have the ambition of becoming the place where the IT sector innovates and standardises.

4 KEY POINTS

1 Link to international standardisation (ISO/IEC JTC 1)

IT standardisation is global – a close link to ISO/IEC JTC 1 is of key relevance.

2 Direct participation

IT is dynamic, agile and direct – this includes SMEs and civil society

3 Free availability of standards “by mouse-click”

Free availability is a key principle of the IT sector.

Open source communities need to implement the standards which makes free availability a must.

4 Implementability in Open Source

IPR Policies need respective option(s) that allow for open source to work and implement the standards, e.g. Royalty-free, non-assert clause.

Thanks very much for your attention

Dr. Jochen Friedrich

jochen@de.ibm.com

<https://www.linkedin.com/in/jochenfriedrich/>