ETSI TS 103 779 V0.0.4 (2022-02)

SmartM2M; Requirements and Guidelines for cross-domain data usability of IoT devices

Reference
DTS/SmartM2M-103779

Keywords
artificial intelligence, IoT, oneM2M, data usability, use case

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

Introduction

1 Scope

The present document has the objective:

• to define minimum requirements for data and services usability on professional and general public IoT devices and platforms, whether they are critical or not;

• to develop a horizontal cross-domain specification encompassing these requirements.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI TS 103 264: "SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping".

[2] ETSI EN 303 645: "CYBER; Cyber Security for Consumer Internet of Things: Baseline Requirements".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI TR 103 778:"SmartM2M; Use cases for cross-domain data usability of IoT devices".

[i.2] E Goldstein, U Gasser, and B Budish: “Data Commons Version 1.0: A Framework to Build Toward AI for Good”, 2018.

NOTE: Available at: https://medium.com/berkman-klein-center/data-commons-version-1-0-a-framework-to-build-toward-ai-for-good-73414d7e72be [Accessed 15 November 2021].

[i.3] 3GPP TS 22.891 V14.2.0: “Technical Specification Group Services and System Aspects; Feasibility Study on New Services and Markets Technology Enablers”, September 2016.

[i.4] 3GPP R1-162204: “Numerology requirements”, April 2016.

[i.5] M Chen, Y Miao, Y Hao, and K Hwang: “Narrow band internet of things”, IEEE Access, vol. 5, pp. 20557–20577, 2017.

[i.6] Z He, “Automatic cooking system”, US Patent App. 16/155,895, Feb. 2019

[i.7] F Adelantado, X Vilajosana, P Tuset-Peiro, B Martinez, J Melia-Segui, and T Watteyne: "Understanding the limits of lorawan", IEEE Communications Magazine, vol. 55, pp. 34–40, Sep. 2017.

[i.8] C Yi, J Cai, and Z Su: "A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications", IEEE Transactions on Mobile Computing, 2019.

[i.9] A Pal and K Kant: "Nfmi: Connectivity for short-range iot applications", Computer, vol. 52, pp. 63–67, Feb 2019.

[i.10] M Merry: "Environmental problems that batteries cause", Sciencing, Mar 2019.

[i.11] A Froytlog, T Foss, O Bakker, G Jevne, M A Haglund, F Y Li, J Oller, and G Y Li: "Ultra-low power wake-up radio for 5g iot", IEEE Communications Magazine, vol. 57, no. 3, pp. 111–117, 2019.

[i.12] Z Qin, F Y Li, G Y Li, J A McCann, and Q Ni: "Low-power wide-area networks for sustainable iot", IEEE Wireless Communications, 2019.

[i.13] B Safaei, A M H Monazzah, M B Bafroei, and A Ejlali: "Reliability side-effects in internet of things application layer protocols", in 2017 2nd International Conference on System Reliability and Safety (ICSRS), pp. 207–212, IEEE, 2017.

[i.14] N A Mohammed, A M Mansoor, and R B Ahmad: "Mission-critical machine-type communications: An overview and perspectives towards 5g", IEEE Access, 2019.

[i.15] M B Mollah, S Zeadally, and M A K Azad: "Emerging wireless technologies for internet of things applications: Opportunities and challenges", 2019.

[i.16] J Wu and P Fan: "A survey on high mobility wireless communications: Challenges, opportunities and solutions", IEEE Access, vol. 4, pp. 450-476, 2016.

[i.17] M Ryu, J Yun, T Miao, I-Y Ahn, S-C Choi, and J Kim: "Design and implementation of a connected farm for smart farming system", in 2015 IEEE SENSORS, pp. 1–4, IEEE, 2015.

[i.18] L F Ochoa, G P Harrison: "Minimizing energy losses: optimal accommodation and smart operation of renewable distributed generation", IEEE Trans Power Syst, 26 (1) (2011), pp. 198-205

[i.19] T Hedberg Jr, S Krima, J A Camelio: "Embedding X.509 digital certificates in three-dimensional models for authentication, authorization, and traceability of product data", Journal of Computing and Information Science in Engineering 17(1):11008–11011. https://doi.org/10.1115/1.4034131, 2016

[i.20] T Hedberg Jr, S Krima, J A Camelio: "Method for enabling a root of trust in support of product data certification and traceability", Journal of Computing and Information Science in Engineering 19(4):041003. https://doi.org/10.1115/1.4042839, 2019

[i.21] D Yaga, P Mell, N Roby, K Scarfone: "Blockchain technology overview", National Institute of Standards and Technology, Gaithersburg, MD, https://doi.org/10.6028/NIST.IR.8202, 2018

[i.22] S Krima, T Hedberg Jr, A Barnard Feeney: "Securing the digital threat for smart manufacturing", National Institute of Standards and Technology, Gaithersburg, MD, AMS 300-6. https://doi.org/10.6028/NIST.AMS.300-6, 2019

[i.23] D Wu, M J Greer, D W Rosen, D Schaefer: "Cloud manufacturing: Strategic vision and state-of-the-art", Journal of Manufacturing Systems 32(4):564–579. https://doi.org/10.1016/j.jmsy.2013.04.008, 2013

[i.24] X Vincent Wang, X W Xu: "An interoperable solution for cloud manufacturing", Robotics and Computer-Integrated Manufacturing 29(4):232–247. https://doi.org/10.1016/j.rcim.2013.01.005, 2013

[i.25] L Zhang, Y Luo, F Tao, B H Li, L Ren, X Zhang, H Guo, Y Cheng, A Hu, Y Liu: "Cloud manufacturing: a new manufacturing paradigm", Enterprise Information Systems 8(2):167–187. https://doi.org/10.1080/17517575.2012.683812, 2014

[i.26] L Ren, L Zhang, L Wang, F Tao, X Chai: "Cloud manufacturing: key characteristics and applications", International Journal of Computer Integrated Manufacturing 30(6):501–515. https://doi.org/10.1080/0951192X.2014.902105, 2017

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TR 103 778 [i.1] and the following apply:

Data consumer: AI, monitoring algorithm or human that uses the data provided by an IoT platform or device

NOTE: After the data consumer has used the data, they remain available for further usage.

ML algorithms: specific algorithms used to analyse data as well as any pre-processing or post-processing performed on the data before use in the ML algorithm

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the [following] abbreviations [given in ... and the following] apply:

AI	Artificial Intelligence
AI/ML	Artificial Intelligence/Machine Learning
API	Application Programming Interface
IoT	Internet of Things
IP	Intellectual Property
ROI	Return Of Investment

4 Recommendations for data usability

ETSI TR 103 778 [i.1] identifies and describes use cases where the IoT data and services require data usability for humans and for machines consuming data for AI (for example machine learning). The data that IoT devices and platforms provide should be easily accessed to all authorized users, understood and acted upon by a large non-technical public in the case of humans (e.g., medical teams and their patients in the medical sector, mechanics in the automotive sector, first responders in the emergency sector, etc.) and by machines and processes when the data are fed to the AI components of a system (e.g., machine learning). Its main objective is to analyse these use cases to derive requirements and guidelines towards a horizontal cross-domain standard, with the specification of minimum requirements for data usability of professional and general public IoT services, whether they are critical or not. In that aim, TR 103 778 [i.1] analyses the impact of these use cases from the data usability point of view for both machines (algorithms and AI/ML) and humans.

Potential solutions build up a list of what can mitigate the identified issues with the intent of decreasing the likelihood of these issues. Each use case has been analysed again to determine which potential solutions could be applied and then identify the residual impact assessment, with a goal to have the minimal residual impacts for each use case.

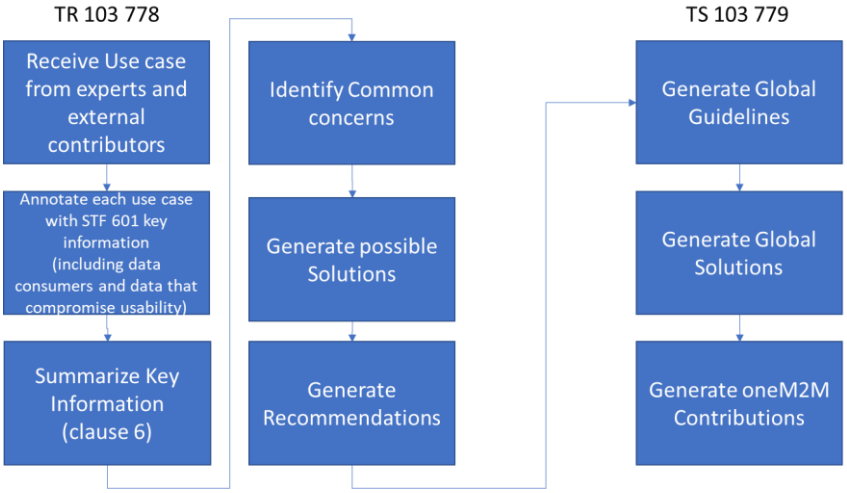

Figure 1: Link between the use cases and the specifications

This clause contains a summary describing the major points of attention to consider when an AI system is deployed. It provides a table describing a list of recommendations grouped by type and, for each of them, the recommendation that may be addressed to handle some of the impact to issues raised under the use cases that have been described in TR 103 778 [i.1]. The aim of this clause is to connect the outcomes of the work performed in TR 103 778 [i.1] with the set of requirements provided in clause 5.

Table 1: Summary of recommendations in TR 103 778 [i.1]

Category	Recommendation	Description
Setup	IoT infrastructure/devices bootstrap.	Easy way for sensor data to be directed to a data consumer (human or ML algorithm). Each deployed IoT infrastructure/device has to be properly setup in order to grant an efficient and effective flow of involved data. During the bootstrap operation it is necessary to check if all data gathered by sensors are easily provided to the target data consumers. Target data consumers may be both humans or ML algorithms.
	Data format description and intelligibility.	Data formats used within a deployed IoT infrastructure/device have to be properly described in order to avoid ambiguity for the target data consumers using such data. Target data consumers may be both humans or ML algorithms.
Configuration	Mitigation of data heterogeneity.	A complex IoT infrastructure/device may include data produced by means of different data formats (e.g., different sensor manufacturers, external API services). It may be necessary to foresee operations to mitigate the data heterogeneity. Such an operation is necessary to standardize the input data format exploited by ML algorithms and/or humans. Use of ontologies thought for specific domains (e.g., SAREF [1]) can be foreseen.
	Data quality.	Each IoT infrastructure/device has to be accompanied with appropriate metadata for each data source used, of the granularity and frequency with which each data source provides data. Such information is exploited for determining the suitability of data sources in different scenarios as well as for understanding how to configure ML algorithms to better exploit such data.
Machine Learning or monitoring output	Explainability.	Transparency is one of the most important challenges to address in ML field. Associated with the output produced by a ML algorithm (e.g., classification of an object based on the features provided as input), it is important to reconstruct the classification process through the meaning provided to the data of interest generated such a classification.

	Terminology.	Misunderstanding concerning the usage of terms is common. The definition of a precise vocabulary associated with the output produced by ML algorithms and with the meaning of each data feature is recommended. The usage of an ontology may be a proper way for providing such a terminology.
	Output management.	Output provided by ML algorithms has to be stored and described within an effective and efficient repository. Such a repository works as an enabler for making data easy to find for target data consumers and for supporting the retrieval and understanding of important information linked with them.
	Data duplication.	Data duplication is an issue that may affect the effectiveness of ML algorithms. This may happen when multiple instances of the same raw data are stored within the same repository. This fact may lead to the generation of biases during the building/update of classification models due to the usage of same data instance more than once.
	Traceability.	It is necessary to reconstruct the classification process through the identification of the ML modules providing specific outputs. This need is the basis for preserving the traceability of the data flow within the entire infrastructure.
IoT system operation	Data coordinates.	IoT infrastructure/device has to label data provided with both timing and location information when they are used in scenarios exploiting such information.
	Data access.	The deployment of an IoT infrastructure/device has to ensure a precise policy for managing the authorization to access data by all authorized data consumers and not authorized data consumers.
	IoT data interoperability.	IoT infrastructure/device may include IoT devices provided by different manufacturers adopting, in turn, different data format and exporting methods. It is recommended the integration of data interoperability modules for supporting the effective and efficient sharing of data provided between different IoT systems.
	Maintenance of IoT infrastructure/devices.	Complex IoT infrastructure/device has to define a maintenance policy ensuring the proper monitoring and maintenance of all components.
Security	Preservation of integrity, privacy and security.	All components of the deployed IoT infrastructure/device have to be compliant with standards and regulations related to privacy and security of data. Specific procedures have to be put in place for avoiding/managing data integrity breaches.

5 Requirements and guidelines for preserving data usability

5.1 General considerations

This clause describes the essential guidelines to follow for preserving the data usability. Here, an abstract conceptual model is provided for giving unambiguous definitions of each guideline and, at the same time, to pave the way for future developments. The secondary aim of this clause is to provide examples showing how this abstract conceptual model may be used for defining a checklist to address before deploying a new AI system.

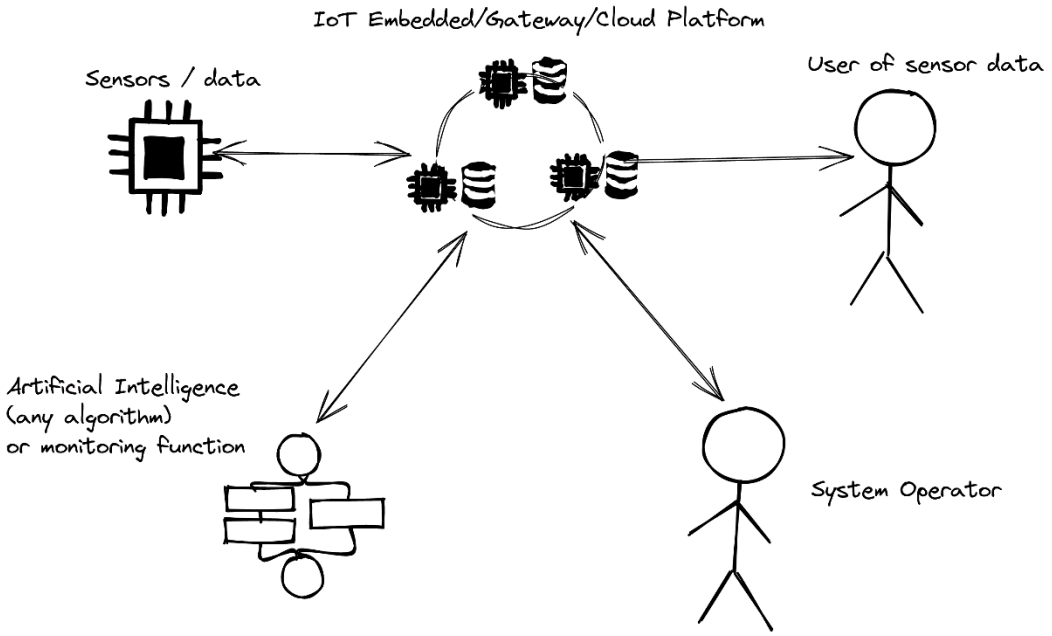

Figure 1: Architecture of an ML deployment

5.2 Service requirements

5.2.1 Requirements to be fulfilled by sensor/data sources

Requirement number	Related recommendation	Requirement
REQ_SERV_1_001	Terminology.	Data generated or provided by a sensor should have a description of the data using a shared terminology defined within an ontology.
REQ_SERV_1_002	Mitigation of data heterogeneity.	A data mitigation procedure should have foreseen if sensors generating data are provided by different manufacturers. Pre-processing of raw data from sensors to a format required by a ML algorithm may be a sufficient mitigation.
REQ_SERV_1_003	Data format description and intelligibility.	Data generated or provided by a sensor should have a description of the format used for generating or providing such data.
REQ_SERV_1_004	Data quality.	Data generated or provided by a sensor should have a description of the granularity (in terms of numerical precision, if any, and frequency) adopted for generating or providing such data.
REQ_SERV_1_005	IoT infrastructure/devices bootstrap.	The sensor data shall be available to their consumer (human or machine algorithm).
REQ_SERV_1_006	Data quality.	The sensor data confidence level should be known to enable proper processing by the data consumers. For example, a temperature sensor may provide meta-data describing the accuracy of the measurement from the device.

5.2.2 Requirements to be fulfilled by IoT platform

Requirement number	Related recommendation	Requirement
REQ_SERV_2_001	Data format description and intelligibility.	If historical data is available the IoT Platform shall allow download of data in a bulk format such as CSV, Apache Parquet, or other formats.
REQ_SERV_2_002	Terminology.	IoT platform shall allow linking of data to a semantic definition of the data.
REQ_SERV_2_003	Traceability.	IoT Platform shall support discovery of services or algorithms to process data coming from IoT data sources (e.g., sensors).
REQ_SERV_2_004	Data format description and intelligibility.	Data from the IoT platform shall be easily understandable for a data consumer monitoring the platform
REQ_SERV_2_005	Data format description and intelligibility.	Data presentation and integrity from an IoT platform shall ensure a valid algorithm / AI decision.
REQ_SERV_2_006	Mitigation of data heterogeneity.	Data from different sources should be transformed and/or aggregated, as necessary to fit into the ML algorithm and enable scalability.
REQ_SERV_2_007	Data quality.	The IoT platform should be designed in a scalable manner as a large number of objects may need to be tracked reliably with position, identification, and timestamp.
REQ_SERV_2_008	Traceability	Each data shall be uniquely identifiable. In cases where ML Algorithms generate a copy of data, a reference to the original source should be available as well.

5.2.3 Requirements to be fulfilled by AI/ML or monitoring function

Requirement number	Related recommendation	Requirement
REQ_SERV_3_001	Explainability.	Used machine learning algorithms shall be transparent and to provide explanations about the output produced. An appropriate ontology should be used.
REQ_SERV_3_002	Data format description and intelligibility.	Used artificial intelligence components shall provide a description of the features received as input. An appropriate ontology should be used.
REQ_SERV_3_003	Data format description and intelligibility.	Used artificial intelligence components have to provide the list of the data formats that they are able to read. For example: “Component X shall receive as input a list of natural language sentences already tokenized.”, or, “Component Y shall receive as input a set of 24 numbers in double precision in the race [0,1].”. When needed, an appropriate ontology should be used.
REQ_SERV_3_004	Data format description and intelligibility.	Used artificial intelligence components shall provide a description of the format provided as output. An appropriate ontology should be used.
REQ_SERV_3_005	Terminology. Output management.	The content of the report produced by the ML algorithm should be comprehensive and unambiguous to enable proper operation of the IoT system.
REQ_SERV_3_006	Output management.	The events generated by the platform shall be easy to understand without ambiguity by the system operator.
REQ_SERV_3_007	Preservation of integrity, privacy and security.	Monitoring components shall not be able to access data instances for which they are not granted authorization.
REQ_SERV_3_008	Output management.	The algorithm output should be able to highlight important data.

5.2.4 Requirements to be fulfilled by operator of system

Requirement number	Related recommendation	Requirement
REQ_SERV_4_001	Preservation of integrity, privacy and security.	The operators shall verify that the system is compliant with the regulations related to data privacy.
REQ_SERV_4_002	Preservation of integrity, privacy and security.	The operators shall verify that the system is compliant with regulations related to the ethical management of data.
REQ_SERV_4_003	Preservation of integrity, privacy and security.	The operators shall verify that the infrastructure does not present data integrity breaches.
REQ_SERV_4_004	Preservation of integrity, privacy and security.	Procedures for the management and resolution of possible data-related issues shall be defined.

5.2.5 Requirements to be fulfilled by data users

Requirement number	Related recommendation	Requirement
REQ_SERV_5_001	Output management.	Users should have the knowledge to access the outcome of the artificial intelligence components.
REQ_SERV_5_002	Data format description and intelligibility.	Users shall be equipped with tools able to read properly the data format with which outputs are produced.

5.3 Operational requirements

5.3.1 Requirements to be fulfilled by sensor/data sources

Requirement number	Related recommendation	Requirement
REQ_OPE_1_001	Data coordinates.	The data measured by the sensors shall be time-stamped. This will allow to evaluate a potential repetition rate.
REQ_OPE_1_002	Data quality.	When relevant, the geolocation measurement/ configuration of the remote sensors shall be reliable.
REQ_OPE_1_003	Data quality	The accuracy of the measurement results (quality of sensor data) shall be expressed as a percentage.
REQ_OPE_1_004	Maintenance of IoT infrastructure/devices.	The data consumer shall be able to reset the condition that led to an alert or to maintenance once it has been processed.

5.3.2 Requirements to be fulfilled by IoT platform

Requirement number	Related recommendation	Requirement
REQ_OPE_2_001	Output management.	Key metrics (latency, throughput, memory usage, processor utilization, disk space, resource capabilities (CPU and memory speed), temperature) should be defined and provisioned.
REQ_OPE_2_002	Data coordinates.	All data handled by the IoT platform should be properly timestamped and geolocated when relevant, to ensure traceability of the subsequent processing.
REQ_OPE_2_003	Data access.	Data from the IoT platform should be easily available to an authorized data consumer accessing from an external device.
REQ_OPE_2_004	IoT data interoperability.	The platform shall be able to propagate any data change to all components easily.
REQ_OPE_2_005	Maintenance of IoT infrastructure/devices.	The data consumer shall be able to understand how to act on the IoT platform to check the validity of data delivered by sensors (e.g. to identify faulty devices and sensors).

5.3.3 Requirements to be fulfilled by AI/ML or monitoring function

Requirement number	Related recommendation	Requirement
REQ_OPE_3_001	Data quality.	Used machine learning/monitoring algorithms shall verify the integrity of the data received as input.
REQ_OPE_3_002	Mitigation of data heterogeneity.	Used machine learning/monitoring algorithms shall verify the format of the data received as input.
REQ_OPE_3_003	Output management.	Used machine learning algorithms shall communicate the output of the data processing operations.
REQ_OPE_3_004	Maintenance of IoT infrastructure/devices.	Monitoring components shall alert in the case that new data are not provided.
REQ_OPE_3_005	Maintenance of IoT infrastructure/devices.	Monitoring components shall alert in the case that undesired events are detected.
REQ_OPE_3_006	Maintenance of IoT infrastructure/devices.	Monitoring components shall verify the persistency of the connection with data sources.
REQ_OPE_3_007	Data duplication.	Used machine learning algorithms shall mitigate data duplication issues to avoid biases during training operations.
REQ_OPE_3_008	Explainability.	Used machine learning algorithms shall provide the description of the semantic meaning of input characteristics.
REQ_OPE_3_009	Explainability.	Used machine learning algorithms shall provide a description concerning the motivations for which a specific classification has been provided by the platform with respect to the input features.
REQ_OPE_3_010	Data quality.	Used AI algorithms or monitoring functions should implement a semantic-oriented policy to describe fine-grained details of data features (e.g., data range provided by a specific sensor, security levels)

5.3.4 Requirements to be fulfilled by operator of system

Requirement number	Related recommendation	Requirement
REQ_OPE_4_001	IoT infrastructure/devices bootstrap.	At the time of deployment, operators shall verify that the overall infrastructure works properly and that all components are able to communicate each other.
REQ_OPE_4_002	IoT infrastructure/devices bootstrap.	At the time of deployment, operators shall verify that all human target users are able to receive required data from the system.
REQ_OPE_4_003	IoT infrastructure/devices bootstrap.	At the time of deployment, operators shall verify that all artificial intelligence components are able to receive required data from the system.

REQ_OPE_4_004	Maintenance of IoT infrastructure/devices.	Maintenance should be performed periodically to verify the proper operation of the system and prevent failure of the devices and sensors.
REQ_OPE_4_005	IoT data interoperability.	At the time of deployment, operators shall verify that the format of the IoT platform data is understandable by any external device or human expected to consume them ~~use it~~.
REQ_OPE_4_006	Preservation of integrity, privacy and security.	All data consumers who may need to access them shall be granted authorized access to the IoT platform data.
REQ_OPE_4_007	Data format description and intelligibility	The deployed system should be scalable, accepting inputs from all sorts of sensors if relevant.
REQ_OPE_4_008	Data format description and intelligibility.	Object identification should be setup and configured properly to prevent mishandling of objects by the IoT platform.
REQ_OPE_4_009	Data coordinates.	Data from all object sources should be synchronized (e.g. identical time reference).
REQ_OPE_4_010	Preservation of integrity, privacy and security	Privacy of personal data should be ensured for the IoT platform user and all affected humans (see also [2]).
REQ_OPE_4_011	Preservation of integrity, privacy and security	The data flow for safety applications shall be secured (see also [2]).

5.3.5 Requirements to be fulfilled by user of data

Requirement number	Related recommendation	Requirement
REQ_OPE_5_001	Data access.	Users shall possess the required authorization for accessing data.

6 Conclusion

Editor's note: [Final remarks.]

Annex A (informative): Challenges in adopting the guidelines and about the integration of such guidelines within automatic validation systems

A.0 Introduction

This annex provides a more in-depth discussion about some specific challenges in adopting data usability guidelines and about the integration of such guidelines within automatic validation systems. In particular, we intended to deepen the Interoperability, Data collection, Granularity, and Traceability aspects of data.

A.1 Interoperability

Interoperability is a characteristic of good quality data, and it relates to broader concepts of value, knowledge creation, collaboration, and fitness-for-purpose. Interoperability exists in varying degrees and forms, and interoperability issues need to be broken down into their key components, so that they can be addressed with concrete, targeted actions. Conceptual frameworks help us to consider interoperability in different contexts and from different perspectives. For instance:

- from a diversity of technological, semantic, or institutional viewpoints, recognizing that interoperability challenges are multi-faceted and manifest in different ways across scenarios and use cases; and
- within the context of the data value chain, as well as within the context of broader data ecosystems.

Following the Data Commons Framework [i.2], we can split out the concept of interoperability into a number of narrow and broad layers that relate to standardization and semantics respectively. These layers can help in the development of projects, plans, and roadmaps to better understand interoperability needs at various points and can be summarised thus:

1. Technology layer: this layer represents the most basic level of data interoperability, and is exemplified by the requirement that data be published, and made accessible through standardized interfaces on the web;
2. Data and format layers: these layers capture the need to structure data and metadata according to agreed models and schemas, and to codify data using standard classifications and vocabularies;
3. Human layer: this layer refers to the need for a common understanding among users and producers of data regarding the meaning of the terms used to describe its contents and its proper use (there is an overlap here with the technology and data layers, in that the development and use of common classifications, taxonomies, and ontologies to understand the semantic relationships between different data elements are crucial to machine-to-machine data interoperability);
4. Institutional and organisational layers: these layers are about the effective allocation of responsibility (and accountability) for data collection, processing, analysis and dissemination both within and across organizations.

Table 2: Summary of basic recommendations on interoperability aspects

Action Areas	Initial Recommendations
Modelling data structures	Starting from a set of source tables, identify elementary datasets to be modelled (variables or indicators). Identify key entities that are described in the information contained in the dataset (e.g., places, people, businesses…): • identify the dimensions and attributes needed to describe each entity at the target level of granularity (e.g., location, time period, sex, etc.); • to the extent possible, re-use standard dimensions and naming conventions from existing data models (e.g., from existing SDMX data structure definitions); • consider merging or splitting columns from original tables to define more useful dimensions for data exchange. Create a separate table of distinct values for each dimension, assigning a unique numeric ID to each row.

Modelling metadata	Identify a minimum set of metadata elements relevant to describe the dataset. Map all relevant metadata elements to DCAT vocabulary classes.
Quality considerations	Internally consider user needs and data quality considerations when deciding an approach to modelling.
Using common classifications and vocabularies	Identify relevant, publicly available, and widely used classifications and vocabularies that can be re-used to codify and populate the content of dimensions, attributes, and measures in a data set. Adopt standard vocabularies and classifications early on, starting at the design phase of any new data collection, processing or dissemination system.
Creating semantic interoperability Between classifications	Engage in structural and semantic harmonization efforts, mapping “local” terminology used to designate measures and dimensions to commonly used, standard vocabularies and taxonomies.
Using open data formats	Make the data available through bulk downloads, for example as CSV files. Use other widely-available open data formats to encode common data elements and sub-elements in such a way that data and metadata are linked together but clearly distinguishable from each other. Use a data containerization format such as the Data Package standard format to publish data sets.
Using standard APIs	Set up a webpage and document all the functionality of existing web APIs in use, describing the resources being provided, the operations that can be performed, as well as the inputs needed for, and outputs provided by, each of operation. Provide additional information such as contact information, any licences used, terms of use, etc..
Enhancing user experience	Follow common design patterns and rules of communication, so users can easily and intuitively interact with system interfaces.
Linking data on the semantic web	Select datasets to be openly linked on the semantic web. Create HTTP URIs to identify datasets. Map the dimensions used to describe the data to existing vocabularies and ontologies.
Publishing open linked data	Publish the original dataset using JSONLD, Microdata, RDF, or any other format that references the mapped metadata terms. Publish any new concepts using existing vocabularies or ontologies (e.g., SKOS), and make them available on the web. If the new concepts are specializations of existing ones, extend those existing vocabularies with sub-classes and sub-properties derived from base concepts and properties.

A.2 Collecting data from sensors

It is worth noting that both of human-oriented and machine-oriented IoT applications demand some specific requirements for preserving an effective and efficient data collection from IoT devices, including, but not limited to, data rate, latency, coverage, power, reliability, and mobility [i.3], [i.4]. These requirements may overlap with each other and may cause a trade-off for the application’s performance. These requirements represent the six main aspects which should be considered at design time and during the deployment of a distributed platform. In particular, drawbacks associated with each of such aspects may affect the effectiveness of possible AI-based solutions relying on gathered data.

1) Data Rate. IoT applications can have different data transmission rates from tens of kbps up to tens of Gbps. Three different application groups can be identified in terms of data rate as follows: 1) high data-rate (greater than 10Mbps), 2) medium data-rate (less than 10Mbps and greater than 100kbps), and 3) low data-rate applications (less than 100 kbps) [i.5]. First, high data-rate applications such as streaming video and web applications. mostly transmit multimedia contents that require high data rate connectivity technologies. Second, medium data rate applications such as smart home applications include a set of connected devices in homes such as connected cooking systems with medium data rate requirements [i.6]. Finally, low data-rate applications include most of the monitoring sensors, goods tracking, smart parking and intelligent agriculture systems [i.7]

2) Latency. Most of IoT applications are sensitive to latency. But, the level of sensitivity varies for different applications. Due to this difference, the applications with high and low sensitivity to the latency are categorized into delay-sensitive and delay-tolerant groups, respectively [i.8]. Autonomous vehicles and health-care systems are two examples of delay-sensitive applications where the shortest possible latency is a critical factor that affects their performance [i.3]. To be specific, autonomous vehicles are such driver-less cars that can move automatically and sense their environment to avoid any hazard or accident. Consequently, when the vehicles move at a high speed, latency plays

a pivotal role in sensing the environment and make a decision as soon as possible. Likewise, health-care systems (e.g., cardiac telemetry) require to report the possible risks to a distant monitoring station with low latency to assist patients with early treatment.

3) Coverage. The maximum range of communications for IoT applications varies from couple of meters up to tens of kilometres. The IoT applications which require a communication range of up to tens of meters are categorized as short-range IoT applications. For example, smart home and smart retail applications include a range of connected items/objects in the range of 100m that are considered as short-range applications. On the other hand, the applications with distant connected items/objects (i.e., up to tens of kilometres) are classified as long-range IoT applications (e.g., smart farming and UAV) [i.5], [i.9]. The current technologies would not be able to support this massive connectivity. Therefore, the emerging technologies (e.g., NOMA, mMIMO, ML-assisted cellular IoT) can be used in future IoT connectivity paradigms.

4) Power. Power efficiency is an important requirement that affects the cost of IoT devices. Battery production, recycling, and environmental issues are also important factors that need to be considered in designing IoT applications. For example, even though the smart electric vehicles will not be using the fossil fuel to power the vehicles, they can still cause other environmental problems if the vehicles are not recharged or recycled properly [i.10]. Therefore, all the IoT applications seek the lowest possible power consumption technologies for low maintenance costs and also for achieving a lower impact on the environment. Most of the human-oriented applications (e.g., smartphones) are able to be charged regularly. However, the most challenging issues appear for ultra-low power consumption applications adopting technologies suffering from the limit of not being able to be charged regularly. For example, applications like agricultural metering sensors normally require the terminal service life with a constant volume battery up to 10 years [i.5], [i.11], [i.12].

5) Reliability. In terms of the reliability of the transmissions, IoT applications can be categorized into two major groups of mission critical and mission non-critical applications [i.13]. Smart grids, manufacturing robots, autonomous vehicles, and mobile health-care are some examples of mission critical applications [i.14]. On the other hand, the majority of IoT applications are mission non-critical IoT applications such as humidity sensors, smart green houses, smart parking, and energy and water meters. Overall, in order to guarantee sufficient reliability for such applications in both critical and non-critical systems, different requirements of end-to-end latency, ubiquity, availability, security, and robustness of the technologies should be assessed [i.3].

6) Mobility. IoT applications can be classified into two categories in terms of mobility: low and high mobility applications. Low mobility applications can easily rely on existing connectivity technologies [i.15]. The challenging issues appear in high mobility applications where the speed can go up to hundreds km/h and consequently they demand for handover, redirection, and cell reselection in connected states. Some examples of high mobility IoT applications are such as vehicles, trains and airplanes demanding enhanced connectivity for in-vehicle/on-board entertainment, accessing the Internet, enhanced navigation through instant and real-time information, autonomous driving, and vehicle diagnostics [i.3]. In general, high mobility applications utilize cellular connectivity technologies. However, they require significant improvements in current cellular technologies (e.g., 4G and 5G) to overcome high mobility issues for future high mobility applications [i.16]. It is evident that IoT applications can be mapped into multiple categories at the same time to find the best possible connectivity technology. For example, smart agricultural sensors, [i.17], are usually considered as machine-oriented, low data rate, delay-tolerant, long-range, low power, non-critical, and low mobility applications.

A.3 Granularity

Data granularity is the level of detail considered in a model or decision-making process or represented in an analysis report. The greater the granularity, the deeper the level of detail. Increased granularity can help you drill down on the details of each marketing channel and assess its efficacy, efficiency, and overall Return Of Investment (ROI).

For example, within the pharma industry, knowing which marketing channels work for each brand segment is far more informative than knowing what is working for the company as a whole. Increased granularity can help you examine each brand’s performance and make specific, targeted adjustments to discrete variables to improve sales and

profitability. Rather than using a shotgun approach, increasing data granularity allows you to focus your marketing with laser-scope precision.

Even if one can assume that increased data accuracy leads to more reliability of the systems, this does not necessarily imply it improves its effectiveness. The consequences of adding precision to the input of optimization models is rarely discussed in scientific literature. Ochoa and Harrison [i.18] provide a first step in the discussion by advocating the use of multi-period data models for loss minimization instead of the popular one-period data models. Multi-period data models evaluate the system at one moment in time, using a snapshot of the systems performance for optimization. One-period models, on the other hand, allow to evaluate the system over a span of time, thereby accounting for time variations and time dependencies.

The aforementioned research on granularity and decision making acknowledges the stochastic nature of the short-term fluctuations by considering several likely data profiles (samples or snapshots) rather than one average profile. This, way the computational burden of the analysis can be kept to a reasonable level, without needing to compromise on the complexity of the model. The stochastic nature of short-term fluctuations could impact optimal decisions, which may not be adequately captured using a small number of samples. Hence, it is recommended to take the full span of possible realizations into account by framing the problem in the language of stochastic optimization. This allows for consideration of the complete range of stochastic fluctuations in the model at the cost of the level of detail that can be included in the model.

Further analysis results recommend that for optimization purposes it is not always necessary to use fine-grained data. In fact, the high-resolution data show that many solutions are similar in outcome, such that even near-optimal solutions can give satisfactory outcomes. Considering the computational burden and limits to modelling flexibility that come with using high resolution data it is thus advised not to use data with time steps smaller than one hour for optimization. However, when evaluating the current state of a system rather than optimizing the system it may be relevant to increase granularity. When done so it is advised to acknowledge the full spectrum of the probabilistic nature of the variables, rather than just a couple of scenarios, such that the optimization process is less prone to be influenced by outliers in the samples. Also, when the objective is not to optimize some sort of average performance of the system (cost effectiveness, real losses etc.) but to increase performance under worst case scenarios (reliability), the short-term fluctuations may be important for the process of optimization.

The intuitive time-domain and phase domain granulation was shown to require precise alignment of the granulation window with the significant changes in the data. If such an alignment is not performed the methods return a generally poor result. The optimisation of the information density approach results in a much-improved granulation that exhibits several desirable features:

- information granules are compact;
- small data groupings are fully taken into account;
- the local nature of optimisation allows for distributed computations; and,
- the linear computational complexity with respect of the dimensionality of the pattern space makes it suitable for highly
- dimensional data.

A.4 Traceability

A.4.1 Logging

Data traceability is paramount to enabling trustworthiness throughout the product lifecycle. Simply providing a digital signature on data is neither sufficient nor feasible due to the complexity of the supply chain and the heterogeneity of the data exchanged. This gap was realized through validation of existing work [i.19], [i.20].

In a complex environment composed of numerous partners and exchanges, embedding traceability data in only files can bloat the product data with information not required by every actor. A complete traceability cannot be guaranteed due to the heterogeneity of the data and the need for every file format to support a traceability mechanism. Proprietary and/or binary files are heavily used and do not offer an efficient transparent way of auditing the traceability information. Moreover, numerous open-formats may not support such a mechanism either. Lastly, embedding traceability information in files makes the audit process cumbersome, requiring access to and processing of all the files, which is an

enormous amount of data. Therefore, to overcome these challenges and to address efficient audit needs, we suggest to combine previous work with recording traceability information externally in a safe and shared repository such as a distributed ledger [i.21] that offers a shared, trusted, and virtually tamper-resistant source of information.

We identified three main types of recommendations concerning the traceability of data transactions throughout the running lifecycle of an AI-enabled system depending on the origin of the data. A data transaction occurs anytime data ownership is declared or when data is exchanged between two actors.

A.4.2 File-Based Traceability Recommendation

File-only transactions are asynchronous, require significant leveraging of certificates (or any equivalent validation process), and require trust of the other actors with whom data is exchanged. Traceability is managed with metadata stored within the data files.

Three actors are, in general, depicted in the file-based traceability option: (1) data owner, (2) data consumer, and (3) bad actor. The data owner (herein owner) and data consumer (herein consumer) are the normal roles that would typically share data while executing tasks. When the owner is prepared to release the data to the consumer, the owner could review and sign the data. Then, the owner would send the data to the consumer. The owner and consumer would store that signed data in their respective data repositories. The consumer would use the data to complete all agreed-upon tasks for the owner (e.g., supplier fabricates a part for a customer). This portion of the use case represents typical manufacturing-related business relationships.

Data could be compromised and/or stolen from owners and consumers by bad actors. In the file-based traceability option, a bad actor could steal data from the consumer by compromising (e.g., gaining unauthorized access) the consumer's data repository. The bad actor would have access to the signed data. If the owner then found the signed data in the possession of an unauthorized actor, the owner could go back to his/her repository and determine all the consumers the data was sent to by querying and reviewing the certificate and metadata. This would provide the owner the ability to discover who received the data and request those consumers to investigate their systems for breaches. In this case, the owner would simply discover that he/she has a data problem, but the owner would not immediately know the root cause of that problem without further investigation.

However, the file-based traceability option represents a solid foundation with which to build data-traceability principals and methods. Having the ability to quickly impart additional metadata into a file and then later be able to trace where the data came from, its purpose, and potential uses would reduce the risk of errors being introduced due to the wrong data being used or because of changes that went unnoticed.

A.4.3 Distributed Ledger Recommendation

Distributed-ledger transactions are synchronous and, usually, require leveraging certificates and a technology like blockchain [i.21]. Traceability is managed with transactions registered in a distributed ledger. The same three actors depicted in clause A.4.2 are also depicted here. The owner and consumer are still the normal roles that would typically share data between each other for the purposes of executing tasks.

However, in this case, when the owner is prepared to release and send the data to the consumer, the owner would review and sign the data and register the signature fingerprint in a distributed ledger to prove ownership of the data. [i.22] recommends storing only the signature fingerprint in the distributed ledger, registering the signature fingerprint in a transaction sent by the owner to him/herself for proving ownership, and then registering the signature fingerprint in transactions whenever the data is sent to a user [i.22].

The owner and consumer would still store signed data in their respective data repositories. The consumer would also still use the data to complete all agreed-upon tasks for the owner (e.g., supplier fabricates a part for a customer). This portion of the use case, like the file-only transactions, represents typical manufacturing-related business relationships with the only difference being that each action on the data is registered in a distributed ledger.

The strength of the distributed-ledger traceability option is in dealing with bad actors. If the owner found signed data in possession of a bad actor, the owner could query the distributed ledger and determine the exact transaction that was related to the compromised data. This provides the owner the ability to discover exactly who was authorized to receive the data originally and request that consumer to investigate his/her systems for breaches.

In this case, the distributed-ledger traceability scenario is differentiated from the file-based traceability one because the owner would discover that he/she has a data problem and immediately know the root cause of the problem without further investigation.

A.4.4 Streaming-Data Packages Recommendation

Streaming-data packages is an emerging scenario for providing data traceability and protecting the IP included in the packages. In the additive-manufacturing domain, a few commercial proprietary platforms exist that claim to stream data directly to the manufacturing machines for fabricating hardware. However, all the commercial solutions are closed platforms and the state of their standards implementations are unknown.

The research literature also includes several papers related to distributed cloud manufacturing or manufacturing-as-a-service [i.23] [i.24] [i.25] [i.26]. These papers claim the data could be streamed to a localized manufacturing services regardless of process. Putting the feasibility of the technologies aside, most of the research literature proposes different methods for streaming-data packages. The research literature does not currently propose a common method for streaming-data packages. While there is a significant amount of activity and possible solutions available for streaming data packages, more work to achieve consensus on an approach is needed. Further, there is little research in digitally signing streaming data. Therefore, the file-based traceability and distributed ledger options are the recommended approaches until consensus is achieved for streaming-data packages.

Annex (informative): Change History

Date	Version	Information about changes
10-2021	0.0.1	Early draft uploaded for TC SmartM2M Ad-hoc meeting
12-2021	0.0.2	Revised Early draft uploaded for TC SmartM2M #60
01-2022	0.0.3	Stable draft uploaded for TC SmartM2M Ad-hoc meeting
02-2022	0.0.4	Revised stable draft uploaded after comments from TC SmartM2M Ad-hoc meeting

History

Document history		
<Version>	<Date>	<Milestone>