ETSI GR ENI 004 V2.1.1 (2019-10)

GROUP REPORT

Experiential Networked Intelligence (ENI); Terminology for Main Concepts in ENI

Disclaimer

The present document has been produced and approved by the Experiential Networked Intelligence (ENI) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference

RGR/ENI-0010

Keywords

artificial intelligence, network, terminology

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

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Contents

Intellectual Property Rights 4

Foreword 4

Modal verbs terminology 4

1 Scope 5

2 References 5
2.1 Normative references 5
2.2 Informative references 5

3 Definition of terms, symbols and abbreviations 5
3.1 Terms 5
3.2 Symbols 15
3.3 Abbreviations 15

Annex A: Bibliography 19

History 20

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 Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Experiential Networked Intelligence (ENI).

Modal verbs terminology

In the present document "**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.

1 Scope

The present document provides terms and definitions used within the scope of the ETSI ISG ENI. The purpose is to define a common lexicon for use across all deliverables of ENI.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

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 GS NFV 003 (V1.3.1): "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.2] MEF PDO CfC: "Policy-Driven Orchestration", v0.8, February 2018.

[i.3] MEF 55.0.3: "Amendment to MEF 55: Service Orchestration Functionality", January 2018.

[i.4] MEF 55: "Lifecycle Service Orchestration (LSO): Reference Architecture and Framework", March 2016.

[i.5] MEF MCM CfC: "MEF Core Model", March 2018.

[i.6] Gamma E., Helm R., Johnson R. and Vlissides J.: "Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley, November 1994. ISBN 978-0201633610.

[i.7] ISO/IEC 2382-28: "Information technology -- Vocabulary -- Part 28: Artificial intelligence -- Basic concepts and expert systems".

[i.8] ISO/IEC/IEEE 42010: "Systems and software engineering -- Architecture description".

[i.9] IETF RFC 4949 (August 2007): "Internet Security Glossary, Version 2", R. Shirey.

3 Definition of terms, symbols and abbreviations

3.1 Terms

The purpose of the present document is to provide the terms to be used in ETSI ISG ENI deliverables.

0 to 9

Void.

A

abstraction: process of focusing on the important characteristics and behaviour of a concept and realizing this as a set of one or more elements in an information or data model

NOTE: When applied to modelling, it defines a generic set of characteristics and behaviours for a class that all of its subclasses inherit. This enables the definition of concepts to be separated from their implementation.

action: set of operations that may be performed on a set of managed entities, it represents a transformation or processing in the system being modelled

NOTE: An Action either maintains the state, or transitions to a new state, of the targeted managed entities. The execution of an Action may be influenced by applicable attributes and metadata. As defined in MEF PDO CfC [i.2].

agent: computational process that implements the autonomous, communicating functionality of an application:

- **software agent:** software that acts on behalf of a user or another program

- **software autonomous agent:** software agent that acts on behalf of the entity that owns it without any communication from the owning entity

- **software intelligent agent:** software agent that reasons about its environment and take the best set of actions to satisfy a set of goals

NOTE: This has the connotation of containing AI mechanisms to provide the reasoning and decision-making capabilities.

- **software multi-agent:** set of software agents that are physically separate that work together to satisfy a set of goals

Application Programming Interface (API): API, or application programming interface, is a set of communication protocols, code, and tools that enable one set of software components to interact with either a human or a different set of software components

API broker: mediates between two systems with different APIs, defining the correct way for one system to request services from the other system

architecture: set of rules and methods that describe the functionality, organization, and implementation of a system:

- **cognitive architecture:** defines a system that learns, reasons, and makes decisions in a manner resembling that of a human mind

NOTE: Specifically, the learning, reasoning, and decision-making is performed using software that makes hypotheses and proves or disproves them using non-imperative mechanisms that typically involve constructing new knowledge dynamically during the decision-making process.

- **deliberative architecture:** defines a symbolic world model that enables problem-solving components to be built using a sense-plan-act paradigm

- **hybrid architecture:** combines reactive and deliberative components into a hierarchy of interacting layers, where each layer reasons at a different level of abstraction

- **reactive architecture:** defines a system that is aware of changes that affect its computations and adjusts accordingly (https://www.reactivemanifesto.org/)

NOTE: The adjustment is made by reacting to an event in real-time without centralized control. The availability of new information drives program logic execution.

- **software architecture:** defines the high-level structure and organization of a software-based system. This includes the objects, their properties and methods, and relationships between objects

assisted system: system that the ENI System is providing recommendations and/or management commands to is referred to as the "Assisted System"

Artificial Intelligence (AI): computerized system that uses cognition to understand information and solve problems

NOTE 1: ISO/IEC 2382-28 [i.7] defines AI as "an interdisciplinary field, usually regarded as a branch of computer science, dealing with models and systems for the performance of functions generally associated with human intelligence, such as reasoning and learning".

NOTE 2: In computer science AI research is defined as the study of "intelligent agents": any device that perceives its environment and takes actions to achieve its goals.

NOTE 3: This includes pattern recognition and the application of machine learning and related techniques.

NOTE 4: Artificial Intelligence is the whole idea and concepts of machines being able to carry out tasks in a way that mimics the human intelligence and would be considered "smart".

B

Void.

C

capability: set of features that is available from a component

NOTE: These features may, but do not have to, be used. All capabilities should be announced through a dedicated reference point. As defined in MEF PDO CfC [i.2].

choreography: set of processes that define how entities interact from a global point-of-view

NOTE: That is without a single point of control. Compare this definition to orchestration.

closed loop control: self-regulating mechanism in which outputs of a system are provided to a system that compares the current state to a desired state (or set of states); the comparison is then used to adjust the behaviour of the system

NOTE 1: Positive feedback increases the correction value, while negative feedback reduces the correction value.

NOTE 2: Positive and negative feedback can be combined to achieve the needs of a system. In addition, more complex forms of closed loop control exist, such as proportional-integral-derivative (PID) control. See control theory.

cognition: process of understanding data and information and producing new data, information, and knowledge

component: part of a system that has operational and/or management significance

NOTE: A software component is an encapsulation of a set of related functions and/or data that perform a set of specific purposes and have a set of associated semantics and behaviour.

compute node: object that performs a set of calculations according to a set of algorithms

condition: set of attributes, features, and/or values that are to be compared with a set of known attributes, features, and/or values in order to determine what decision to make

container: object that stores collections of other objects in an organized manner

context: context of an entity is a collection of measured and inferred knowledge that describe the environment in which an entity exists or has existed

control plane: communication between entities that enables forwarding and routing of traffic to work

NOTE: Control plane packets are destined to, or locally originated, by entities themselves (e.g. they go to a network entity and direct how traffic flows). Compare to data plane.

control theory: application of mechanisms to regulate the behaviour of a target system

NOTE: Control theory includes linear and nonlinear control mechanisms.

D

data model: representation of concepts of interest to an environment that is dependent on data repository, data definition language, query language, implementation language, and/or protocol (typically, but not necessarily, all five)

NOTE: As defined in MEF PDO CfC [i.2].

data plane: path that the end-user traffic takes through a network

NOTE It is made up of traffic that goes through network entities, not to a network entity. Compare to control plane.

decision making: set of processes that result in the selection of a set of actions to take from among several alternative possible actions

declarative policy: type of policy that uses statements to express the goals of the policy, but not how to accomplish those goals

NOTE 1: State is not explicitly manipulated, and the order of statements that make up the policy is irrelevant.

NOTE 2: In the present document, Declarative Policy will refer to policies that execute as theories of a formal logic.

NOTE 3: As defined in MEF PDO CfC [i.2].

design pattern: general, reusable solution in a given context to a commonly occurring software problem:

NOTE: This type of design pattern is not an architecture and not even a finished design; rather, it describes how to build the elements of a solution that commonly occurs. It may be thought of as a reusable template.

- **design pattern, architecture:** general, reusable solution in a given context to a commonly occurring problem in the design of the software architecture of a system

- **design pattern, software**: general, reusable solution in a given context to a commonly occurring problem in the design of a software system

designated entity: operator, NMS, EMS, controller, or orchestrator acting on behalf of the Assisted System

NOTE: The Designated Entity is a trusted entity [i.9].

domain: collection of entities that share a common purpose, and which are governed in a common way

NOTE: As defined in MEF MCM CfC [i.5].

E

ENI application programming interface: set of communication mechanisms applied between two or more software components

NOTE: It consists of tools, object methods and other elements of a model and/or code. APIs simplify producing programs, since they abstract the underlying implementation and only expose objects and flow of information, and the characteristics and behaviour of those objects. This prevents the unnecessary exposure of objects.

ENI external reference point: reference point that is used to communicate between an ENI functional block and an external functional block (e.g. a functional block of the OSS, BSS, or assisted system)

NOTE: Where an ENI external reference point crosses between two organizational entities is not specified in this release.

ENI framework: set of abstractions that provide reusable and extensible mechanisms to provide generic functionality

NOTE 1: The ISO/IEC/IEEE 42010 [i.8] defines the term **architecture framework** as: "An architecture framework establishes a common practice for creating, interpreting, analysing, and using architecture descriptions within a particular domain of application or stakeholder community".

NOTE 2: The ENI framework also uses its abstractions to enable the ENI system to dynamically adapt to changing business goals, user needs, and environmental conditions. The ENI Framework hence provides a standard way to build and deploy applications and application components.

ENI hardware interface: point across which electrical, mechanical, and/or optical signals are conveyed from a sender to one or more receivers using one or more protocols

NOTE: A hardware interface decouples the hardware from other functional blocks in a system.

ENI interface: interface is a point across which two or more components exchange information

NOTE 1: An interface describes the public characteristics and behaviour that specify a contract for performing a service.

NOTE 2: In ENI, there are Hardware Interfaces and Software Interfaces.

ENI internal reference point: reference point that is used to communicate between two or more ENI functional blocks

NOTE: This relationship stays within the ENI framework, and cannot be addressed by systems that are external to the ENI framework.

ENI ISG PoC proposal: initial description of a PoC project, submitted as a contribution for review and acceptance by the ENI ISG before the PoC project starts

ENI ISG PoC report: detailed description of the results and findings of a PoC project, submitted once the PoC Project has finished

ENI reference point: logical point of interaction between specific functional blocks

NOTE: Reference point defines a set of related interfaces that specify how the functional blocks communicate and interact with each other.

ENI software interface: point through which communication with a set of resources (e.g. memory or CPU) of a set of objects is performed

NOTE: This decouples the implementation of a software function from the rest of the system.

ENI system: set of entities, based on the "observe-orient-decide-act" control loop model, that produces commands, recommendations, and knowledge to assist or direct the management of another system

NOTE: The ENI system is an innovative, policy-based, model-driven entity that uses artificial intelligence and other mechanisms to provide intelligent service operation and management. It is the enabler of intelligent Infrastructure management, network operations service operation and management, and assurance. It automates complex human-dependent decision-making processes. It also provides the ability to ensure that automated decisions taken by the system are correct and are made to increase the reliability, security and maintenance of the network and the applications that it supports. It also includes hardware and software components, programs, and system and user documentation.

Event-Condition-Action (ECA): type of imperative policy in which actions can only execute if the event and condition clauses are true

NOTE: An ECA policy rule is activated when its event clause is true; the condition clause is then evaluated and, if true, enables the execution of one or more actions in the action clause. This type of policy explicitly defines the current and desired states of the system being managed.

Experiential Networked Intelligence (ENI): processes associated with assimilating and understanding knowledge and learning through experience

NOTE: Adding closed-loop artificial intelligence mechanisms based on context-aware, metadata-driven policies enables the network to more quickly recognize and incorporate new and changed knowledge, and hence, make actionable decisions. This enables the network functionality to evolve and become better able to meet the demands of its operators with continued usage.

F

formal logic: use of inference applied to the form, or content, of a set of statements

NOTE: The logic system is defined by a grammar that can represent the content of its sentences, so that mathematical rules may be applied to prove whether the set of statements is true or false. Refer to MEF PDO CfC [i.2].

formal methods: set of mathematical theories, such as logic, automata, graph or set theory, that provide associated notations for describing and analysing systems

NOTE: As used in MEF PDO CfC [i.2].

functional block: modular unit that defines the properties, behaviour, and relationships of a part of a system

NOTE: With respect to ENI, functional blocks may be categorized as external (meaning that other systems external to ENI can see them) and internal (meaning that the functional block is only visible to other ENI functional blocks). External functional blocks use Reference Points to provide access to their functionality. Internal functional blocks use private interfaces to provide access to their functionality. As used in MEF 55.0.3 [i.3].

G to H

Void.

I

imperative policy: type of policy that uses statements to explicitly change the state of a set of targeted objects

NOTE 1: The order of statements that make up the policy is explicitly defined.

NOTE 2: In the present document, imperative policy will refer to policies that are made up of events, conditions, and actions. As defined in MEF PDO CfC [i.2].

information model: representation of concepts of interest to an environment in a form that is independent of data repository, data definition language, query language, implementation language, and protocol

NOTE: As defined in MEF PDO CfC [i.2].

intent policy: type of policy that uses statements to express the goals of the policy, but not how to accomplish those goals

NOTE 1: Each statement in an intent policy may require the translation of one or more of its terms to a form that another managed functional entity can understand. As defined in MEF PDO CfC [i.2].

NOTE 2: In the present document, Intent Policy will refer to policies that do not execute as theories of a formal logic. They typically are expressed in a restricted natural language and require a mapping to a form understandable by other managed functional entities.

J

Void.

K

knowledge: analysis of data and information, resulting in an understanding of what the data and information mean

NOTE: Knowledge represents a set of patterns that are used to explain, as well as predict, what has happened, is happening, or is possible to happen in the future; it is based on acquisition of data, information, and skills through experience and education.

- **inferred knowledge:** knowledge that was created based on reasoning, using evidence provided
- **measured knowledge:** knowledge that has resulted from the analysis of data and information that was measured or reported