



Experiential Networked Intelligence (ENI); System Architecture

STANDARD PREVIEW
(standards.iteh.ai)
Full standard:
<https://standards.iteh.ai/catalog/standards/sist/4692cf-5a12-4912-9a9e-c18b8fd078fb/etsi-gs-eni-005-v1.1.1-2019-09>

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.

ReferenceDGS/ENI-005

Keywords

management, network, policy management

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

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2019.

All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.

3GPP™ and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.

GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Contents

Intellectual Property Rights	8
Foreword.....	8
Modal verbs terminology.....	8
Executive summary	8
Introduction	8
1 Scope	9
2 References	9
2.1 Normative references	9
2.2 Informative references.....	10
3 Definition of terms, symbols and abbreviations.....	11
3.1 Terms.....	11
3.2 Symbols.....	13
3.3 Abbreviations	13
4 Overview of System Architecture (informative).....	13
4.1 Introduction	13
4.2 Motivation for ENI.....	13
4.3 Benefits of ENI.....	14
4.4 High-Level Description of the ENI System Architecture.....	15
4.4.1 Overall Description.....	15
4.4.2 The Assisted System.....	15
4.4.2.1 Introduction.....	15
4.4.2.2 Class 1: An Assisted System that has No AI-based Capabilities	16
4.4.2.3 Class 2: An Assisted System with AI that is Not in the Control Loop.....	17
4.4.2.4 Class 3: An Assisted System with AI Capabilities in its Control Loop	18
4.4.2.4.1 Introduction	18
4.4.2.4.2 Class 3 Options.....	18
4.4.2.5 Summary of Interaction between the Assisted System and ENI.....	19
4.4.3 Mode of Operation.....	20
4.4.3.1 Allowed Modes of Operation.....	20
4.4.3.2 Setting the Mode of Operation	20
4.4.3.3 Interaction with the Assisted System	21
4.4.3.4 Selecting a Mode of Operation for a Class of Decisions.....	21
4.4.3.5 Communication of Mode of Operation	21
4.4.3.6 Normal Operation of the Selected Mode of Operation.....	21
4.4.3.7 Exception Handling for the Selected Mode of Operation	22
4.4.4 Communication.....	22
4.4.4.1 Overview	22
4.4.4.2 Discovery	22
4.4.4.3 Direct Configuration	22
4.4.4.4 Negotiation.....	22
4.4.4.5 Switching the Mode of Operation	23
4.4.4.5.1 Overview	23
4.4.4.5.2 Case 1: ENI Indirectly Instructs the Assisted System to Switch Modes.....	24
4.4.4.5.3 Case 2: ENI Directly Instructs the Assisted System to Switch Modes.....	24
4.4.5 Functional Concepts	24
4.4.5.1 Introduction.....	24
4.4.5.2 Functional Block Definitions	24
4.4.5.3 State.....	24
4.4.5.4 Control Loop Operation	25
4.4.5.5 Inferencing	25
4.4.5.6 Data, Information, Knowledge, and Wisdom.....	25
4.4.5.6.1 Introduction	25
4.4.5.6.2 Data	27

4.4.5.6.3	Information	27
4.4.5.6.4	Knowledge.....	27
4.4.5.6.5	Wisdom	27
4.4.5.7	Measured vs. Inferred Knowledge	27
4.4.6	ENI Reference Points.....	28
4.4.6.1	Definition of an ENI Reference Point	28
4.4.6.2	Definition of an ENI External Reference Point.....	28
4.4.6.3	Definition of an ENI Internal Reference Point.....	28
4.4.7	ENI Interfaces	28
4.4.7.1	Definition of an ENI Interface	28
4.4.7.2	Definition of an ENI Hardware Interface	28
4.4.7.3	Definition of an ENI Software Interface	28
4.4.7.4	Definition of an ENI Application Programming Interface	28
4.4.7.5	Comparison of ENI Software Interfaces with ENI APIs.....	29
4.5	Functional Architecture	29
4.5.1	Functional Block Diagram of the ENI System	29
4.5.2	API Broker.....	31
4.5.3	ENI System Functional Blocks	31
4.5.3.1	Introduction	31
4.5.3.2	Input Processing	31
4.5.3.2.1	Overview	31
4.5.3.2.2	Data Ingestion Functional Block	31
4.5.3.2.3	Normalization Functional Block	32
4.5.3.3	Analysis.....	32
4.5.3.3.1	Knowledge Management and Processing.....	32
4.5.3.4	Situation-based, Model-driven, Policy Generation	33
4.5.3.4.1	Overview	33
4.5.3.4.2	Situation Awareness Functional Block	33
4.5.3.4.3	Model Driven Engineering Functional Block.....	33
4.5.3.5	Output Generation.....	34
4.5.3.5.1	Overview	34
4.5.3.5.2	Denormalization Functional Block	34
4.5.3.5.3	Output Generation Functional Block	35
4.5.4	Decision-Making	35
4.5.4.1	Overview	35
4.5.4.2	Decision-Making using Hindsight	35
4.5.4.3	Decision-Making using Deterministic Processing	35
4.5.4.4	Decision-Making using Predictive Processing.....	35
4.5.4.5	Decision-Making using Cognitive Processing	36
5	ENI Architectural Requirements	36
5.1	Introduction	36
5.2	Functional Architectural Requirements for ENI Operation.....	36
5.3	Architectural Requirements for Mode of Operation.....	39
5.4	Non-Functional Architectural Requirements for ENI Operation.....	40
5.5	Reference Point Requirements	40
6	ENI Reference Architectural Framework.....	41
6.1	Introduction	41
6.2	Design Principles of the ENI System architecture	41
6.2.1	Overview	41
6.2.2	Nesting of Functional Blocks.....	42
6.2.3	Negotiation	43
6.2.3.1	Introduction.....	43
6.2.3.2	Distributive Negotiation.....	43
6.2.3.3	Integrative Negotiation.....	43
6.2.3.4	Functional Model: an Informative Example	44
6.3	Architectural Functional Blocks of the ENI System	44
6.3.1	Introduction.....	44
6.3.2	Data Ingestion Functional Block	45
6.3.2.1	Introduction.....	45
6.3.2.2	Motivation.....	45

6.3.2.3	Function of the Data Ingestion Functional Block.....	46
6.3.2.3.1	Introduction	46
6.3.2.3.2	Data Filtering.....	46
6.3.2.3.3	Data Correlation	46
6.3.2.3.4	Data Cleansing.....	47
6.3.2.3.5	Data Anonymization and Pseudonymization.....	47
6.3.2.3.6	Data Augmentation.....	47
6.3.2.3.7	Data Labelling	47
6.3.2.4	Operation of the Data Ingestion Functional Block.....	47
6.3.3	Data Normalization Functional Block	48
6.3.3.1	Introduction.....	48
6.3.3.2	Motivation.....	49
6.3.3.3	Function of the Data Normalization Functional Block	49
6.3.3.4	Operation of the Data Normalization Functional Block.....	49
6.3.3.4.1	Introduction	49
6.3.3.4.2	Database Design Analogy	49
6.3.3.4.3	Normalization for Machine Learning	50
6.3.3.4.4	Applying Normalization to ENI	51
6.3.4	Knowledge Management Functional Block.....	52
6.3.4.1	Introduction.....	52
6.3.4.2	Inferencing	52
6.3.4.3	Motivation.....	52
6.3.4.4	Function of the Knowledge Management Functional Block.....	53
6.3.4.5	Operation of the Knowledge Management Functional Block.....	53
6.3.4.5.1	Introduction	53
6.3.4.5.2	Observe Functionality	54
6.3.4.5.3	Orient Functionality.....	54
6.3.4.5.4	Decide Functionality	55
6.3.4.5.5	Act Functionality	55
6.3.5	Context-Aware Management Functional Block.....	55
6.3.5.1	Introduction.....	55
6.3.5.2	Motivation.....	56
6.3.5.3	Function of Context Awareness	56
6.3.5.4	Operation of the Context Awareness Functional Block	56
6.3.6	Cognition Framework Functional Block.....	58
6.3.6.1	Introduction.....	58
6.3.6.2	Motivation.....	59
6.3.6.3	Function of the Cognition Framework Functional Block.....	59
6.3.6.4	Operation of the Cognition Framework Functional Block (informative).....	59
6.3.6.4.1	Introduction	59
6.3.6.4.2	The Symbolic Approach.....	60
6.3.6.4.3	The Connectionist Approach	61
6.3.7	Situational Awareness Functional Block	62
6.3.7.1	Introduction.....	62
6.3.7.2	Motivation.....	62
6.3.7.3	Function of Situational Awareness	62
6.3.7.4	Operation of the Situational Awareness Functional Block	62
6.3.7.5	Difference between Context Awareness and Situational Awareness.....	65
6.3.8	Model Driven Engineering Functional Block.....	65
6.3.8.1	Introduction.....	65
6.3.8.2	Motivation.....	65
6.3.8.3	Function of the Model Driven Engineering Functional Block	66
6.3.8.4	Operation of the Model Driven Engineering Functional Block	66
6.3.9	Policy Management Functional Block.....	68
6.3.9.1	Introduction.....	68
6.3.9.2	Motivation.....	68
6.3.9.3	Function of the Policy Management Functional Block.....	69
6.3.9.4	Operation of the Policy Management Functional Block	70
6.3.10	Denormalization Functional Block.....	72
6.3.10.1	Introduction.....	72
6.3.10.2	Motivation.....	73
6.3.10.3	Function of the Denormalization Functional Block.....	73

6.3.10.4	Operation of the Denormalization Functional Block	73
6.3.11	Output Generation Functional Block	74
6.3.11.1	Introduction	74
6.3.11.2	Motivation	75
6.3.11.3	Function of the Output Generation Functional Block	75
6.3.11.4	Operation of the Output Generation Functional Block	75
6.4	API Broker	76
7	Reference Points	76
7.1	Introduction	76
7.2	Reference Point Overview	76
7.3	Reference Point Definitions	79
7.3.1	Reference Point E _{oss-eni-dat}	79
7.3.2	Reference Point E _{oss-eni-cmd}	79
7.3.3	Reference Point E _{app-eni-ctx}	79
7.3.4	Reference Point E _{app-eni-oth}	80
7.3.5	Reference Point E _{app-eni-kno}	80
7.3.6	Reference Point E _{bss-eni-dat}	80
7.3.7	Reference Point E _{bss-eni-cmd}	81
7.3.8	Reference Point E _{usr-eni-pol}	81
7.3.9	Reference Point E _{or-eni-dat}	81
7.3.10	Reference Point E _{or-eni-cmd}	81
7.3.11	Reference Point E _{inf-eni-dat}	82
7.3.12	Reference Point E _{inf-eni-cmd}	82
8	Interacting with Other Standardized Architectures	82
8.1	Introduction	82
8.2	Generic Architecture	83
8.3	Generic SDO Interaction Architecture	84
8.3.1	Introduction	84
8.4	Interaction with NFV MANO	84
8.4.1	High Level description of the NFV MANO - ENI Interaction	84
8.4.2	Initial proposals for interaction scenarios	86
8.4.2.1	Introduction	86
8.4.2.2	Scenario 1: Passive Notification to NFV MANO	86
8.4.2.3	Scenario 2: Active Data Analysis for NFV MANO	86
8.4.2.4	Scenario 3: Active Assistance to the NFV MANO System	86
8.4.2.5	Scenario 4: Active Assistance to the Assisted System	86
9	Areas for Future Study	87
9.1	Introduction	87
9.2	Future Study for the Assisted System	87
9.3	Models	87
9.3.1	Future Study for the Information Model	87
9.3.2	Future Study for Data Models	88
9.4	Input and Output Processing	88
9.4.1	Future Study for Data Ingestion and Normalization	88
9.4.2	Future Study for Denormalization and Output Generation	88
9.5	Knowledge Processing	88
9.5.1	Future Study for Knowledge Management	88
9.5.2	Future Study for Context Management	89
9.5.3	Future Study for Cognition Management	89
9.5.4	Future Study for Situation Awareness	89
9.6	Future Study for MDE Processing	89
9.7	Future Study for Policy Management and Processing	89
9.8	Future Study for Reference Points	90
9.9	Future Study for Interfaces	90
9.10	Future Study for APIs and the API Broker	90
9.11	Future Study for New ENI Functional Blocks	91
9.12	Future Study for Interaction with Other Organizations	91
9.13	Future Study for Security	91
Annex A (informative):	SDO and Open Source Interactions	93

A.1	Integration with Other SDOs and Open Source Communities.....	93
A.1.1	Introduction	93
A.1.2	Interaction with MEF-LSO.....	93
A.1.3	Interaction with BBF CloudCO.....	93
A.2	Interaction with Open Source Communities	95
Annex B (informative):	ENI Architectural Evolution.....	96
B.1	ENI Architecture Evolution Motivation.....	96
B.2	ENI Architecture Evolution Proposal.....	96
B.3	Proposed Definition of the ENI Phases	96
Annex C (informative):	Authors & contributors.....	97
History		98

iTeh STANDARD PREVIEW
 (standards.iteh.ai)
 Full standard:
<https://standards.iteh.ai/catalog/standards/sist/46302cf-5a12-4912-9a9e-c18b8fd078fb/etsi-gs-eni-005-v1.1.1-2019-09>

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

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

The present document specifies a high-level functional abstraction of the ENI System Architecture in terms of Functional Blocks and External Reference Points. This includes describing how different classes of systems interact with ENI. Processes, models, and detailed information are beyond the scope of the present document.

Introduction

The present document defines a high-level functional abstraction of the ENI System Architecture. The organization of the present document is as follows. Clause 1 defines the scope of the present document. Clauses 2 and 3 provide normative and informative references and definition of terms, respectively. Clause 4 provides an informative overview of the ENI System Architecture, including its motivation, benefits, important concepts, and an overview of its Functional Blocks. Clause 5 lists requirements of the ENI System Architecture. Clause 6 defines important design principles of the ENI System Architecture, and then specifies the different Functional Blocks that make up the ENI System Architecture. Clause 7 specifies the External Reference Points of the ENI System Architecture. Clause 8 describes how ENI interacts with other SDO Systems and clause 9 delineates a list of future study items.

1 Scope

The present document specifies the functional architecture of an ENI System, which is a high-level decomposition of an ENI System into its major components, along with a characterization of the externally visible behaviour (e.g. as defined by a set of reference points) of the components. This includes:

- defining the functionality and behaviour of a system that satisfy the ENI Requirements (ETSI GS ENI 002 [6]);
- defining a functional architecture, in terms of Functional Blocks, that addresses the goals specified by the ENI Use Cases (ETSI GS ENI 001 [5]);
- defining Reference Points used by the above Functional Blocks for all communication with systems and entities that are external to the ENI System;
- proposing a progression plan towards full support of the proposed ENI System and intermediary level of compliance (e.g. support of some architecture components or a subset of the Reference Points).

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] OMG formal/2015-06-03: "OMG Systems Modeling Language".
- [2] BBF TR-384: "Cloud Central Office Reference Architectural Framework", January 2018, G. Karagiannis, D. Hai.
- [3] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".
- [4] IETF RFC 4949: "Internet Security Glossary, Version 2", Shirey, R., August 2007.
- [5] ETSI GS ENI 001 (V2.1.1): "Experiential Networked Intelligence (ENI); ENI use cases".
- [6] ETSI GS ENI 002 (V2.1.1): "Experiential Networked Intelligence (ENI); ENI requirements".
- [7] MEF 78: "MEF Technical Specification: MEF Core Model", Strassner, J., editor, January 2019.
- [8] Strassner, J., Agoulmine, N., Lehtihet, E.: "FOCALE - A Novel Autonomic Networking Architecture", ITSSA Journal 3(1), 64-79, 2007.
- [9] Boyd, J. R.: "The Essence of Winning and Losing", June, 1995.
- [10] Strassner, J.C.: "Knowledge Representation, Processing, and Governance in the FOCALE Autonomic Architecture", book chapter, 2011, Elsevier.
- [11] NIST: "Role-Based Access Control, Second Edition".

NOTE: Available at <https://us.artechhouse.com/Role-Based-Access-Control-Second-Edition-P1386.aspx>.

[12] NIST: "Attribute Based Access Control".

NOTE: Available at <https://csrc.nist.gov/publications/detail/book/2017/attribute-based-access-control>.

[13] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3", August 2018.

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] Strassner, J.: "Policy-Based Network Management", Morgan Kaufman, ISBN 978-1558608597, September 2003.
- [i.2] Strassner, J., de Souza, J.N., Raymer, D., Samudrala, S., Davy, S., Barrett, K.: "The Design of a Novel Context-Aware Policy Model to Support Machine-Based Learning and Reasoning", Journal of Cluster Computing, Vol 12, Issue 1, pages 17-43, March, 2009.
- [i.3] ETSI GR ENI 003 (V1.1.1): "Experiential Networked Intelligence (ENI); Context-Aware Policy Management Gap Analysis".
- [i.4] Strassner, J., Betser, J., Ewart, R., Belz, F.: "A Semantic Architecture for Enhanced Cyber Situational Awareness", Secure and Resilient Cyber Architectures Conference, MITRE, 2010.
- [i.5] Gamma, E., Helm, R., Johnson, R., Vlissides, J.: "Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley, Nov., 1994. ISBN 978-0201633610.
- [i.6] Bäumer, D., Riehle, W., Siberski, M., Wulf: "The Role Object Pattern", Proceedings of the 1997 Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA '97), ACM Press, 1997, Page 218-228.
- [i.7] Rowley, J.: "The wisdom hierarchy: representations of the DIKW hierarchy", Journal of Information and Communication Science, 33(2): 163-180.
- [i.8] Chin, K.O., Ganb, K.S., Alfred, R., Anthony, P, and Lukose, D.: "Agent Architecture: An Overview", Transactions on Science and Technology, vol 1, No 1, pp 18-35, 2014.
- [i.9] Shehory, O and Sturm, A., editors: "Agent-Oriented Software Engineering", Springer, 2014.
- [i.10] Martin, R. C.: "Agile Software Development, Principles, Patterns, and Practices", Prentice Hall, 2003 ISBN 978-0135974445.
- [i.11] Ritter, F.E., Tehranchi, F., Oury, J.D.: "ACT-R: A Cognitive Architecture for Modeling Cognition", Wiley Interdisciplinary Reviews, Cognitive Science 10(4): e1488.

NOTE: Available at <http://act-r.psy.cmu.edu>.

- [i.12] IETF RFC 8328: "Policy-Based Management Framework for the Simplified Use of Policy Abstractions (SUPA)", Liu, W., Xie, C., Strassner, J., Karagiannis, G., Klyus, M., Bi, J., Cheng, Y., and D. Zhang .

NOTE: Available at <https://www.rfc-editor.org/info/rfc8328>.

- [i.13] Rothenberg, J.: "The Nature of Modelling", Artificial Intelligence, Simulation, and Modeling, John Wiley and Sons, Inc., 1989, pp. 75-92.

NOTE: Available at <https://www.rand.org/content/dam/rand/pubs/notes/2007/N3027.pdf>.

- [i.14] Recommendation ITU-T 9594-1: "Information Technology - Open Systems Interconnection - The Directory: Overview of Concepts, Models, and Services".
- [i.15] Recommendation ITU-T 9594-7: "Information Technology - Open Systems Interconnection - The Directory: Selected Object Classes".
- [i.16] ETSI GR ENI 003 (V1.1.1): "Experiential Networked Intelligence (ENI); Context-Aware Policy Management Gap Analysis".
- [i.17] MEF Technical Specification: "Policy-Driven Orchestration", Call for Comments v0.7, August 2017, Strassner, J., editor.

NOTE: Available at <https://wiki.mef.net/download/attachments/59846378/MEF%20PDO%20CfCB1.zip?api=v2>.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

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

API: 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

NOTE: This is also known as an Application Programming Interface.

API Broker: software entity that mediates between two systems with different APIs, enabling the two different systems to communicate transparently with each other

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

- **cognitive architecture:** 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:** symbolic world model that enables problem-solving components to be built using a sense-plan-act paradigm
- **hybrid architecture:** system made up of reactive and deliberative components that are combined into a hierarchy of interacting layers, where each layer reasons at a different level of abstraction
- **reactive architecture:** system that is aware of changes that affect its computations and adjusts accordingly

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:** 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"

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

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

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

designated entity: operator, nms, ems, controller, or orchestrator acting on behalf of the assisted system

NOTE: The Designated Entity is a trusted entity [4].

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

formal: study of (typically linguistic) meaning of an object by constructing formal mathematical models of that object and its attributes and relationships

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
- **propositional knowledge:** knowledge of a proposition, along with a set of conditions that are individually necessary and jointly sufficient to prove (or disprove) the proposition

location: physical geographic location (e.g. a geocode or a bounding polygon) of an entity (e.g. a server)

NOTE: Contrast this with Placement.

Model-Driven Behaviour (MDB): approach in which the behaviour of components, modules of systems are managed using MDE. For ENI, this applies to Functional Blocks, not components within a Functional Block

Model-Driven Engineering (MDE): approach in which models are central to all phases of the development and implementation processes

negotiation: set of communications that is intended to reach a beneficial outcome for a set of conflicting issues

- **distributive negotiation:** zero-sum game, in which each participant assumes that there is a fixed amount of value to be divided between the (winning) bidders
- **integrative negotiation:** win-win (or non-zero-sum) game, in which all collaborating participants receive optimal value

placement: logical placement of an entity (e.g. a virtual machine) on or in another entity (e.g. a server).

NOTE: Contrast this with Location.

repository: centralized location of a set of storage devices that enable different functional blocks to store and retrieve information

- **active repository:** repository that pre- and/or post-processes information that is stored or retrieved

NOTE: It may contain dedicated (typically internal) Reference Points that provide the loading, activation, deactivation, and unloading of specialized functions that change the pre- and/or post-processing functionality according to the needs of the application.

- **passive repository:** repository that stores or retrieves information without pre- or post-processing

SDO system: part of an Assisted System that is defined by another SDO

NOTE: Examples include NFV MANO and MEF LSO.

semantics: study of the meaning of something (e.g. a sentence or a relationship in a model)

telemetry: process of recording and transmitting data to receiving equipment for monitoring purposes

NOTE: The process is typically automated, and the data transfer may include wireless, cellular, optical, and other mechanisms.

3.2 Symbols

Void.

3.3 Abbreviations

Void.

4 Overview of System Architecture (informative)

4.1 Introduction

This clause provides an informative introduction to the ENI System Architecture. Clauses 4.2 and 4.3 describe the motivation and benefits of ENI. Clause 4.4 provides a high-level description of the ENI System Architecture, including how it interacts with the Assisted System (and/or its Designated Entity), different types of Assisted Systems, the mode of operation that the Assisted System (or its Designated Entity) can choose, direct and indirect communication between the ENI System and the Assisted System (or its Designated Entity), and important concepts used in this System Architecture. Clause 4.5 describes the functional architecture of the ENI System in terms of Functional Blocks. It ends with a discussion of how decision-making is done in the ENI System.

NOTE: See clause 9 for further applicable future work.

4.2 Motivation for ENI

Current network management provisioning and monitoring functions are time-consuming and error-prone. This is due to the proliferation of different technologies, as well as different implementations from different vendors. In addition, users are demanding more complex services (e.g. context-aware, personalized services). Hence, operators are concerned about the increasing complexity of integration of different platforms in their network and operational environment. These human-machine interaction challenges increase the time to market of innovative and advanced services. Moreover, there is no efficient and extensible standards-based mechanism to provide contextually-aware services (e.g. services that adapt to changes in user needs, business goals, or environmental conditions).

These and other factors contribute to a very high Operational EXpenditure (OPEX) for network operation and management. Operators need to optimize the use of networked resources (e.g. through the automation of their network configuration and monitoring processes to reduce this OPEX). More importantly, operators need to improve the use and maintenance of their networks.