ETSI GR ENI 031 V4.1.1 (2024-05)

GROUP REPORT

Experiential Networked Intelligence (ENI); Construction and application of fault maintenance network knowledge graphs

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

DGR/ENI-0031v411_Net-know-grap

Keywords

access, algorithm, wireless ad-hoc network

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 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are 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 Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, 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.

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.

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 purpose of the present document is to describe use cases and a construction method of knowledge graphs that are used for fault maintenance. The present document focus on the following topics:

- defines the data requirements;
- defines a schema design;
- describes the knowledge application interface for fault maintenance knowledge graphs.

In addition to the main target related to the construction of fault maintenance network knowledge graphs, obtaining computer-readable and writable network fault maintenance domain knowledge and enabling learning and reasoning of relevant algorithm models of fault self-repair are also envisaged.

The present document will encompass research and investigation activities that will address wireless networks at the first stage. Subsequent efforts possibly will extend the work into access networks.

NOTE: The general solutions for knowledge graph and knowledge management are out of the scope of the present document.

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 GR ENI 004 (V3.1.1): "Experiential Networked Intelligence (ENI); Terminology".

[i.2] ETSI GS ENI 005 (V3.1.1): "Experiential Networked Intelligence (ENI); System Architecture".

[i.3] ETSI GR ENI 010 (V1.1.1): "Experiential Networked Intelligence (ENI); Evaluation of categories for AI application to Networks".

[i.4] TM Forum IG1218: "Autonomous Networks Business Requirements and Architecture".

[i.5] "Autonomous Systems - An Architectural Characterization, Joseph Sifakis", Verimag Laboratory, 2019.

[i.6] ETSI GS ENI 019 (V3.1.1): "Experiential Networked Intelligence (ENI); Representing, Inferring, and Proving Knowledge in ENI".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR ENI 004 [i.1] , ETSI GS ENI 005 [i.2], ETSI GR ENI 010 [i.3] and the following apply:

co-reference resolution: task of finding all expressions in a text that refer to the same real-world entity

data cleansing: process of detecting and correcting (or removing) incomplete, incorrect, inaccurate, or corrupt data from being further processed

entity disambiguation: process of finding references in a dataset that refer to the same entity

extraction: activity of transforming valuable information into structured data from input data of different sources and structures:

- **attribute extraction:** extract detailed features describing entities/relationships/events from input data
- **entity extraction:** identify named entities from input data
- **knowledge extraction:** extracting entities, relationships, and new knowledge from patterns or inference from data
- **relationship extraction:** extract various semantic associations between different entities from input data

knowledge discovery: finding new knowledge from existing knowledge

knowledge entity: concept of reflecting specific things in the knowledge graph

knowledge fusion: process of combining knowledge from multiple sources to create a more comprehensive and accurate representation of the system

knowledge graph: structured representation of knowledge that uses a graph data structure and formal logic to represent entities and their relationships

knowledge integration: combining knowledge from different sources in a way that preserves the semantics of the knowledge

knowledge reasoning: process of using knowledge to draw new conclusions:

- **graph-based knowledge reasoning:** embed high-dimensional knowledge graphs into low latitude continuous vector spaces

  NOTE: They represent entities and relationships as numerical vectors for computation, and combine other algorithms to achieve reasoning, including methods such as distributed feature representation, Graph Neural Network (GNN) algorithms.

- **rule-based knowledge reasoning:** mining and reasoning by defining or learning rules that exist in knowledge, including methods such as predicate logic reasoning, ontology reasoning, and random reasoning

ontology alignment: finding mappings between overlapping concepts in two or more ontologies to align them

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR ENI 004 [i.1], ETSI GS ENI 005 [i.2] and the following apply:

KaaS	Knowledge as a Service
KGC	Knowledge Graph Completion
KGQA	Knowledge Graph Question Answering
LAN	Local Area Network
MMKG	Multi-Modal Knowledge Graph
MR	Measurement Report
NFV	Network Function Virtualisation
R&D	Research and Development
RCA	Root Cause Analysis
RDF	Resource Description Framework
SDN	Software Defined Networking
SQL	Structured Query Language
WQMKG	Wireless Quality Maintenance Knowledge Graph

4 Introduction

4.1 Background of network fault management knowledge graph construction

The telecommunications industry has been exploring digitization, automation and intelligence, from focusing on customer service and business layer in the early stage of transformation, gradually extending to the internal management and operation layer, and then to the network layer. Initially, the telecommunications industry looked to reduce cost and complexity while increasing business and network agility by using SDN, NFV and cloud technologies. However, network automation based on SDN/NFV technology still cannot completely solve the problems caused by the large-scale deployment of various applications, such as the introduction and expansion of new network technologies in the future. A common problem faced by the industry is how to improve efficiency and throughput in a large-scale for the entire process, and how to continuously, rapidly and iteratively introduce new technologies.

Autonomous Networks (refer to GR ENI 010 [i.3]) was born in this context. By applying a variety of intelligent technologies and giving full play to the integration advantages, it will drive the telecommunications industry to an era where intelligence is embedded in the network. In order to achieve the ultimate goal of autonomous networks (refer to [i.4]), it is essential to transform the existing business models to some new production, business and collaboration models, including eKnowledge-as-a-Service operations model (KaaS). KaaS is about delivering the right knowledge to the right person in the right context at the right time via desktop, laptop or any mobile device.

According to the highly autonomous system architecture proposed by Joseph Sifakis, a Greek/French computer scientist and winner of the Turing Award in 2007, knowledge management is understood to become one of the key technical features of autonomous networks (refer to [i.5]). Constructing the knowledge graph of a telecommunication domain could be an important means to improve the level of network intelligence.

With the rapid development of communication networks in recent years, the number of network connections and network data are growing rapidly, the network structure is becoming more and more complex, and there are more and more faults. The traditional fault processing approach is difficult to handle the increasing number of network faults, which is mainly reflected in the following three aspects:

- Traditional OAMP relies on the experience of experts, but the experience of OAMP experts varies greatly. To accurately derive the root cause of the fault, the required skill level for OAMP personnel is very high.
- Thousands of operation and maintenance personnel deal with a large number of repetitive problems. The accumulation of operation and maintenance experience is slow and the processing efficiency is low.

- Reference documents, such as vendor manuals and lab documents created by operation and maintenance personnel based on their experience, do not have standardized terminology. For example, different vendors may name the same KPI differently, and calculate it using different formulas. This makes it difficult for the operation and maintenance personnel to use.

Due to the existence of these problems, the automatic and intelligent network fault diagnosis technology has attracted the attention of researchers, which are examining how to improve the collection of relevant context-aware telemetry to more rapidly diagnose and repair faults and protect services.

In view of this problem, the present document will describe research work on the application of knowledge graphs in the domain of telecommunication networks, integrating the experience of experts in the domains of telecommunications and information knowledge to build knowledge graphs for the telecommunications domain, and intelligently discover the root cause of faults by using the representation, reasoning and human-computer interaction technology of knowledge graphs. However, the construction of the whole domain knowledge graph of telecommunication network is a huge project that cannot be accomplished in a reasonable time frame (refer to ETSI GS ENI 019 [i.6]). Therefore, taking the wireless access network field as the starting point, the present document puts forward the construction schema of wireless access network fault maintenance knowledge graphs, so as to provide technical reference for promoting the cooperation of industrial ecological partners to jointly build the general ability of fault diagnosis in the telecommunication network domain.

4.2 Values of network fault maintenance knowledge graph

The knowledge graphs play two roles for R&D personnel in the process of fault maintenance:

- Horizontal integration of information. For example, different types of log files collected on site are organized according to the dimensions of entity objects. This makes it easy to perform Named Entity Recognition (NER). NER can be used for physical objects (e.g. a router or a geographic area), logical objects (e.g. a device interface or a VPN), events (e.g. faults and notifications, such as the uploading of a new configuration or that data is available) and other networking concepts, When troubleshooting, the R&D personnel can intuitively see some events (alarms and operations) on the knowledge graph corresponding to the instance of a network element. The knowledge graph can also display the events related to the instance object. After the 5G wireless access network log file is visualized using the knowledge graph, the time for the R&D personnel to analyse the log will be reduced.

- Transform the experience of OAMP expert personnel into a knowledge graph, enabling the knowledge from multiple personnel and reference works to be more easily accumulated, augmented with new information, and applied. This provides the potential to be edited into reasoning rules and reused in each existing network that uses similar technologies and devices.

The power of a knowledge graph is its use of logic to manage and represent complex data by creating interconnected models of entities and their relationships. This provides the ability to reason about ingested telemetry and prove its conclusions. This in turn helps improve the operator's experience.

NOTE: The reasoning rules transformed from expert experience in the knowledge graphs are for further study.

5 Construction of wireless network fault maintenance knowledge graphs

5.1 Framework of network fault maintenance knowledge graph construction process

The framework of the network fault maintenance knowledge graph construction process includes four steps: data acquisition and processing, ontology/schema design, knowledge extraction and knowledge fusion, knowledge verification. The main process of building the network fault maintenance knowledge graph is shown in Figure 5-1.

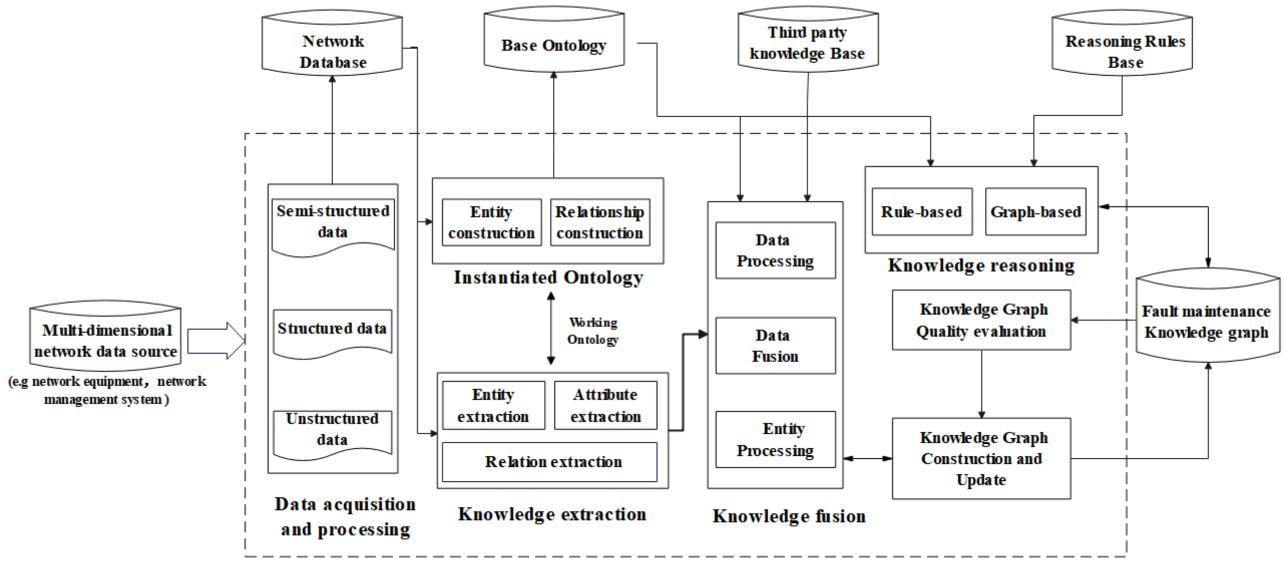

Figure 5-1: The main process of the network fault maintenance knowledge graph construction

An overview of building the fault maintenance knowledge graph is described below.

1) Data acquisition and processing

For data of interest, such as log and alarm data provided by the network, the system will regularly collect data, process the data (e.g. perform tasks such as data filtering, correlation, cleansing, and deduplication) and normalize the collected data(refer to [i.5]). After processing, the data is stored in the Network Database for further processing.

2) Ontology design and update

The Base Ontology is a set of ontologies that contain a starting ontology to work with. For example, ontologies defining information about network entities, people, and other relevant concepts could be used. These are then combined into a single starting ontology.

The Instantiated Ontology is used for two purposes. First, data and information that has been ingested and processed in step 1 is incorporated directly in the Instantiated Ontology if it is in a suitable form. If not, then it is first processed in the Working Ontology, which converts the data into entities, attributes, and relationships between entities, possibly using the information and/or data models defined in ETSI GS ENI 019 [i.6] as a guide. These entities, attributes, and relationships are then uploaded to the Instantiated Ontology, which verifies and integrates them, and uploads a new version of the Ontology to the Base Ontology and also to the Working Ontology. Once this process is complete, the Working Ontology is then used for further processing. This keeps a backup in the Instantiated Ontology in case of problems in further processing.

3) Knowledge extraction and knowledge fusion

Knowledge extraction is the secondary processing of data in the database, aiming to extract knowledge entities (i.e. the entities in the instantiated and working ontologies) and their relationships in the data, and add processed information to the knowledge graph.

The main tasks of knowledge fusion include three categories of measurements. Exemplary processing tasks of each category are provided.

- Data Processing
  - Tokenization transforms the text into smaller units called tokens. Tokens can be words, punctuation marks, numbers, symbols, or any other meaningful elements of a text. This is used as the basis for many natural language processing tasks.
  - Lemmatization is a process of finding the base or root form of a word. For example, the words "running", "runs", and "ran" are all different forms of the word "run". This is used to normalize words for more complex tasks, such as named entity recognition and co-reference resolution.

  - Data Annotation provides additional information to better describe the datum and its purpose and functionality. This is typically in the form of metadata attached to the datum.

- Entity Processing
  - Named Entity Resolution is a process in Natural Language Processing that involves recognizing when two observations relate semantically to the same entity, despite possibly having been described differently. It also involves recognizing when two observations do not relate to the same entity, despite having been described similarly. Named Entity Resolution helps in standardizing entities and improving the accuracy of information extraction from text data.
  - Relation extraction is the task of identifying the relationships between entities. This is essential for understanding the meaning of the data.
  - Co-reference resolution is used to resolve ambiguities when extending knowledge graphs with new facts. It also aids in semantic integration, helping to model semantic relationships between different structures so that a coherent global view can be obtained.

- Data Fusion
  - Data Alignment is a preprocessing step that ensures the collected data from different sources is in a format that can be effectively combined or fused. This process often involves the temporal alignment of multimedia data.
  - Semantic Processing determines the meaning of the data elements, the relationships between the data elements, and the context in which the data is collected.
  - Semantic data integration is the process of combining data from disparate sources into a single, unified dataset in a way that preserves the meaning of the data. This is done by understanding the semantics of the data, which includes the meaning of the data elements, the relationships between the data elements, and the context in which the data is collected.
  - Semantic Data Fusion combines the different data into a single unified dataset with common semantics. There are a number of different approaches for achieving this, including:
    - Feature-level fusion: This approach involves combining features from different data sources at the feature level. This can be done using a variety of techniques, such as averaging, weighted averaging, and principal component analysis.
    - Decision-level fusion: This approach involves making separate decisions based on each data source and then combining the decisions. This can be done using a variety of techniques, such as majority voting, weighted voting, and Dempster-Shafer theory.
    - Model-level fusion: This approach involves combining the models from different data sources to create a single, unified model. This can be done using a variety of techniques, such as ensemble learning and Bayesian inference.
    - Knowledge-based Fusion: This approach uses knowledge about the domain to fuse data. Knowledge-based fusion can be used to fuse data from sources that are not directly compatible. For example, a knowledge-based fusion system could be used to fuse data from a database and a set of expert rules to create a more complete and accurate view of a system.

NOTE: It is recommended that multiple fusion approaches be used to provide more accurate results and ensure that varying semantics from each data source are taken into account.

  - Define Data Fusion Rules: this defines a set of rules for choosing among the above approaches as well as handling conflicts and inconsistencies that arise during the fusion process.

4) Knowledge graph quality evaluation

Knowledge graph quality evaluation is the process of measuring the accuracy, completeness, reliability, and consistency of knowledge graphs. High-quality knowledge graphs are crucial for various intelligent applications and data analysis tasks, as they rely on accurate and reliable data. The evaluation typically includes multiple dimensions, such as correctness (accuracy of information), completeness (comprehensiveness of information), consistency (uniformity of information across different parts), timeliness (frequency of information updates), and credibility (reliability of information sources).

5) Knowledge graph construction and update

Knowledge graphs and ontologies have different goals, which reflect their different organization of knowledge. The typical way to overcome this problem is to enrich and refactor the ontology to make it suitable for constructing a knowledge graph. There are several ways to do this. One way consists of the following five steps:

1) Extract the concepts from the ontologies using either an ontology parser or NLP techniques.

2) Extract the relationships from the ontologies using the same method as in the previous step.

3) Create a schema for the knowledge graph. The schema should define nodes and edges, and optionally, properties for each.

4) Populate the knowledge graph with data from the ontologies. This can be done using a variety of techniques, such as graph databases and SPARQL (or similar) queries.

5) Validate the correctness of the knowledge graph.

5.2 Data acquisition and processing for knowledge graph

5.2.1 Content and format of data collection

There are three types of data available for building knowledge graph: structured data, unstructured data, and semi-structured data. Structured data is data that is organized in a predefined format, such as a table or database. Unstructured data is data that does not have a predefined format, such as user intent and fault cases. Semi-structured data is data that has some structure, but it is not as rigid as structured data. Table 5-1 shows the specific data format and source. In all cases, data is collected using ENI APIs from appropriate ENI External Reference Points (see ETSI GS ENI 005 [i.2], clause 7).

Table 5-1: Data format and source

Name	Data type	Data sources
Equipment alarm data	Structured data	Network telemetry
Operation and maintenance log	Semi-Structured data	Network telemetry
Expert handbook	Semi-structured data	Manual
Equipment manual	Semi-structured data	Manual
Operation and maintenance database	Structured data	Open source
Operation and maintenance data table from website	Semi-structured data	Manual
User intent	Unstructured data	End user or application
Failure case	Unstructured data	Manual or Open source//Third party database
Device profile	Structured data	UE and Base Station
Blogs from website	Unstructured data	Manual or Open source//Third party database

5.2.2 Data normalization

In all cases, data normalization is realized by using ENI APIs from appropriate ENI External Reference Points (see ETSI GS ENI 005 [i.2], clause 7).

5.3 Ontology design

5.3.1 Diagram of ontology

Ontology design is a set of specifications necessary for knowledge graph data production, which is used to describe the structure of normalized data. Figure 5-2 is a partial schematic diagram of the ontology/schema design of the network fault maintenance knowledge graph, where the yellow block represents the type of entity, the grey block represents the attribute of entity, cyan blocks represent the relationship between the entities. Blue lines indicate the properties of the entities, and purple lines represent subclasses. When a new fault event is generated, the technician performs processing by executing the appropriate troubleshooting process. Each Technician, Antenna and device belong to different cells. In a 5G network, a base station can cover multiple cells, and each cell is associated with one base station. Troubleshootings solve multiple root causes, and different root causes are located on different devices. For example, a faulty access point, misconfiguration of a UE, and interference from other devices in the cell might all be part of the same performance issue.

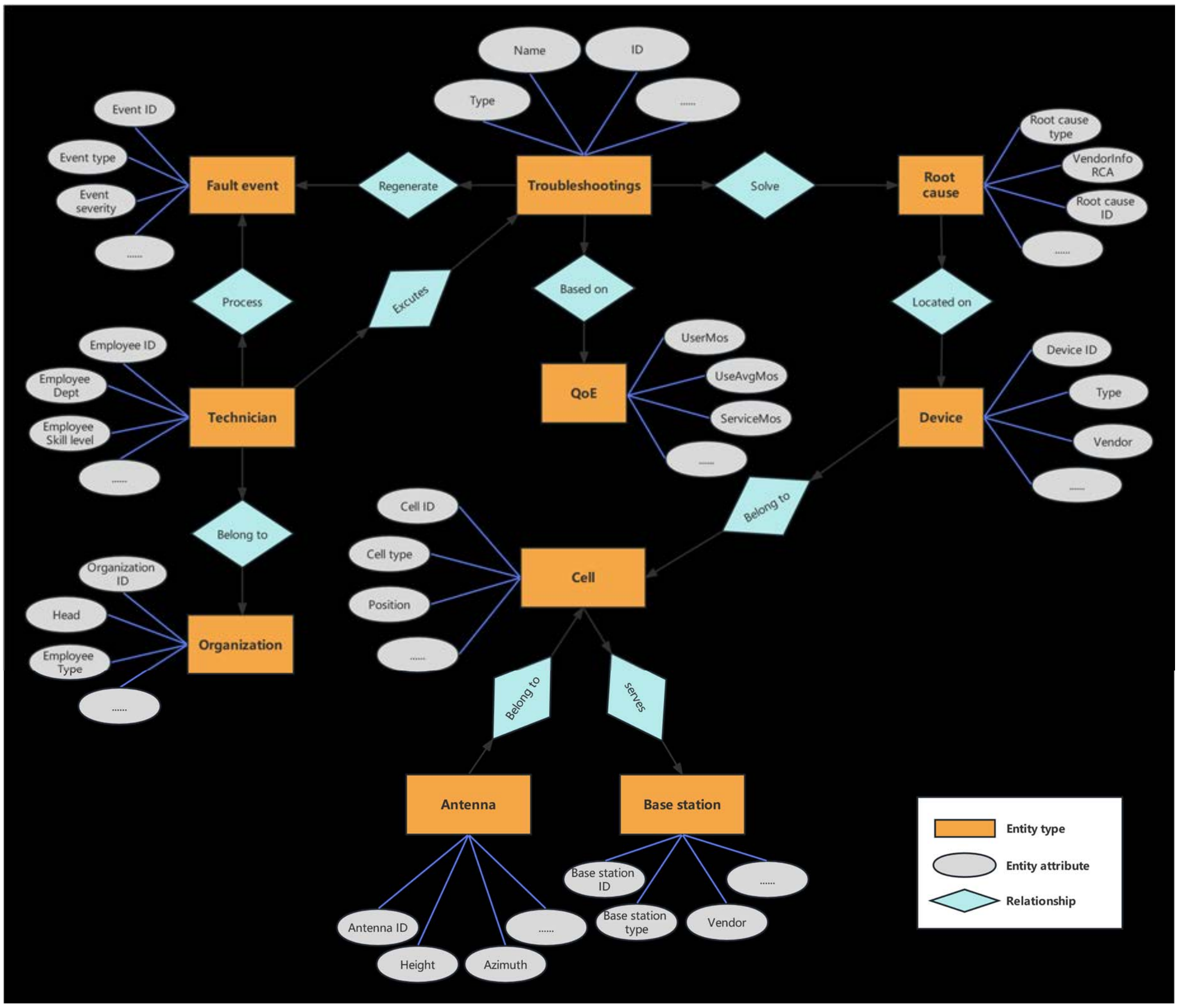

Figure 5-2: Ontology design for the network fault maintenance knowledge graph

5.3.2 Definition of types and attributes of wireless fault maintenance knowledge entities

Entities are the main units that constitute the knowledge graph. The knowledge entities maintained by wireless network and their attributes are as follows:

- **Fault event**

Attribute	Data type	Explain
eventID	TimeAndDate	Unique ID for this event (it disambiguates this event from all other events, even of the same type)
eventType	Enum	Type of event (e.g. link up/down, service unavailable)
eventSeverity	Enum	{critical, major, minor, informational}
eventStatus	Enum	{open, in_progress, closed, etc.}
eventDescription	String	Optional attribute providing information about the event
workOrderID	String	ID of the job to distinguish different work order
Location	String	Look at the Models GS
eventTimeBegin	Time	Start time of the job
eventTimeEnd	Time	End time of the job

- **Cell**

Attribution	Data type	Explain
cellID	String	The ID of the cell to distinguish different cells
position	String	The physical location of the cell
cellType	Enum	The type of cell (e.g. macro, micro, pico)
cellFrequencyBand	String	The frequency band that the cell is operating in
cellStatus	Enum	Current status of cell (e.g. operational, deployed_but_not_operational, in_progress, planned)
cellUsers	Integer	Number of active users of cell
cellErrors	Integer	Number of errors in a given measurement time period
cellPerformance	Custom	Data structure that contains the throughput, latency, jitter, packet loss, and other key metrics
cellRange	String	The maximum distance that the cell can support
cellCapacity	Integer	The maximum number of users that the cell can support
cellTrafficLoad	String	The current traffic load of the cell
cellSectors	Integer	The number of sectors that the cell is divided into
cellBeamformingEnabled	Boolean	TRUE if the cell is using beamforming
cellMimoEnabled	Boolean	TRUE if the cell is using MIMO to transmit and receive multiple data streams simultaneously
cellModulationScheme	String	The modulation scheme that the cell is using
cellCodingSceme	String	The coding scheme that the cell is using
cellOutputPowerMax	String	The maximum output power of the cell
cellOutputPowerCurrent	String	The current output power of the cell

- **Base station**

Attribution	Data type	Explain
baseStationID	String	ID of base station
baseStationType	Enum	The type of base station, such as macro base station, micro base station, or pico base station
vendor	String	The responsible manufacturer of the base station
bsFrequencyBands	String	The frequency bands that the base station is supporting
baseStationStatus	Enum	Current status of the base station (e.g. operational, deployed_but_not_operational, in_progress, planned)
baseStationUsers	Integer	Number of active users of base station
baseStationErrors	Integer	Number of errors in a given measurement time period
baseStationPerformance	Custom	Data structure that contains the throughput, latency, jitter, packet loss, and other key metrics
baseStationRange	String	The maximum distance that the base station can support
baseStationCapacity	Integer	The maximum number of users that the base station can support
baseStationTrafficLoad	String	The current traffic load of the base station
baseStationSectors	Integer	The number of sectors that the base station is divided into
bsBeamformingEnabled	Boolean	TRUE if the base station is using beamforming
bsMimoEnabled	Boolean	TRUE if the base station is using MIMO to transmit and receive multiple data streams simultaneously
bsModulationScheme	String	The modulation scheme that the base station is using
bsCodingSceme	String	The coding scheme that the base station is using
bsOutputPowerMax	String	The maximum output power of the base station
bsOutputPowerCurrent	String	The current output power of the base station

- **Antenna**

Attribution	Data type	Explain
Antenna ID	String	The ID of Antenna, unambiguously distinguishing different Antennas, even if they are of the same type and made by the same vendor
height	Int	The height of Antenna, in meters
tiltAngle	Int	Antenna's tilt angle in degrees
azimuth	Int	The direction in which the Antenna is pointing, in degrees
antennaType	ENUM	The type of antenna, such as omnidirectional, directional, or sector
antfrequencyBands	String	The frequency bands that the antenna is supporting
antennaeamwidth	Integer	The beamwidth of the antenna, which is the angle over which the antenna transmits its signal
antennaGain	Integer	The gain of the antenna, which is a measure of how much the antenna amplifies the signal
antennaVSWR	Integer	The voltage standing wave ratio, which is a measure of how well the antenna is matched to the transmission line
antennaPolarization	Integer	The polarization of the antenna, which is the direction in which the antenna's electromagnetic waves oscillate
antennaPowerMax	Integer	The maximum power that the antenna can handle
antennaPowerCurrent	Integer	The current power that the antenna is handling

- **Technician**

Attribution	Data type	Explain
employeeFirstName	String	The first name of the employee
employeeLastName	String	The last name of the employee
employeeID	String	The ID of the employee to uniquely distinguish different persons
employeeDept	String	Employee's department
employeeSkill Level	Int	The technical level of employees is 0-9, and 9 is the highest
employeeContactInfo	Custom	Custom datatype that includes primary and secondary cell phones, office phone, and emails
employeeRole	RoleObject	Employee's role in the organization (e.g. triage lead, manager, etc.)

- **Organization**

Attribution	Data type	Explain
organizationID	String	The ID of the organization to uniquely distinguish different organizations
head	String	The head of the organization
employeeType	String	The type of employees in the organization
Name	String	The name of the organization

- **QoE**

Attribution	Data type	Explain
userMOS	Integer	Mean Opinion Score (range is 1-5) for an individual user
userAvgMOS	Integer	MOS for all users in a given measurement interval
userMOSVariance	Float	The variance of the MoS scores across users. This can be used to measure the consistency of the user QoE
serviceMOS	Integer	MOS for an individual service
serviceAvgMOS	Integer	MOS for all services in a given measurement interval
serviceMOSVariance	Float	The variance of the MoS scores across services. This can be used to measure the consistency of the service QoE
deviceMOS	Integer	MOS for an individual device
deviceAvgMOS	Integer	MOS for all devices in a given measurement interval
deviceMOSVariance	Float	The variance of the MoS scores across device. This can be used to measure the consistency of the deviceQoE.
locationMOS	Integer	MOS for an individual location
locationAvgMOS	Integer	MOS for all locations in a given measurement interval
locationMOSVariance	Float	The variance of the MoS scores across locations. This can be used to measure the consistency of the location QoE.

- **Root cause**

Attribution	Data type	Explain
rootCauseID	String	Caused to unambiguously identity this root cause object
rootCauseType	ENUM	Type of root cause (e.g. hardware or software error)
rootCauseAffectedComponent	Array of Component IDs	The set of component IDS of the 5G system that were affected by the event
vendorInfoRCA	Array of Strings	This is an array of Strings that contain a 2-tuple: the name of each vendor's component affected, along with the component of that vendor that had an error
componeInfoRCA	Array of custom objects	This is an array of custom data structures that identify the pertinent details of the failed component (e.g. vendor name, partNumber, serialNumber, software version, and other pertinent info)

- **Troubleshootings**

Attribution	Data type	Explain
Type	String	Type of troubleshootings
Name	String	Name of troubleshootings
ID	String	troubleshoots ID is used to distinguish different troubleshootings
troubleshootingsTimePropose	TimeAndDate	the datae and time that this troubleshooting was proposed as completed (it is still subject to subsequent verification)
troubleshootingsTimeVerify	TimeAndDate	The datae and time that this troubleshooting was verified as completed
troubleshootingsEvents	Array of EventIDs	The set of event IDs that are used to verify that this troubleshooting has been completed

5.3.3 Definition of relationship of wireless fault maintenance knowledge entities

Relationship is the way of association between entities. It helps each entity to form a diagram by points, reflecting the logical relationship between entities in the real world. The relationships involved in the present document are as follows:

Table 5-2: Relationship of fault maintenance knowledge entities

Relationship	From	To
Processes	Technician	Fault event
Executes	Technician	Troubleshootings
Solve	Resolution actions	Root cause
Located on	Root cause	Device
Belong to	Technician, Antenna, Device	Cell
Covers	Cell	Base station

5.4 Knowledge Graph Quality Evaluation Methods

Knowledge quality evaluation is used to measure the accuracy, completeness, and consistency of knowledge. There are three main categories: statistical method, manual sampling method, and rule-based reasoning method. Statistical methods are efficient at measuring consistency and detecting outliers, which represent inconsistent data. The method based on manual sampling first samples the knowledge graph in a certain batch, and then manually evaluates the knowledge graph to detect possible problems. However, for large-scale knowledge graphs, it is particularly important to determine the appropriate sampling quantity and obtain results with statistical significance while reducing the cost of manual labelling as much as possible, which is also the research focus of the method based on manual sampling; The above two methods can be used to evaluate the quality of the knowledge graph, but it is difficult to use them to effectively detect and correct errors. After generating a knowledge graph, Knowledge graph quality evaluation assesses the knowledge graph and then feeds the assessment results back to the Graph construction and update module for model updates.

The rule-based reasoning method can implement the three tasks of quality evaluation, problem discovery, and quality improvement in a unified framework, and detect and repair errors by selecting reasonable rules, such as SQL rules, formal logic, graph function dependency, etc.

In terms of the quality evaluation of the graph, several indicators such as the consistency, coverage, accuracy, and extraction quality of the graph are proposed for the vocabulary itself:

1) Consistency: The knowledge graph is consistent with the constraints defined at the schema level.

2) Coverage/Completeness: Avoid missing elements related to agreed key edges (such as root cause), otherwise incomplete query results or derivation results, biased models, etc. may be generated.

3) Accuracy: Evaluate the degree to which entities and relationships (encoded by nodes and edges in the graph) correctly represent scene phenomena by combining random sampling.

4) Extraction quality/rate: the correspondence between the effective data involved in the graph construction and the graph scale (number of nodes and edges), and the data loss in the conversion process is evaluated.

NOTE: Integrity, Timeliness, Understandability, Accessibility, Universality and other evaluation indicators and dimensions are also very important for the knowledge graph, but they are not what the present document focusses on.

6 Application of wireless network fault maintenance knowledge graph

6.1 Knowledge reasoning

6.1.1 Motivation

Knowledge reasoning is mainly applied in two aspects, namely, the construction of knowledge graph and the application of knowledge graph. The former is to improve the preliminary constructed knowledge graphs. The latter is to apply the existing knowledge graphs to specific business scenarios.

The main purpose of applying knowledge reasoning to the construction of knowledge graph is to achieve Knowledge Graph Completion (KGC) and knowledge denoising. KGC is to infer the missing information in the knowledge base according to the existing triple, and improve the completeness of knowledge; Knowledge denoising is to use knowledge reasoning technology to infer contradictory relations and correct the knowledge in the knowledge base.

The main purpose of applying knowledge reasoning to actual business scenarios is to infer new knowledge or results that meet the requirements of business scenarios based on existing knowledge and reasoning strategies. Take the fault diagnosis of 5G wireless network, the key business scenario in the present document, as an example. After the knowledge graph described in clause 5 is constructed, the knowledge reasoning can be used to determine the root cause event and root cause network element of the fault. Then, the solution to root cause events is obtained through knowledge query to achieve rapid recovery of network fault, and realizing automatic maintenance of wireless network.

6.1.2 Functional Requirements

Knowledge reasoning is the application process of stock knowledge graph data, that is, discovering new knowledge through rule iteration or graph representation of the previous fault maintenance knowledge graph. Therefore, the specific functional requirements are as follows:

1) Ability of association analysis is needed, realizing the association analysis of knowledge based on the stock knowledge graphs.

2) Support rule-based knowledge reasoning ability. Rules should be able to provide conditions for similar subgraph search, graph clustering, and path search, and can be used as guidance for the corresponding rule reasoning ability. For example, based on expert experience, it should be able to complete iterative reasoning based on path and node attribute calculation on the graph.

3) Support graph-based knowledge reasoning capability, including but not limited to: the ability to achieve knowledge reasoning through linkage graph storage and graph training framework; the ability of knowledge reasoning is realized through embedded representation of knowledge and deep learning. At the same time, it can support multiple algorithms, and realize the functions of relationship prediction, attribute prediction, recommendation sorting, etc. in combine with downstream business data model.

4) Rules for knowledge reasoning process are provided, that can be used to support the knowledge evaluation module to evaluate whether the reasoning knowledge should be stored as trusted knowledge.

6.1.3 Interface requirements

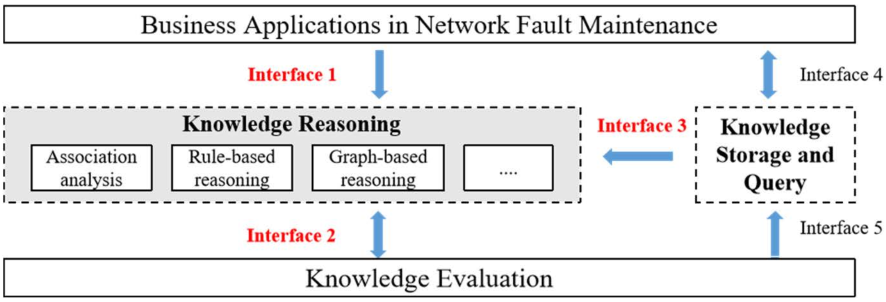

Figure 6-1: Schematic diagram of internal interface of knowledge reasoning

As shown in Figure 6-1, the knowledge reasoning function involves three interfaces. According to the functional requirements described in clause 6.1.2, each interface requirement is defined as follows:

1) **Interface 1:** Unidirectional interface. The business applications in network fault maintenance transmit the generated knowledge requirements to the knowledge reasoning module through Interface 1. According requirements, knowledge reasoning module uses the stock knowledge graph to complete association analysis, rule-based reasoning or graph-based reasoning to discover new knowledge. After being evaluated by the knowledge evaluation module, the reasoning results and new knowledge are output to business applications through the knowledge query module.

2) **Interface 2:** Bi-Directional interface. Knowledge reasoning module and knowledge evaluation module interacts information through this interface. On the one hand, the knowledge reasoning module transmits new knowledge and reasoning rules to the knowledge evaluation module to complete the credibility evaluation of new knowledge; On the other hand, the knowledge evaluation module feeds back the evaluation results to the knowledge reasoning module for the iteration of reasoning rules and new knowledge.

3) **Interface 3:** Unidirectional interface. Knowledge storage and query module transmits the stock knowledge graph data to knowledge reasoning module through Interface 3, which is mainly used to support the knowledge reasoning module to complete the generation of new knowledge.

6.2 Knowledge storage and query

6.2.1 Motivation

The storage and query of knowledge is the foundation of knowledge graph application. The main function of knowledge storage is to achieve reasonable storage of knowledge graph data and provide algorithmic support for rapid data retrieval.

The main function of knowledge query is to use the stock knowledge graph data to query the data information required by end users in a relatively short time. It will retrieve information from existing knowledge graphs to support decision8making and solve complex business problems. By applying knowledge querying, the wealth of information contained in knowledge graphs could be used to extract insights into the relationships between entities, properties, and concepts.

6.2.2 Functional Requirements

The main goal of this module is to efficiently store and query large-scale knowledge graph data.. Therefore, the specific functional requirements are as follows:

1) Two storage types are needed, including RDF (Resource Description Framework) based storage and graph database based storage.

2) Support RDF data storage and querying based on relational data models, such as simple triple lists, Horizontal Schema, attribute tables, vertical partitioning strategies, full indexing strategies, etc.

3) Support RDF data storage and querying based on graph models, such as SPARQL(SPARQL Protocol and RDF Query Language) queries.

4) Support KGQA (Knowledge Graph Question Answering) query methods , such as semantic parsing and information retrieval.

5) Support querying and retrieval of knowledge graph data based on user-specified criteria or parameters. This includes support for filtering, sorting, and aggregating query results.

6) Support knowledge query optimization to improve query task performance, such as the Worst-case Optimal Join method.

7) Data visualization is needed, providing users with a visual representation of the knowledge graph and its relationships, including support for interactive visualization, graph layout algorithms, and customization of visual styles.

8) Scalability is needed, enabling the knowledge query system to handle large and complex knowledge graphs efficiently. This should include support for distributed computing, indexing, and caching.

6.2.3 Interface requirements

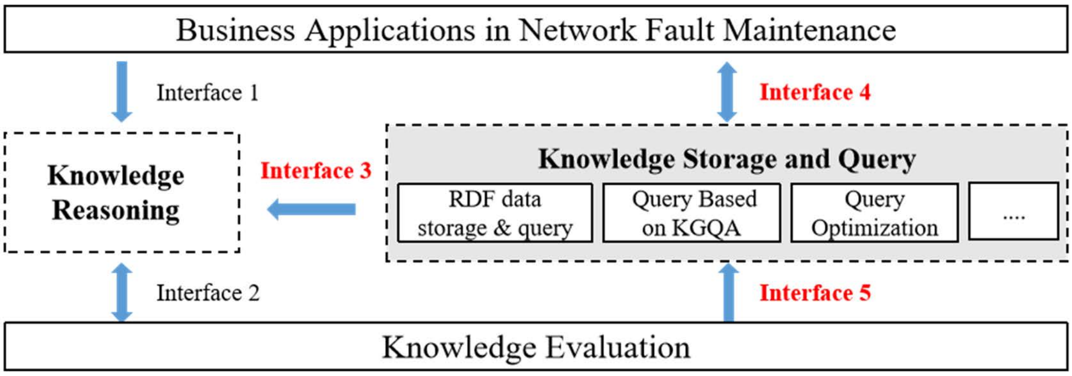

Figure 6-2: Schematic diagram of internal interface of knowledge storage and query

As shown in Figure 6-2, the knowledge storage and query function involves three interfaces. According to the functional requirements described in clause 6.2.2, each interface requirement is defined as follows:

1) **Interface 3:** As defined in clause 6.1.3.

2) **Interface 4:** Bi-directional interface. The interaction between the business applications in network fault maintenance and knowledge storage and query module is carried out through this interface. The business applications in network fault maintenance sends a request to knowledge storage and query module to query knowledge. After knowledge storage and query module completes the query of the corresponding knowledge, it returns the queried results to the business applications in network fault maintenance.

3) **Interface 5:** Unidirectional interface. The knowledge evaluation module interacts with the knowledge storage and query module through this interface. When new knowledge has been evaluated without any issues by the knowledge storage and query module, it is sent to the knowledge query module for storage and subsequent querying.

7 Use Cases

7.1 Cases of knowledge graph construction

7.1.1 Use Case #1-1: Knowledge graph of wireless network quality fault handling

7.1.1.1 Overview

The quality maintenance of wireless networks is flexible and tasking because it involves many infrastructure devices and various mobile terminals. At the same time, as the number of wireless network users continues to expand, wireless network fault maintenance work becomes even more difficult. Therefore, this clause introduces the construction and usage of knowledge graph for wireless network quality fault maintenance as a case study.

Knowledge graph of wireless network quality fault maintenance (WQMKG) refers to the formal description of entities and their relationships related to wireless network quality maintenance work. On the one hand, a large amount of work-order data is accumulated in wireless network quality fault maintenance work, which can be constructed as sub-knowledge graph of business fact; On the other hand, work-order processing experts have experienced long-term root cause diagnosis knowledge summary, which can provide a basis for building the sub-knowledge graph of expert experience. The sub-knowledge graph of business fact and sub-knowledge graph of expert experience are fused to form the WQMKG. After graph neural network knowledge inference based on the WQMKG, it helps to obtain work order solutions, which in turn provides decision assistance to work-order processors and optimizes customer network perception.

7.1.1.2 Data Acquisition and processing

Table 7-1: Data format and source

Name	Data type	Data sources
Equipment alarm data： **Network Performance metrics**	Structured data	Network acquisition
Operation and maintenance log: **Switching data**	Structured data	Network acquisition
Operation and maintenance database: **Quality fault work-order**	Structured data	Open source
Operation and maintenance database: **Measurement Report**	Structured data	Open source
Failure case ： **Root Cause Analysis (RCA) Cases**	Unstructured data	Manual or Open source//Third party database
Device profile ： **Wireless cell basic information**	Structured data	Network acquisition

Table 7-1 shows the specific data format and source of this case：

- **Structured data**

1) **Quality fault work-order:** Data with real resolution actions generated by the work-order system related to wireless network operation, including but not limited to work-order ID, corresponding cell, root cause and actual action, etc.

2) **Network performance metrics:** Network performance metrics is divided into four categories: access, hold, mobility management and latency, such as the number of connection requests and successes of RRU and E-RAB, the number of UE context abnormal release, the success rate of co-channel switching, the average latency of downlink PDCP SDU, etc.

3) **Wireless cell basic information:** the corresponding base stations, longitude, latitude, downlink frequency points, antenna direction angle and inclination angle for each cell in the target area.

4) **Measurement Report (MR):** MR measured by the user terminals, carries information about the upstream and downstream radio links, such as RSRP, weak coverage sampling points, etc.

5) **Switching data:** Switching records occurring between cells.

- **Unstructured data**

1) **Root Cause Analysis (RCA) Cases：**Get RCA cases from some telecom encyclopaedia webpages for wireless network problems, including low rate, dropped lines, fallback of data service and packet loss, dropped calls, unreachable, single pass, etc. of voice service.

7.1.1.3 Ontology/schema design

The ontology/schema structure of the knowledge graph of wireless network quality fault maintenance is shown in Figure 7-1.

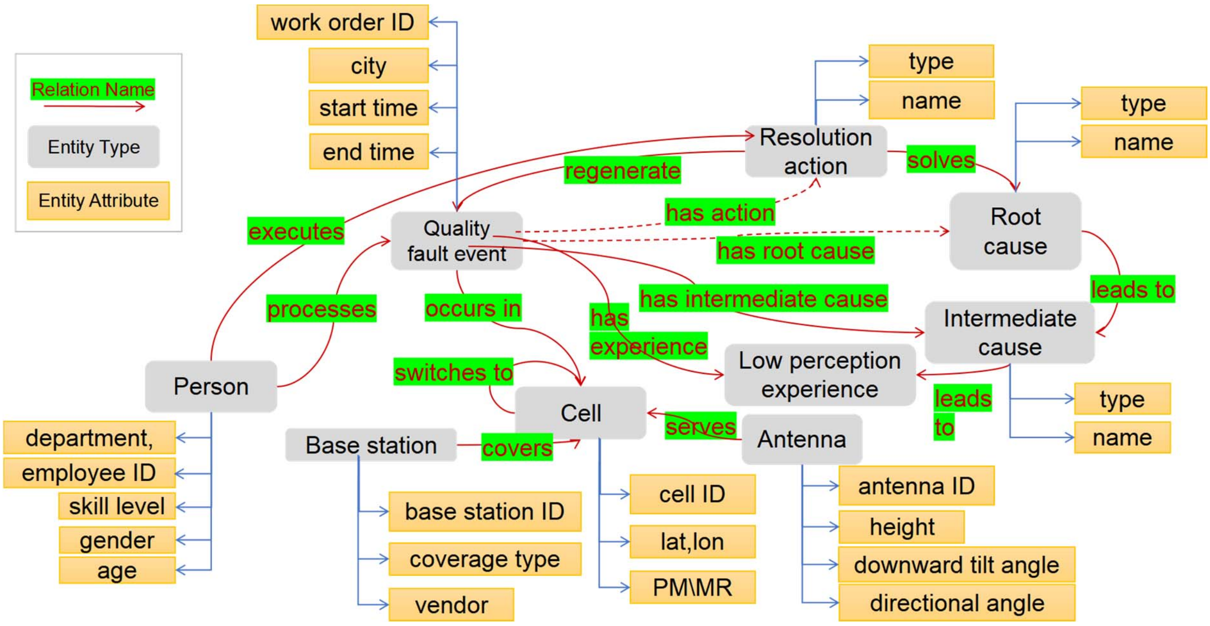

Figure 7-1: Ontology/schema structure of the wireless network quality fault maintenance knowledge graph

The ontology construction is closely related to the aforementioned data in clause 7.1.1.2, and also needs to meet the RCA scenario requirements of wireless network quality faults, specifically including entity construction and relationship construction.

- **Definition of types and attributes of knowledge entities**

  The following entity types are proposed:

1) **Quality fault event:** Describes an event in which the wireless network has experienced low perception, low access, high fallback of data services, or low perception of voice services. Attributes include work order ID, city, start time, receipt time, etc.

2) **Cell:** the area covered by a base station or a part of a base station (sector antenna) in the cellular mobile communication system. Its attributes include cell ID, latitude, longitude, and PM\MR related performance indicators, etc.

3) **Base station:** A radio station that provides wireless coverage for a cell. Its attributes include base station ID, coverage type, vendor, etc.

4) **Antenna:** A component used in radio equipment to transmit or receive electromagnetic waves and is closely related to the coverage cell. Its attributes include antenna ID, height, downward tilt angle, directional angle, etc.

5) **Person:** The person handling the quality fault work-order that can be issuers, forwarders and executors. Its attributes include department, employee ID, skill level, gender, age, etc.

6) **Low perception experience:** A description of the poor quality fault phenomena when using data and voice services on the wireless network. Attributes include type, name, etc.

7) **Intermediate causes:** A description of the surface causes of the low perception experience. Attributes include type, name, etc.

8) **Root cause:** A description of the underlying cause of the low perception experience. Attributes include type, name, etc.

9) **Resolution actions:** Means of resolution and control for quality fault event. Attributes also include type, name, etc.

- **Definition of relationship of knowledge entities**

  Various types of semantic relationships can be constructed between the above entity types: occurs in, switches to, covers, serves, leads to, solves, executes, processes, regenerates, has root cause, etc.

7.1.1.4 Knowledge Extraction and knowledge fusion

Structured tables such as quality fault work-order data can be used to construct knowledge graphs directly based on their field relationships. As for textual or pictorial expert knowledge data originating from telecom encyclopaedia webpages, their entity and relationship extraction can be performed in combination with manually developed rules and dictionaries.

As work-order system and encyclopaedia webpages use different root cause diagnosis systems, there are different descriptions of the same entity, redundant entities, overlapping concepts, and different entity types of entities with the same name, etc. It is necessary to carry out entity disambiguation and alignment according to the semantics of the entities, and complete the fusion of the two sub-graphs of business facts and expert experience.

An example of a knowledge graph of network quality fault maintenance aimed to the wireless coverage problem is shown in the Figure 7-2.

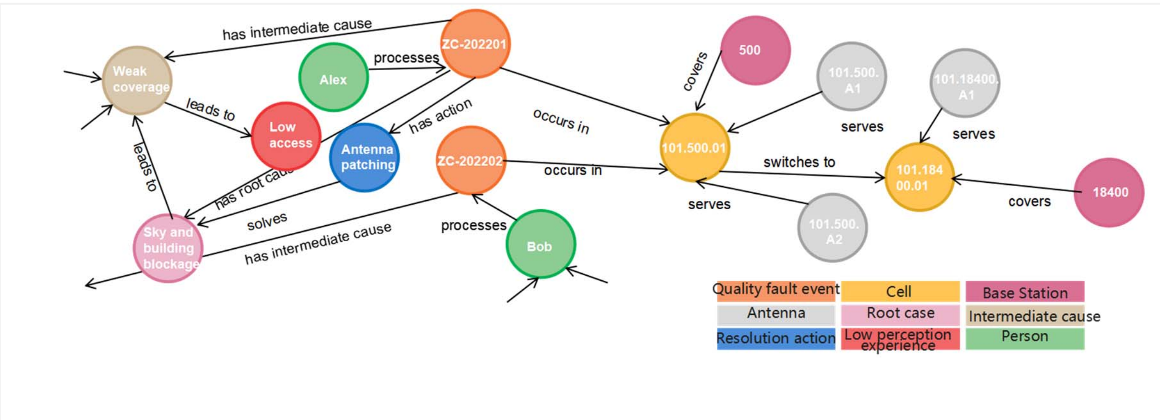

Figure 7-2: Example of the wireless network quality fault maintenance knowledge graph

7.1.1.5 Knowledge graph quality evaluation

The following evaluation indicators (mentioned in clause 5.4) of knowledge graph construction are proposed in this case:

1) Consistency: Compliance of the knowledge graph with the ontology model. The consistency of WQMKG is about 100,0 %.

2) Completeness: Existence of key relation edges (e.g. has root cause, has action) between related nodes. The completeness of WQMKG is 94,7 %.

3) Extraction rate: Ratio of the graph size (number of nodes and edges) to the amount of data involved in the graph construction, assessing the loss of data during the transformation process. The extraction quality of WQMKG is 86,5 %.

7.2 Cases of knowledge graph application

7.2.1 Use Case #2-1: Fault tracing in Wireless Network

7.2.1.1 Overview

AIOps aims to use artificial intelligence and big data technology to apply to business and operational data, complete network fault detection and exception avoidance, and achieve network autonomy. As the network scale continues to expand and business scenarios continue to increase, an auxiliary means is needed to enhance the fault tracing capability of AIOps.

7.2.1.2 Motivation

The current AIOps technology is difficult to cope with the increasingly complex network environment, so external experts are required to fill the gaps in various scenarios, so as to assist the operation and maintenance personnel to track faults more accurately.

7.2.1.3 Actors and roles

The presence of the following actors/entities as well as their associated roles are envisaged in the current use case:

- Network data collector: collect network layer data and complete preliminary sorting of data.
- Knowledge extractor: extract entities in data, conduct knowledge fusion, and build knowledge graph.
- Fault Tracker: detect network data to determine whether there is a fault. In case of network fault, conduct root cause analysis according to the knowledge graph.

7.2.1.4 Operational flow of actions

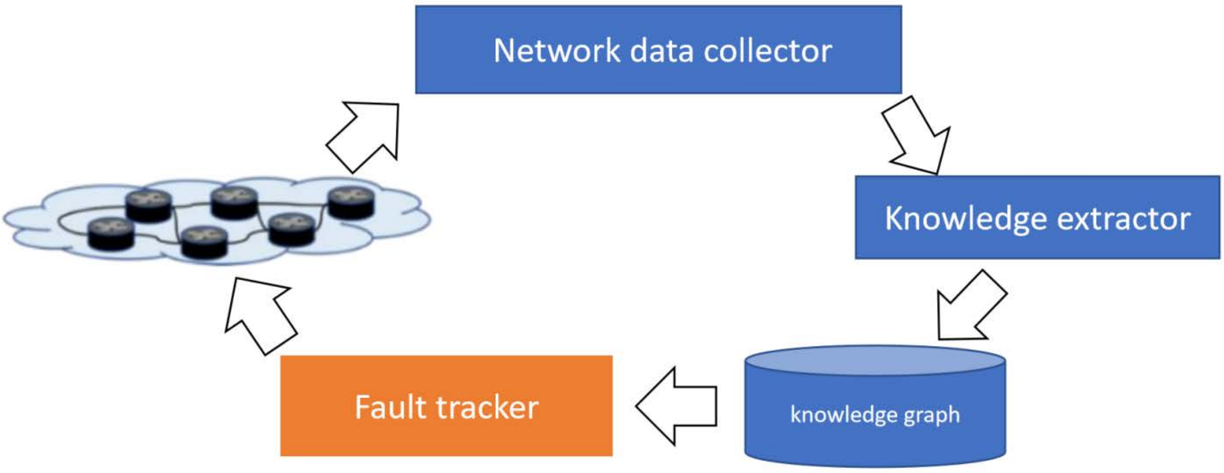

Figure 7-3: Fault Tracking Based on Knowledge Graph

The following sequence of actions may be identified:

1) At the beginning, network data collector collects and standardizes network data.
2) The knowledge extractor extracts the effective information from the data at regular intervals, to establish knowledge subgraphs. Then the knowledge subgraphs is fused and verified to generate the final knowledge graph.

3) After the knowledge graph is generated, the network data collector and knowledge extractor still update the knowledge graph periodically.

4) When the network data is abnormal, the fault tracker queries the source of abnormal traffic according to the knowledge graph, finds the node with fault, and completes root cause analysis.

7.2.1.5 Post-conditions

- When a fault occurs, find the fault and analyse the root cause.
- Automatically generate and maintain knowledge graph.

7.2.2 Use Case #3-1: Intelligent Management of Home LAN

7.2.2.1 Overview

Unlike large commercial networks, there are not any engineers in a home LAN to ensure proper network operation. Besides, with the development of home switches and home routers, these types of equipment are capable of deployment of the knowledge graph, which may be in use of intelligent management of home LAN.

7.2.2.2 Motivation

In this use case, typically, the workload of a home LAN is way lower than other networks. Apart from that, faults that occur in a home LAN are simpler to deal with. In most scenarios, these faults are caused by the customer's misoperation and detected by the router. User data and error information will be collected for building and deploying a knowledge graph. With a knowledge graph deployed in a router, a home LAN can be intelligently managed without supervision by an expert engineer.

7.2.2.3 Actors and roles

The presence of the following actors/entities as well as their associated roles are envisaged in the current use case:

- Customer: end users that enjoy the delivery of a service in-home LAN.
- Network data collector: programmable module deployed in a router to collect live data of home LAN and to detect network errors from the data.
- Knowledge extractor: programmable module running in a router to extract knowledge from data and build a knowledge graph.
- Policymaker: programmable module that automatically generates new policies.
- Basic router functions: functions like transmission, forwarding, firewall, etc.

7.2.2.4 Initial context condition

The network is operating in perfect conditions with all its components working in good shape. The router is pre-installed some equipment manuals and expert handbooks.

7.2.2.5 Operational flow of actions

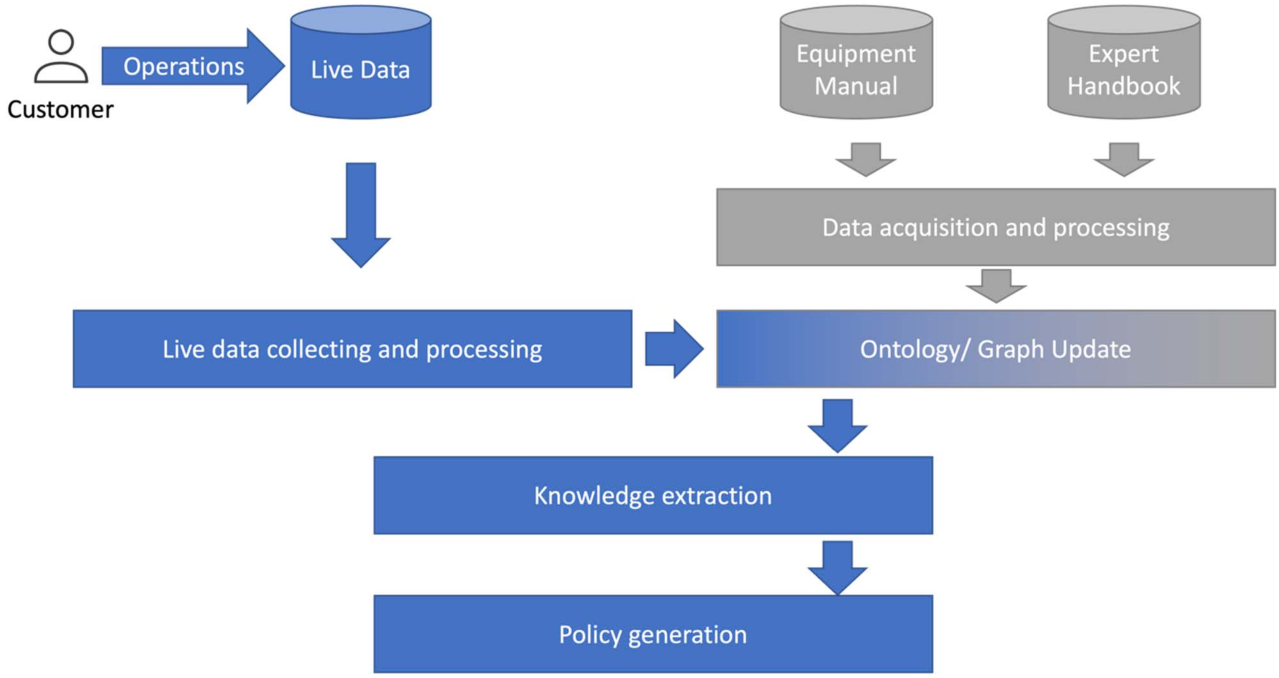

Figure 7-4: Procedure of intelligent management of home LAN

The following sequence of actions may be identified:

1) At the beginning, knowledge extractor builds an initial knowledge graph using pre-installed equipment manuals and expert handbooks.

2) As the network is operating, the network data collector collects home LAN QoS, and crucial network information e.g. link bandwidth, transmission delay, etc. Simultaneously, the network data collector transfers the data to the knowledge extractor.

3) Knowledge extractor gets knowledge from the data network data collector has collected. If the key of knowledge does not exist in the current knowledge graph, the knowledge extractor inserts the knowledge into the graph and creates a logical link between relevant knowledge. If the key of knowledge exists, the knowledge extractor compares the current value of the knowledge with the value extractor gets and replaces the current one if they do not match.

4) Customer misoperates by accident, causing glitches in current network policies.

5) Network data collector detects malfunctions in the network and reports to the policy maker.

6) Policymaker retrieves necessary knowledge from the knowledge graph deployed in the router. The knowledge corrects the parameters of AI models used in policy generation, which brings better precision to the output and shortens the processing time.

7) Basic router functions read new policies to maintain customer service.

7.2.2.6 Post-conditions

- Home LAN achieves self-management.
- The network remains in good condition.
- Network resources are adjusted dynamically and automatically.

8 Research on cross network fields fault maintenance knowledge graphs construction methods (e.g. access networks)

The traditional maintenance of cross network fields fault rely entirely on the professional ability of maintenance personnel. Through past experience, the correlation between common alarms is summarized and programmed by expert system. This method has poor scalability and its correctness cannot be guaranteed in the short term.

With the continuous emergence of new technologies, networks, and businesses, the magnitude of network fault events is growing rapidly, and the correlation is also showing diversification. Therefore, the trend of efficient cross network fields collaborative fault maintenance is becoming more urgent, and it is necessary to achieve rapid compression of cross network fields alarms and intensive distribution of fault work orders. The expert rule summary method is completely unable to meet the above needs. Constructing a cross network fields knowledge graph of wireless, transmission, core network, and even end-to-end, has become one of the effective ways to solve the above problems.

The process and key technical requirements for constructing cross network fields knowledge graph and single network field knowledge graph are basically consistent, but it is necessary to solve problems such as definition conflicts, content duplication, unclear references, and hierarchical confusion caused by different network fields and vendors in the stages of data processing, ontology/schema design, and knowledge fusion, to achieve a unified knowledge representation and trusted sharing of graph data.

NOTE: At present, the implementation and practice of the cross fields knowledge graph in the mobile network is still in the initial stage, and there is no mature cross network fields knowledge graph and case studies published in the industry. So this part needs further study.

9 Conclusions

In the evolution of network intelligence, data and knowledge have always played an important role. With the increasing complexity and flexibility of 5G/B5G and future 6G network structures, as well as the skyrocketing scale of network data, the difficulty and complexity of network management will also increase. It is expected to use knowledge graphs to deeply understand the correlation between network data, and comprehensively perspective network performance from dimensions such as network, terminal, user, and business, accurately locate key factors that affect user experience quality, and continuously detect and eliminate anomalies in a timely manner. Ultimately achieving true intelligent operation and maintenance.

Based on the above current situation and development trends, a framework of Network Fault Maintenance Knowledge Graph Construction Process is proposed in the present document. Through a detailed analysis of construction and application cases of wireless network fault maintenance knowledge graph, the key technologies and implementation methods are further elaborated. It is expected that the present document can provide reference for the scientific research and application practice of the knowledge graph of communication network operation and maintenance in the industry, and assist in the evolution of Autonomous Network. However, these studies have not yet been fully completed， there are still many problems that need to be solved together in order to construct a comprehensive knowledge graph of communication networks. Standardization research and formulate work can be carried out in the following aspects to settle these problems:

- Construction of fault maintenance knowledge graphs in each network field：Communication network is a sophisticated system. Currently, loose coupling Q&M management mechanisms are adopted between different network fields such as wireless , transmission and core network. It is necessary to establish knowledge graphs for each of these network fields, and finally integrate them to obtain a complete and comprehensive knowledge graph to support comprehensive intelligence in Q&M management.

- Construction of Network Data Knowledge Graph Based on Communication Protocol and Principles : Generally speaking, data determines the superior limit of the knowledge graph. The construction of network data knowledge graphs aims to intuitively display the causal relationships and association rules between data fields and indicators related to mobile communication protocols, performance, and processes in the form of a knowledge graph. The data fields and indicators are represented by nodes, and the causal relationships and association rules between data fields and indicators are represented by edge connection relationships. This will provide a reference for subsequent research on knowledge graph fusion in different network fields.

- Construction of Multi-Modal Knowledge Graph (MMKG): Traditional knowledge graphs mainly focus on studying the entities and relationships of text and databases, while MMKG construct entities under multiple modalities (such as visual modalities) and multi-modal semantic relationships between multi-modal entities on the basis of traditional knowledge graphs. It may become one of the key technologies for the intelligent Q&M of polymorphic smart network.
- Trusted Sharing of Knowledge Graphs: Communication networks are connected by different types of devices from different manufacturers, so network fault maintenance involve the integration and collaboration of multi-source heterogeneous device Q&M processes and experiences. Therefore, when construct an end-to-end unified trusted knowledge graph for the entire communication network, how to achieve trusted sharing of knowledge graphs is a key problem that needs to be solved.

Annex A:
Change history

Date	Version	Information about changes
March 2022	V0.0.1	First draft
December 2022	V0.0.2	Second draft
March 2023	V0.0.3	ENI(23)025_022 & ENI(23)025_025r1
June 2023	V0.0.4	ENI(23)026_037r5

History

Document history		
V4.1.1	May 2024	Publication