ITU-T

**TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU**

Y.4414/H.623

(11/2015)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Internet of things and smart cities and communities – Frameworks, architectures and protocols

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Broadband, triple-play and advanced multimedia services – Advanced multimedia services and applications

Web of things service architecture

Recommendation ITU-T Y.4414/H.623

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	
General	Y.4000–Y.4049
Definitions and terminologies	Y.4050–Y.4099
Requirements and use cases	Y.4100–Y.4249
Infrastructure, connectivity and networks	Y.4250–Y.4399
Frameworks, architectures and protocols	**Y.4400–Y.4549**
Services, applications, computation and data processing	Y.4550–Y.4699
Management, control and performance	Y.4700–Y.4799
Identification and security	Y.4800–Y.4899

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.4414/H.623

Web of things service architecture

Summary

Recommendation ITU-T Y.4414/H.623 defines a web of things (WoT) service architecture that can encompass service discovery, accessibility, sharing and mash-up for IoT devices and services with web technologies. It includes an overview of WoT services, the functional architecture of WoT services and WoT service/resource functions.

The WoT service architecture supports accessibility and reusability across IoT resources, and supports portability across heterogeneous network environments. Therefore, this Recommendation is applicable to seamless and interoperable IoT services with information interaction and exchange over physical IoT devices.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Y.4414/H.623	2015-11-29	16	11.1002/1000/12647

Keywords

Web of things, WoT, WoT service architecture.

---

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

Table of Contents

Recommendation ITU-T Y.4414/H.623

Web of things service architecture

1   Scope

The objective of this Recommendation is to define a WoT service architecture. The scope of this Recommendation covers the following:

−   overview of the WoT applications

−   a functional architecture for the WoT service

−   functional entities for the WoT service

2   References

The following ITU-T Recommendations and other references contain provisions, which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T M.3030]   Recommendation ITU-T M.3030 (2002), *Telecommunications Markup Language (tML) framework*.

[ITU-T Y.2002]   Recommendation ITU-T Y.2002 (2009), *Overview of ubiquitous networking and of its support in NGN*.

[ITU-T Y.2232]   Recommendation ITU-T Y.2232 (2008), *NGN convergence service model and scenario using web services*.

[ITU-T Y.4000]   Recommendation ITU-T Y.4000/Y.2060 (2012), *Overview of the Internet of things*.

[ITU-T Y.4400]   Recommendation ITU-T Y.4400/Y.2063 (2012), Framework of the w*eb of things*.

3   Definitions

3.1   Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1   client [b-W3C WACterms]: The role adopted by an application when it is retrieving and/or rendering resources. A program establishes connections for the purpose of sending requests.

3.1.2   device [b-W3C digloss]: An apparatus through which a user can perceive and interact with the Web.

3.1.3   Internet of things (IoT) [ITU-T Y.4000]: A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on, existing and evolving, interoperable information and communication technologies.

NOTE 1 – Through the exploitation of identification, data capture, processing and communication capabilities, the IoT makes full use of things to offer services to all kinds of applications, whilst ensuring that security and privacy requirements are fulfilled.

NOTE 2 – In a broad perspective, the IoT can be perceived as a vision with technological and societal implications.

3.1.4   metadata [ITU-T M.3030]: Data that describes other data.

3.1.5   resource [b-IETF RFC 3986]: This specification does not limit the scope of what might be a resource; rather, the term "resource" is used in a general sense for whatever might be identified by a URI. Familiar examples include an electronic document, an image, a source of information with a consistent purpose (e.g., "today's weather report for Los Angeles"), a service (e.g., an HTTP-to-SMS gateway), and a collection of other resources. A resource is not necessarily accessible via the Internet; e.g., human beings, corporations, and bound books in a library can also be resources. Likewise, abstract concepts can be resources, such as the operators and operands of a mathematical equation, the types of a relationship (e.g., "parent" or "employee"), or numeric values (e.g., zero, one, and infinity).

3.1.6   server [b-W3C WACterms]: The role adopted by an application when it is supplying resources.

3.1.7   service provider [ITU-T Y.2232]: An entity that provides services.

3.1.8   SOAP [b-W3C SOAP 1]: The formal set of conventions governing the format and processing rules of a SOAP message. These conventions include the interactions among SOAP nodes generating and accepting SOAP messages for the purpose of exchanging information along a SOAP message path.

3.1.9   SOAP intermediary [b-W3C SOAP 1]: A SOAP intermediary is both a SOAP receiver and a SOAP sender and is targetable from within a SOAP message. It processes the SOAP header blocks targeted at it and acts to forward a SOAP message towards an ultimate SOAP receiver.

3.1.10   the World Wide Web (WWW, or simply web) [b-W3C webarch]: An information space in which the items of interest, referred to as resources, are identified by global identifiers called Uniform Resource Identifiers (URI).

3.1.11   thing [ITU-T Y.4000]: In the Internet of things, object of the physical world (physical things) or of the information world (virtual things), which is capable of being identified and integrated into the communication networks.

3.1.12   URI [b-IETF RFC 3986]: A URI is an identifier consisting of a sequence of characters matching the syntax rule named <URI> in Section 3 of [b-IETF RFC 3986]. It enables uniform identification of resources via a separately defined extensible set of naming schemes (Section 3.1 of [b-IETF RFC 3986]). How that identification is accomplished, assigned, or enabled is delegated to each scheme specification.

3.1.13   web of things [ITU-T Y.4400]: A way of realization of the IoT where (physical and virtual) things are connected and controlled through the World Wide Web.

3.1.14   web resource [b-W3C WACterms]: A resource, identified by a URI.

3.1.15   web services [ITU-T Y.2232]: Web services is a service provided using web services systems.

3.1.16   web services gateway (WSG) [ITU-T Y.2232]: A gateway which handles the web services message between the WSP and WSR.

3.1.17   web services provider (WSP) [ITU-T Y.2232]: A service provider that exposes a capability for use to create web services.

3.1.18   web services registry [ITU-T Y.2232]: An entity that stores web services information (e.g., WSDL).

3.1.19   Web services requester (WSR) [ITU-T Y.2232]: Client software that makes use of the services provided by a WSP.

3.1.20 web services system [ITU-T Y.2232]: A web services system is a software system designed to support interoperable machine-to-machine interaction over a network using web services standards.

NOTE – It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the web services in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards.

3.1.21 WSDL [b-W3C WSDL 1]: Web services description language Version 2.0 (WSDL 2.0) provides a model and an XML format for describing web services. WSDL 2.0 enables one to separate the description of the abstract functionality offered by a service from concrete details of a service description such as "how" and "where" that functionality is offered.

3.2   Terms defined in this Recommendation

None.

4   Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

API	Application Programming Interface
CPU	Central Processing Unit
DPWS	Device Profile for Web Services
GPU	Graphics Processing Unit
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
ID	Identification
IoT	Internet of Things
IP	Internet Protocol
JSON	JavaScript Object Notation
NFC	Near Field Communication
QoS	Quality of Service
RDFa	Resource Description Framework in attributes
REST	Representational State Transfer
SOAP	Simple Object Access Protocol
TCP	Transmission Control Protocol
UI	User Interface
URI	Uniform Resource Identifiers
Web-RM FE	Web Resource Management Functional Entity
WoT	Web of Things
WoT-BM FE	WoT Broker Management Functional Entity
WoT-DM FE	WoT Description Management Functional Entity
WoT-EN FE	WoT Enrolment Functional Entity
WoT RMF	WoT Resource Management Function

WoT-S CF	WoT Service Control Function
WoT-S SF	WoT Service Support Function
WoT-S-CR FE	WoT Service Creation Functional Entity
WoT-S-DIS FE	WoT Service Discovery Functional Entity
WoT-S-EX FE	WoT Service Execution Functional Entity
WoT-S-MA FE	WoT Service Management Functional Entity
WoT-S-MON FE	WoT Service Monitoring Functional Entity
WoT-S-PM FE	WoT Service Policy Management Functional Entity
WoT-S-PRM FE	WoT Service Profile Management Functional Entity
WoT-S-RE FE	WoT Service Registration Functional Entity
WSDL	Web Service Description Language
XML	extensible Markup Language

5 Conventions

None.

6 Overview of the WoT service

The web is used as a global standard platform to deliver services to an end user and applications to intercommunicate with each other over a network. [ITU-T Y.4400] describes that the web has program language independent properties, uses message driven communications and easily bounds to different transport protocols. Web technology allows the exposure of physical devices as well as several kinds of content as web resources such as URI and HTTP. Therefore, users and service developers can interact with the devices using web interfaces. The WoT can provide capabilities such as device reusability, portability across several heterogeneous networks and accessibility based on web standards technologies.

Figure 1 shows a general concept of the WoT service. The physical devices as well as content are integrated into the web and are viewed as the sort of services provided by the web environment. In addition, each service has a mash-up capability to create a new service by composition of the existing services in the web environments. According to [ITU-T Y.4400], the web technologies can support to create applications of networked devices and services.
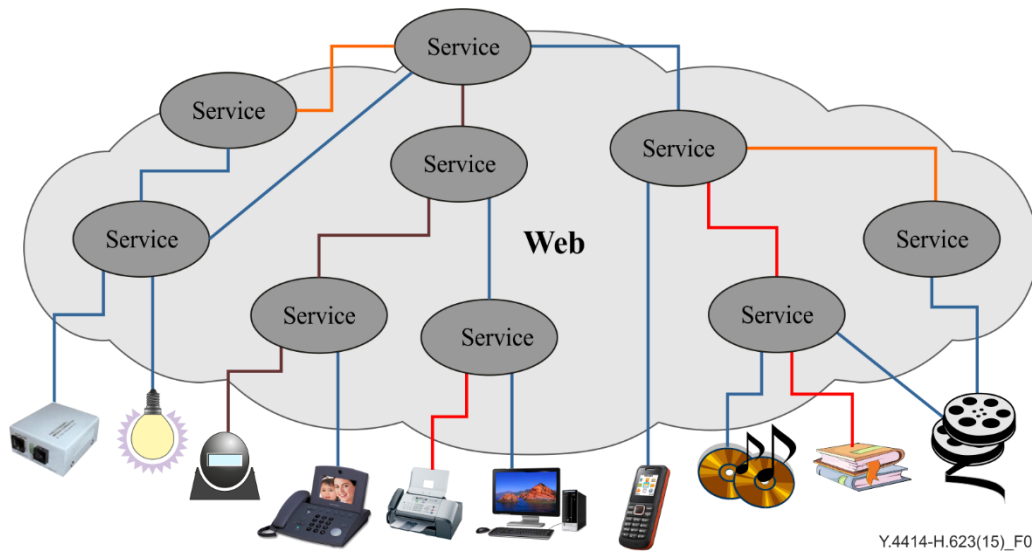
Figure 1 – General concept of a web of things service

7 Functional architecture of the WoT service

Figure 2 shows an overview of the functional architecture of a WoT service.

Figure 2 – Functional architecture of WoT service

NOTE – Virtual things exist in the information world and are capable of being stored, processed and accessed. Examples o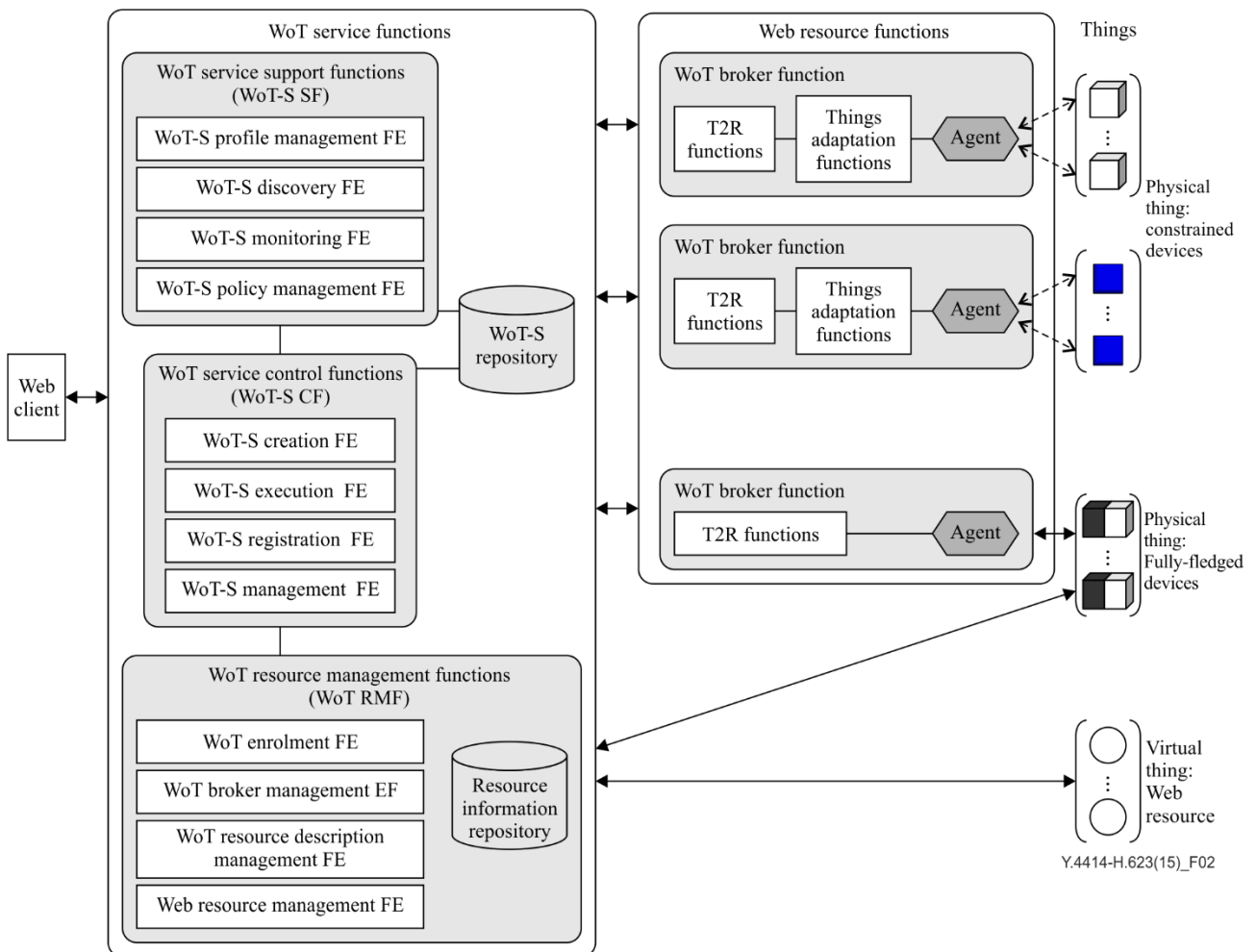f virtual things include multimedia content and application software [ITU-T Y.4000]. In the case of a WoT service architecture, virtual things exist as the web resources on the web environments.

7.1 WoT service functions

The WoT service functions provide WoT services to WoT clients. They manage overall behaviours related to the WoT service. If a WoT client requests a WoT service, the WoT service functions analyse the request then discover and provide the services for a WoT client. The WoT service functions have three sub-functions as follows:

– **WoT service support functions (WoT-S SF)**: The WoT-S SF manages overall behaviours of the WoT service. The WoT-S SF are responsible for providing service profile management, service discovery, service monitoring, QoS management, access control and policy management of WoT service. And it is also responsible for containing and updating the WoT service information.

– **WoT service control functions (WoT-S CF)**: They control the WoT service which are registered in the WoT service providers. It is responsible for executing and creating the WoT service and mash-up services.

– **WoT resource management functions (WoT RMF)**: They contain resource information (things information) which is used by and registered at the WoT service functions.

7.2 Web resource functions

The web resource functions have responsibility for accessing the things by enabling web technologies. If there are no relevant web technologies to access the things (devices) directly, the WoT broker function should be used which provides access to the devices on the web. Also it is responsible for integrating the web resources (e.g., fully-fledged device and constrained device) to the web [ITU-T Y.4400]. If it is a virtual thing, it can be accessed directly by the WoT resource management function. The T2R function has a role for integrating and exposing things to the web resources. In [ITU-T Y.4400], the service layer of the WoT broker is responsible for the T2R role. And, the things adaptation functions are responsible for adaptation.

7.3 Web client

With the web client, an end user can use WoT services provided by an accessed physical device or virtual thing. The web browser and web applications allow the web client to use WoT services. The web client can control devices by using a web browser and web technologies such as HTTP, RESTful API etc.

7.4 Things

In the WoT service, the things can be classified into two types; physical things and virtual things. The physical things can be classified into two types of devices, constrained devices and fully-fledged devices. The things can be accessed and exposed on the web as a WoT service.

7.4.1 Physical things

7.4.1.1 Constrained device

The constrained device is a non-capable device to connect to the Internet or web alone. And it has no functionality of web. It has just its own proprietary interfaces and protocols such as ZigBee, Bluetooth, NFC, etc. These interfaces and protocols do not have interoperability with web technologies. In this case, the devices need the WoT broker. By exposing on web through the WoT broker, the devices can communication with the WoT broker using their proprietary interface. And the WoT broker has a role of agent for these devices in order to communicate with the web user.

7.4.1.2   Fully-fledged device

The fully-fledged device is a device to support direct web connectivity functionalities, therefore there is no need to translate HTTP requests from the web clients into the appropriate protocol.

However, to enable this capability, the web server is built in the devices itself.

The fully-fledged device is necessary to satisfy the following two capabilities:

–   support of TCP/IP protocol suite

–   web server.

These capabilities enable devices seamless integrating to the web and their functionality through the web interface.

7.4.2   Virtual things

The virtual things are information world objects such as media, image, video and documents, etc. These objects are web resources exposing through the web service.

8   WoT service functions

8.1   Functional entities of the WoT service support functions (WoT-S SF)

The WoT-S SF manages overall behaviours of the WoT service. The WoT-S SF are responsible for providing service profile management, service discovery, service monitoring, QoS management, access control and policy management of a WoT service. The WoT-S SF consists of four functional entities (FEs); the WoT service profile management FE (WoT-S-PRM FE), the WoT service discovery FE (WoT-S-DIS FE), the WoT service policy management FE (WoT-S-PM FE) and the WoT service monitoring FE (WoT-S MON FE). The WoT-S SF also interworks with the WoT-S repository to register and discover WoT services, etc.
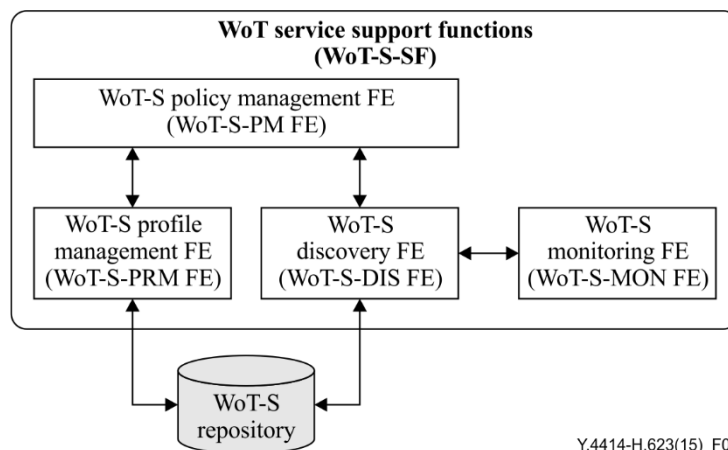
Figure 3 – Functional entities and interfaces of WoT-S SF

8.1.1   WoT service profile management functional entity (WoT-S-PRM FE)

The WoT-S profile management FE (WoT-S-PRM FE) is responsible for managing the profile of the registered services. The service profile is a record that contains information related to the WoT service such as version of service, service life cycle, service classification etc. If a service version is changed, this information is reflected in the registered WoT-S repository. The WoT-S-PRM FE interworks with the WoT-S repository to maintain the latest information of the WoT service.

A WoT user requests a service through the WoT-S-PRM FE and the WoT-S-PRM FE forwards the request to the WoT-S-DIS FE. The WoT-S-DIS FE analyses the request and searches the services in

the WoT-S repository or on the web. The WoT-S-DIS FE returns the result to the WoT-S-PRM FE. And the WoT-S-PRM FE provides the service to the WoT user.

8.1.2 WoT service discovery functional entity (WoT-S-DIS FE)

The WoT service discovery FE (WoT-S-DIS FE) provides a discovery capability in collaboration with web resource functions. The WoT-S-DIS FE interworks with the WoT broker functions to discover the WoT service composed of physical devices.

In the WoT service environments, lots of the WoT services will be presented and provided on the web and many users and service developers want to search and find them on the web. In the context of the WoT, the simplest way of enabling search for the WoT service (things) is using the web search engines like the web documents. However, a search for things as the WoT service is more complicated than a search for documents on the web due to its contextual information such as their absolute location or their relative location.

8.1.3 WoT service monitoring functional entity (WoT-S-MON FE)

The WoT service monitoring FE (WoT-S-MON FE) is responsible for monitoring the registered WoT service. The web client or other FEs (WoT-S-DIS FE, WoT-S-PRM FE) utilize the WoT-S-MON FE to find out the status of the WoT service. The status of the WoT service is recommended to include service availability or predict response time, static/dynamic status information and resource information, e.g., CPU/GPU power.

The WoT-S-MON FE is recommended to provide static and dynamic information in order to guarantee a stable quality of the WoT service. Static information consists of quality requirements, e.g., response time, min performance value, max performance value and normal performance value for the WoT service. Dynamic information consists of a real-time updated status for the WoT system and service. For example, system status includes application ID, network port, the amount of instant data and average data, network bandwidth, and so on. Service status includes execution time, fps, screen size, and so on.

The WoT-S-MON FE is required to monitor system resource information of processor power consumption, e.g., CPU and GPU, in order to predict system power consumption of the WoT service. Therefore, the system resource is necessary to provide the QoS guaranteed WoT service such as a high performance with lower power consumption. A GPU resource is measured by using a power coefficient corresponding to each GPU frequency, GPU utilization and GPU static power.

8.1.4 WoT service policy management functional entity (WoT-S-PM FE)

The WoT service policy management FE (WoT-S-PM FE) provides three capabilities for the WoT service as follows:

–   QoS information management: The WoT-S-PM FE is responsible for managing QoS information about the registered WoT service. It checks accessibility, performance, reliability of the WoT service. The WoT-S-PM FE is also responsible for providing QoS information to the web client.

–   Service access control: The WoT-S-PM FE provides service access control capabilities to control the accessibility of a specific service by an application. It provides authentication and authorization actions required to ensure that the web client has appropriate access rights to the requested service.

–   Security and safety management: The WoT-S-PM FE provides control capabilities regarding security and safety in order to support a trustworthy WoT service. It provides methods to guarantee the safe execution of service in the WoT system, which are comprised of actions by detecting an execution request and controlling execution of the service command.

8.2 Functional entities of the WoT service control functions (WoT-S CF)

The WoT-S CF is responsible for controlling the WoT service creation, service execution, service registration and service management. The WoT-S-CF consists of four functional entities (FEs); the WoT-S management FE (WoT-S-MA FE), the WoT service execution FE (WoT-S-EX FE), the WoT-S creation FE (WoT-S-CR FE), the WoT service registration FE (WoT-S-RE FE) with WoT-S repository.
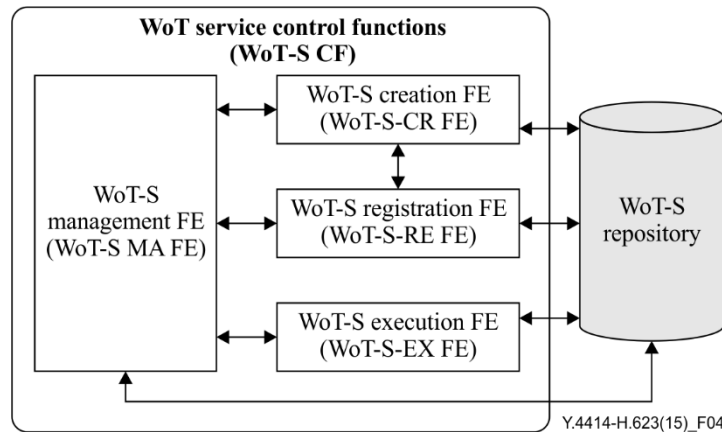

Figure 4 – Functional entities and interfaces of WoT-S CF

8.2.1 WoT service execution functional entity (WoT-S-EX FE)

The WoT service execution FE (WoT-S-EX FE) is responsible for executing the WoT service. The WoT-S-EX FE enables the WoT service user to access and execute the WoT service. When an end user (web client) wants to use the WoT service, the WoT-S-EX FE interacts with the WoT-S-PM FE located in the WoT-S SF to check the authentication and the authorization of the requester. Additionally, the WoT-S-PRM FE also checks availability of the service at that time.

The WoT-S-EX FE interworks with the WoT-S repository which includes logics to execute a service. The WoT execution logics have a method for how to operate services.

The basic functionalities of the WoT-S-EX FE on managing logic are:

–   storing and management logics describing an event in the WoT-S repository, a condition to be satisfied by the event, and an action to be executed when the conditions are satisfied;

   NOTE – When a service requester requests to register the logic, the logic is stored if and only if the logic does not generate a static conflict in which the logic describes different actions upon the same event and the same conditions for the action with other logics.

–   retrieving a logic describing the event storage when an event occurs;

–   inspecting the conditions whether or not the event satisfies a condition described in the retrieved logic;

–   executing an action described in the retrieved logic when the event satisfies the condition.

8.2.2 WoT service creation functional entity (WoT-S-CR FE)

The WoT service creation FE (WoT-S-CR FE) allows the WoT service to be developed by simple composition of the existing WoT services on top of web-based things.

The WoT-S-CR FE is responsible for managing and executing mash-up WoT services. The WoT-S-CR FE provides two functions: mash-up logic management and mash-up engine.

A WoT user is able to search, create or update service execution logic through mash-up logic management via the web access. The service execution logic is stored and retrieved using the WoT-S repository. A WoT user can also request a mash-up engine to operate the service execution

logic for the mash-up WoT service. Each unit of the WoT service for the mash-up service is executed by the WoT-S-EX FE.

NOTE – The web mash-up is defined as "web applications generated by combining content, presentation or application functionality from the disparate web sources". The aim of mash-up is to create new useful applications or services by combining them in a value-adding manner. The WoT allows developers to access and search for the web-enabled things, and also allows owners of the web-enabled things to have a simple and scalable mechanism on sharing them.

8.2.3 WoT service management functional entity (WoT-S-MA FE)

The WoT service management FE (WoT-S-MA FE) is responsible for managing the WoT service. It provides capabilities to manage service updating, service tracking, update management, auditing, version control, logging and service coordination.

The WoT-S-MA FE supports the collection and storage of the WoT service logs. It also provides capabilities to replace or substitute the WoT service. If the WoT-S-MA FE finds better services, it recommends to change the old service to the new service. The WoT-S-MA has the ability to detect failures of services as well as recovery from failures.

8.2.4 WoT service registration functional entity (WoT-S-RE FE)

The WoT service registration FE (WoT-S-RE FE) allows registration and deregistration of the WoT service in the WoT service repository. The WoT-S-RE FE has the ability to analyse services for registration in terms of their characteristics (e.g., service category, service provider and information about service charging).

The WoT-S-RE FE interworks with the WoT service creation FE (WoT-S-CR FE). The WoT-S-RE FE registers the new service after creating a service by the WoT-S-CR FE. The WoT-S-RE FE investigates the service and extracts information of the service.

8.2.5 WoT service repository (WoT-S repository)

The WoT service repository is responsible for storing WoT services and related information such as service characteristics, service type, service location, service version, service status, etc. The WoT service repository also contains the service execution logic, which comprises the service execution description of each WoT service. Mash-up services also have the service execution description which contains service information, service rule and service profiles like other WoT services:

–   service information is related to the linked information with the WoT-S repository or in case of mash-up services, it contains key features for WoT mash-up services such as classification, QoS, owner, etc.;

–   service rule is related to the method on how to operate services, or in case of mash-up services, it contains one or mixed rules of sequential execution, conditional execution and operational execution;

–   service profiles are related to the parameters to provide customised services according to location, preference or other clients' characteristics.
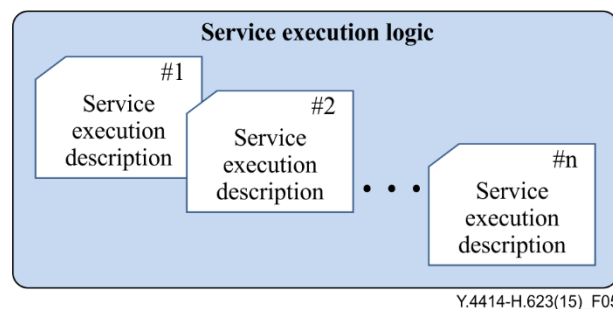

Figure 5 – Service execution logic in a WoT-S repository

8.3 Functional entities of the WoT resource management functions (WoT RMF)

The WoT RMF manage resources that are used for the WoT service. The WoT RMF can register resources into a resource information repository and can delete them. The resources are provided by the WoT broker functions and the web. The WoT RMF consists of four functional entities; the WoT enrolment FE (WoT-EN FE), the WoT broker management FE (WoT-BM FE), the web resource management FE (Web-RM FE) and the WoT description management FE (WoT-DM FE).
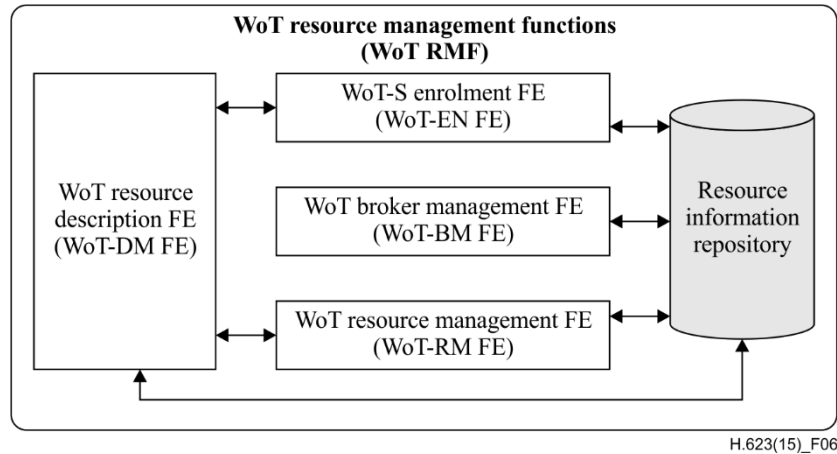

Figure 6 – Functional entities and interfaces of WoT RMF

8.3.1 WoT broker management functional entity (WoT-BM FE)

The WoT-BM FE is responsible for registering information of each WoT broker function with related information such as location, owner, their services, etc.

The WoT broker functions are responsible for providing physical devices as the web resources. The WoT broker functions reside in the WoT resource support functions area. However, there can be a large number of brokers and each WoT broker provides a number of different web resources. Therefore, in order to provide efficient WoT service management of the WoT a broker is necessary.

8.3.2 Web resource management functional entity (Web-RM FE)

The Web-RM FE is responsible for managing the web resource with the resource information repository.

There are many types of content on the web such as image, video, music, documents, weather information, and text as a virtual things. These virtual things are ready to be exposed on the web and the WoT service provider uses them to create new WoT services. The Web-RM FE interworks with the resource information repository to manage these virtual things.

8.3.3 WoT enrolment functional entity (WoT-EN FE)

The WoT-EN FE provides a capability to support the WoT service enrolment, e.g., submission and registration. The web resource functions submit information of the web-enabled things to the WoT-EN FE. The WoT-EN FE checks and filters them using the enrolment policy.

The WoT-EN FE also manages the enrolment policy. The WoT-EN FE updates and deletes the enrolment policy according to the changes on the information and context of the WoT service. The following are functions of the WoT-EN FE:

− Submission handling: The WoT-EN FE receives the enrolment request from the WoT enabled things. For the request, these WoT services can submit their information such as IP, descriptions, name, URI, capability, related services, etc. Then the WoT-EN FE allows or blocks the request based on the enrolment policy.

– Enrolment policy: The enrolment policy contains the rules of allowance for the web-enabled things. The WoT-EN FE manages the enrolment policy according to the changes of related entities such as the WoT service, service providers, users, etc.

– Enrolment management: The WoT-EN FE has the capability for registration and management of the submitted web-enabled things to the resource information repository. The changes on the information and the context of the registered web-enabled things induce change of information of the resource information repository. The WoT-EN FE re-examines the changed web-enabled things with enrolment policy rules. It means that the WoT-EN FE can remove the registered web-enabled things beforehand from the resource information repository. The WoT-EN FE can allow the blocked WoT service to register in the repository if those are acceptable based on the changed enrolment policy.

8.3.4 WoT description management functional entity (WoT-DM FE)

The WoT-DM FE provides a description method and it manages all description information related to things. Also, it provides capabilities for translating descriptions among the WoT services.

There are many kinds of things that exist and all things need a mechanism to describe themselves and their services to be (automatically) discovered and used on the web. There are many description methods to utilize describing things such as web service description language (WSDL), resource description framework in attributes (RDFa), microformat, device profile for web services (DPWS) metadata, SensorML, JSON, etc. The WoT-DM FE provides the capability of description translation to cooperate among the many description methods and models supported by the different service providers.

9 Web resource functions

9.1 WoT broker functions

The WoT broker functions have a role for integrating and exposing the devices to the web. To expose and use physical devices on the web, a web server supporting the HTTP is implemented on the devices. However, every device does not satisfy these conditions because of constraints.

The WoT broker functions support communication between users of the WoT (e.g., the web clients, applications) and fully-fledged devices as well as constrained devices. The WoT broker functions enable the seamless integration of a device onto the web. The agent of the WoT broker has a role to control and communicate with physical devices.
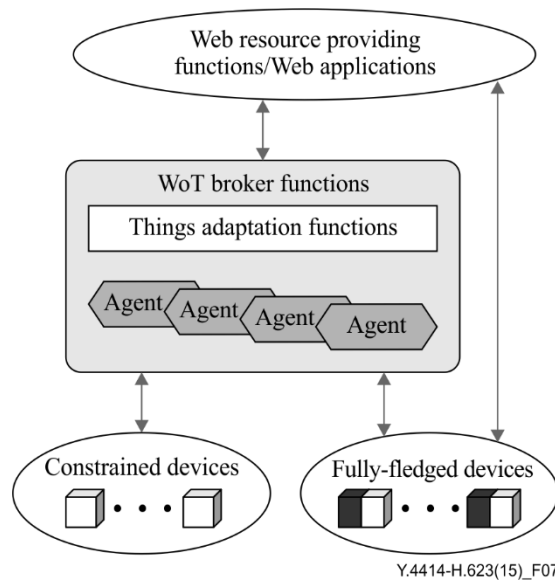
Figure 7 – Overview of WoT broker function architecture

NOTE – Detailed functions and architecture of the WoT broker function are given in [ITU-T Y.4400].

There are several ways to interact between a WoT broker and devices such as http-based connection and web services:

– Web services: enable business entities and applications to intercommunicate openly with each other over a network. Web service systems have program-language-independent properties, use message-driven communication, and are easily bound to different transport protocols [ITU-T Y.2232]. The basic web services for WoT define an interaction [b-W3C SOAP 0] [b-W3C SOAP 1] between WoT broker and constraint/fully-fledged device as an exchange of messages.

– In the web services, a service and services descriptions are described [b-W3C WSDL 0]. A service is enabled by a software module deployed on network-accessible platforms provided by the service provider. The service description contains the details of the interface and implementation of the service including the data types, operations, binding information and network location. The service description is supported by a web services provider.

– Also, in order for an application to take advantage of web services, three behaviours must take place: the publication of service descriptions, the finding and retrieval of service descriptions, and the binding or invoking of services based on the service description. These behaviours can occur singly or iteratively, with any cardinality between the roles. In detail, these operations are the execution (binding or invoking) of services based on the service description, finding and retrieval of service descriptions and publication of service descriptions.

The WoT broker is a web services provider who provides a capability for use to create web services. The web client and web service application function are the web services requester who use the services provided by a WSP.

9.2 RESTful web services

The RESTful web services have some efficient properties such as high scalability achieved by the result of a loosely coupled design and facility for deployment since it builds up on the web infrastructure and standards.

The REST includes the primacy of resources, identified using URIs, and a uniform interface generally implemented by the HTTP protocol. Resources are manipulated through representations portrayed according to a media type (e.g., HTML, Atom, etc.) and some metadata.

The WoT uses HTTP as an application protocol rather than using it as a transport protocol. The WoT exposes its interface following the REST principle because these technical characteristics enable services and applications on the WoT to be built as loosely coupled systems which expose the application programming interface (API) in a uniform manner.

Appendix I

WoT description model

(This appendix does not form an integral part of this Recommendation.)

This appendix provides the following WoT description models.

I.1 Physical characteristics in WoT description

I.1.1 Device description

The device description provides information about things itself and its manufacturers. As a result, browsers, search engines and applications discovery by browsing the product description will be able to render its UI and visualization in an enhanced manner. Device description may have the following general components:

– unique ID

– name

– brand

– manufacturer

– description

– device picture

– tags and

– authoritative information URL.

I.1.2 Location description

The location description provides the physical location information of things. The user and application get the current latitude and longitude of the things. Also they get address, postal code, country name, street information etc.

I.1.3 Owner description

The owner description provides the current owner of things.

I.2 Service characteristics in WoT description

I.2.1 Quality of service description

The quality of service description covers parameters such as bandwidth, up-time, average response time it helps taking the right decision. In a WoT populated by billions of things quality of service information can be of great help to choose the right things for the right application. These data can be based on monitoring services or provided by the manufacturer of device and service provider.

I.2.2 Types of services description

The type of services description covers basic information required to describe things on the web. It contains a description of services that a thing offers (e.g., sensor information, multimedia service, etc.).

Appendix II

An example of information flows

(This appendix does not form an integral part of this Recommendation.)

This appendix describes information flows for providing WoT services. The information flow shows an example of a use case of how to provide a WoT service or how to access things.

II.1 Information flow between web client and things

Figure II.1 shows one of the information flows relating to using WoT services. In this case, the web client wants to get a WoT service which consists of several kinds of things. At first, the web client sends a request for the WoT service to the WoT service functions using web technology (e.g., HTTP). And then the WoT service functions can access things through interworking with WoT broker functions. The detailed operation with information flow is followed as below.
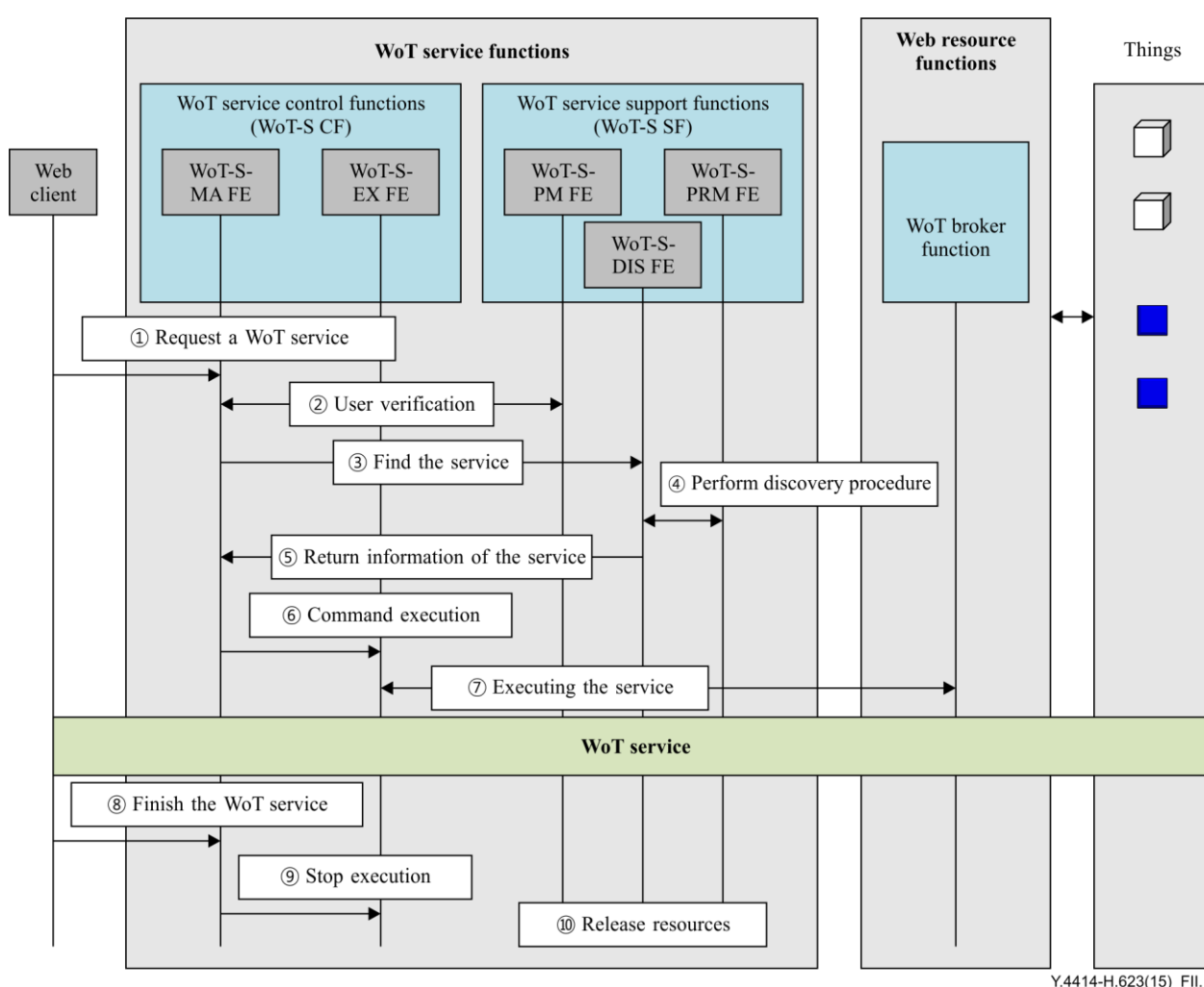
Figure II.1 – Information flow between end user and things

1) Web client requests a WoT service to the WoT-S-MA FE.

2) The WoT-S-MA FE (WoT service management FE) performs user verification with the WoT-S-PM FE (WoT service policy management FE). The WoT-S-MA FE sends a message to the WoT-S-PM FE to verify the user. The WoT-S-PM FE checks user validation and authentication to see whether the user is authorized to access the service. If the user does not have authorization, the WoT-S-PM FE returns a failure message.

3)   When the authentication procedure is finished successfully, the WoT-S-MA FE sends a message to the WoT-S-DIS FE to discover the service.

4)   The WoT-S-DIS FE performs a search procedure with the WoT-S-PRM FE (WoT service profile management FE). The WoT-S-PRM FE searches for the WoT service and retrieves information from the WoT-S repository.

5)   When the search procedure is finished successfully, the WoT-S-DIS FE returns information of the WoT service to the WoT-S-MA FE.

6)   The WoT-S-MA FE requests executing the WoT service to the WoT-S-EX FE (WoT service execution FE).

7)   The WoT-S-EX FE executes the service, interworking between the WoT-S-EX FE and the WoT broker function, to use/access things.

8)   The web client informs the WoT-S-MA FE to finish the WoT service.

9)   The WoT-S-MA FE commands the WoT-S-EX FE to stop executing the WoT service.

10) The WoT-S-EX FE releases resources.

Bibliography

[b-IETF RFC 3986]   IETF RFC 3986 (2005), *Uniform Resource Identifiers (URI): Generic Syntax.*
https://tools.ietf.org/html/rfc3986

[b-W3C digloss]   Glossary of Terms for Device Independence (2005)
http://www.w3.org/TR/di-gloss/

[b-W3C SOAP 0]   W3C Recommendation (2007), *SOAP Version 1.2 Part 0: Primer (Second Edition).*
http://www.w3.org/TR/soap12-part0/

[b-W3C SOAP 1]   W3C Recommendation (2007), *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition).*
http://www.w3.org/TR/soap12-part1/

[b-W3C WACterms]   W3C et (1999) – *Web Characterization Terminology & Definitions Sheet*.
http://www.w3.org/1999/05/WCA-terms/

[b-W3C webarch]   Jacobs, I & Walsh N (2004), *Architecture of the World Wide Web*, Volume One, 2004.
http://www.w3.org/TR/webarch/

[b-W3C WSDL 1]   W3C Recommendation (2007), *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language.*
http://www.w3.org/TR/wsdl20/

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence	H.420–H.429
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	**H.620–H.629**
Ubiquitous sensor network applications and Internet of Things	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	H.700–H.719
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	H.820–H.859
Multimedia e-health data exchange services	H.860–H.869

For further details, please refer to the list of ITU-T Recommendations.

SERIES OF ITU-T RECOMMENDATIONS

Series A   Organization of the work of ITU-T

Series D   General tariff principles

Series E   Overall network operation, telephone service, service operation and human factors

Series F   Non-telephone telecommunication services

Series G   Transmission systems and media, digital systems and networks

Series H   Audiovisual and multimedia systems

Series I   Integrated services digital network

Series J   Cable networks and transmission of television, sound programme and other multimedia signals

Series K   Protection against interference

Series L   Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant

Series M   Telecommunication management, including TMN and network maintenance

Series N   Maintenance: international sound programme and television transmission circuits

Series O   Specifications of measuring equipment

Series P   Terminals and subjective and objective assessment methods

Series Q   Switching and signalling

Series R   Telegraph transmission

Series S   Telegraph services terminal equipment

Series T   Terminals for telematic services

Series U   Telegraph switching

Series V   Data communication over the telephone network

Series X   Data networks, open system communications and security

Series Y   Global information infrastructure, Internet protocol aspects and next-generation networks

Series Z   Languages and general software aspects for telecommunication systems