International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

F.511

(04/2014)

SERIES F: NON-TELEPHONE TELECOMMUNICATION SERVICES

Directory services

Directory service – Support of tag-based identification services

Recommendation ITU-T F.511

International Telecommunication Union

ITU-T F-SERIES RECOMMENDATIONS

NON-TELEPHONE TELECOMMUNICATION SERVICES

TELEGRAPH SERVICE	
Operating methods for the international public telegram service	F.1–F.19
The gentex network	F.20–F.29
Message switching	F.30–F.39
The international telemessage service	F.40–F.58
The international telex service	F.59–F.89
Statistics and publications on international telegraph services	F.90–F.99
Scheduled and leased communication services	F.100–F.104
Phototelegraph service	F.105–F.109
MOBILE SERVICE	
Mobile services and multidestination satellite services	F.110–F.159
TELEMATIC SERVICES	
Public facsimile service	F.160–F.199
Teletex service	F.200–F.299
Videotex service	F.300–F.349
General provisions for telematic services	F.350–F.399
MESSAGE HANDLING SERVICES	F.400–F.499
DIRECTORY SERVICES	**F.500–F.549**
DOCUMENT COMMUNICATION	
Document communication	F.550–F.579
Programming communication interfaces	F.580–F.599
DATA TRANSMISSION SERVICES	F.600–F.699
AUDIOVISUAL SERVICES	F.700–F.799
ISDN SERVICES	F.800–F.849
UNIVERSAL PERSONAL TELECOMMUNICATION	F.850–F.899
HUMAN FACTORS	F.900–F.999

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

Recommendation ITU-T F.511

Directory service – Support of tag-based identification services

Summary

Recommendation ITU-T F.511 provides guidance for the use of directory services to support tag-based identification applications by reference to the directory capabilities specified in the ITU-T X.500 series of Recommendations | ISO/IEC 9594 (all parts), and in the lightweight directory access protocol (LDAP) specifications as developed within Internet Engineering Task Force (IETF). A tag is also called an automatic identification and data capture (AIDC) media. It holds an identifier that identifies the item to which the AIDC media is affixed or associated. The directory may be used to store information associated with the AIDC media.

This Recommendation identifies two cases: one case is where the identifier is used as a whole to access a centralized directory, and the other case is where the structure of the identifier is explored to access distributed directory systems, when it is not feasible for a specific environment to hold all relevant information in a single directory. In the latter situation the top-level information could be held by some types of independent service providers, while the company and/or item-related information may be held by the information owner.

The primary focus is on radio frequency identification (RFID) tags as specified in the GS1 EPCglobal specifications and in ISO and ISO/IEC International Standards.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T F.511	2014-04-06	17	11.1002/1000/12143

* 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.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

		Page
1	Scope	1
2	References	1
3	Definitions	3
	3.1 Terms defined elsewhere	3
	3.2 Terms defined in this Recommendation	4
4	Abbreviations and acronyms	4
5	Conventions	6
6	Involved standards organizations	6
	6.1 GS1 EPCglobal	6
	6.2 ISO and ISO/IEC	6
7	AIDC media and associated information	7
8	Access to associated information using directory technology	7
9	Unique item identifier (UII) and electronic product code (EPC)	8
	9.1 General	8
	9.2 Electronic product code (EPC) types	9
	9.3 Unique item identifier (UII) types	9
	9.4 Global uniqueness of EPCs and UIIs	9
10	Locating associated information for centralized AIDC identification applications	9
	10.1 Search using EPC in a centralized environment	10
	10.2 Search using UII together with AFI in a centralized environment	11
11	Issues for a distributed AIDC environment	11
12	EPC and UII formatting and URN subtree structure	13
	12.1 EPC and UII formatting information	13
	12.2 EPC and UII represented by URN directory subtrees	14
13	EPC/UII format information	17
	13.1 EPC/UII format DIT subtree	17
	13.2 EPC format information	17
	13.3 UII format information	17
14	Support of ITU-T Y.2213	19
	14.1 General	19
	14.2 Multiple user groups for same EPC or UII	20
	14.3 Reverse identifier resolution	20
	14.4 Location-based information	20
	14.5 Avoidance of single-point-of-failure	20

Page

Appendix I – Use of OIDs in an RFID environment 21
I.1 Introduction 21
I.2 OID used in the ISO environment 21
I.3 ISO/IEC 15459 OID structure vs. data identifiers/application identifiers 22
Appendix II – Introduction to ITU-T X.500-series/LDAP concepts 23
II.1 Directory entries and their content 23
II.2 Directory information tree (DIT) 23
II.3 Directory subtrees 23
II.4 Distinguished names and URNs 24
II.5 Directory search operation 24
II.6 Entry information selection 25
II.7 Pointers to other directories (referrals) 25
Appendix III – Introduction to RFID tag structure 26
III.1 Scope of appendix 26
III.2 ISO/IEC 18000-6C and ISO/IEC 18000-3m3 RFID tag structure 26
III.3 Bank '00'B (reserved memory) 26
III.4 Bank '01'B (UII/EPC memory) 27
III.5 Bank '10'B (TID memory) 27
III.6 Bank '11'B (user memory) 28
Appendix IV – Overview of EPC types 30
IV.1 Scope of appendix 30
IV.2 Structure of serialized global trade item number (SGTIN) EPC types 30
IV.3 Structure of serial shipping container code (SSCC) EPC types 31
IV.4 Structure of serialized global location number (SGLN) EPC types 32
IV.5 Structure of global returnable asset identifier (GRAI) EPC types 32
IV.6 Structure of global individual asset identifier (GIAI) EPC types 33
IV.7 Structure of global service relation number (GSRN) EPC types 33
IV.8 Structure of global document type identifier (GDTI) EPC types 33
Appendix V – Example of retrieving EPC format information 35
Appendix VI – Overview of ISO/IEC 15459 UII types 38
VI.1 Overview 38
VI.2 Data identifier field 38
VI.3 Issuing agency codes (IAC) 39
VI.4 Company identification number (CIN) 40
VI.5 Remaining fields 40
Appendix VII – Examples of retrieving UII format information 41
VII.1 Scope of appendix 41
VII.2 IAC=UN (Dun and Bradstreet) 41
VII.3 IAC=J example (UPU) 42

Page

Appendix VIII – The format attribute type 44
VIII.1 Introduction 44
VIII.2 EPC format information 44
VIII.3 UII format information 44
Bibliography 46

Recommendation ITU-T F.511

Directory service – Support of tag-based identification services

1 Scope

The scope of this Recommendation is to provide sufficient information for the use of directories to support tag-based identification applications without having to study the details of the many specifications on tag-based identification. It describes how this support may be provided using systems supporting the ITU-T X.500 | ISO/IEC 9594 series of Recommendations. The capabilities required are primarily limited to those that are also supported by the lightweight directory access protocol (LDAP) specifications.

NOTE 1 – Use of LDAP-based systems provides low-cost solutions.

For environments that are more complex this Recommendation also provides guidance on how directory responsibilities may be split between general service providers and information owners.

This Recommendation is only concerned with tag-based identification as related to automatic identification and data capture (AIDC) media as specified in numerous ISO/IEC and ISO International Standards and in GS1 specifications. This Recommendation is limited to the following types of AIDC media:

– The ISO/IEC 18000-6C tag type as defined in [ISO/IEC 18000-63]. This tag type is identical to the GS1 Class 1 Gen 2 tag type.

– The ISO/IEC 18000-3 Mode 3 tag type as defined in [ISO/IEC 18000-3].

NOTE 2 – Support for other types of AIDC media may be added in the future.

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 X.500-series] Recommendation ITU-T X.5xx (2012) | ISO/IEC 9594-x:2014 series, *Information technology – Open Systems Interconnection – The Directory.*

[ITU-T X.500] Recommendation ITU-T X.500 (2012) | ISO/IEC 9594-1:2014, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services*.

[ITU-T X.501] Recommendation ITU-T X.501 (2012) | ISO/IEC 9594-2:2014, *Information technology – Open Systems Interconnection – The Directory: Models*.

[ITU-T X.509] Recommendation ITU-T X.509 (2012) | ISO/IEC 9594-8:2014, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks*.

[ITU-T X.511] Recommendation ITU-T X.511 (2012) | ISO/IEC 9594-3:2014, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition*.

[ITU-T X.518] Recommendation ITU-T X.518 (2012) | ISO/IEC 9594-4:2014, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation*.

[ITU-T X.519] Recommendation ITU-T X.519 (2012) | ISO/IEC 9594-5:2014, *Information technology – Open Systems Interconnection – The Directory: Protocol specifications*.

[ITU-T X.520] Recommendation ITU-T X.520 (2012) | ISO/IEC 9594-6:2014, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types*.

[ITU-T X.521] Recommendation ITU-T X.521 (2012) | ISO/IEC 9594-7:2014, *Information technology – Open Systems Interconnection – The Directory: Selected object classes*.

[ITU-T X.525] Recommendation ITU-T X.525 (2012) | ISO/IEC 9594-9:2014, *Information technology – Open Systems Interconnection – The Directory: Replication*.

[ITU-T X.680] Recommendation ITU-T X.680 (2008) | ISO/IEC 8824-1:2008, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*.

[ITU-T Y.2213] Recommendation ITU-T Y.2213 (2008), *NGN service requirements and capabilities for network aspects of applications and services using tag-based identification*.

[IETF RFC 3061] IETF RFC 3061 (2001), *A URN Namespace of Object Identifiers*.

[IETF RFC 4510] IETF RFC 4510 (2006), *Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map*.

[IETF RFC 4511] IETF RFC 4511 (2006), *Lightweight Directory Access Protocol (LDAP): The Protocol*.

[IETF RFC 4516] IETF RFC 4516 (2006), *Lightweight Directory Access Protocol (LDAP): Uniform Resource Locator*.

[IETF RFC 5134] IETF RFC 5134 (2008), *A Uniform Resource Name Namespace for the EPCglobal Electronic Product Code (EPC) and Related Standards*.

[ISO/IEC 15459 S] ISO/IEC 15459 all parts, *Information technology – Unique identifiers*.

[ISO/IEC 15459-1] ISO/IEC 15459-1:2006, *Information technology – Unique identifiers – Part 1: Unique identifiers for transport units*.

[ISO/IEC 15459-2] ISO/IEC 15459-2:2006, *Information technology – Unique identifiers – Part 2: Registration procedures*.

[ISO/IEC 15459-3] ISO/IEC 15459-3:2006, *Information technology – Unique identifiers – Part 3: Common rules for unique identifiers*.

[ISO/IEC 15459-4] ISO/IEC 15459-4:2008, *Information technology – Unique identifiers – Part 4: Individual items*.

[ISO/IEC 15459-5] ISO/IEC 15459-5:2007, *Information technology – Unique identifiers – Part 5: Unique identifiers for returnable transport items (RTIs)*.

[ISO/IEC 15459-6] ISO/IEC 15459-6:2007, *Information technology – Unique identifiers – Part 6: Unique identifier for product groupings*.

[ISO/IEC 15961-1] ISO/IEC 15961-1:2013, *Information technology – Radio frequency identification (RFID) for item management: Data protocol – Part 1: Application interface*.

[ISO 17363] ISO 17363:2013, *Supply chain applications of RFID – Freight containers*.

[ISO 17364] ISO 17364:2013, *Supply chain applications of RFID – Returnable transport items (RTIs) and returnable packaging items (RPIs)*.

[ISO 17365] ISO 17365:2013, *Supply chain applications of RFID – Transport units*.

[ISO 17366] ISO 17366:2013, *Supply chain applications of RFID – Product packaging*.

[ISO 17367] ISO 17367:2013, *Supply chain applications of RFID – Product tagging*.

[ISO/IEC 18000-3] ISO/IEC 18000-3:2010, *Information technology – Radio frequency identification for item management – Part 3: Parameters for air interface communications at 13,56 MHz*.

[ISO/IEC 18000-6] ISO/IEC 18000-6:2013, *Information technology – Radio frequency identification for item management – Part 6: Parameters for air interface communications at 860 MHz to 960 MHz General*.

[ISO/IEC 18000-63] ISO/IEC 18000-63:2013, *Information technology – Radio frequency identification for item management – Part 6: Parameters for air interface communications at 860 MHz to 960 MHz Type C*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 attribute [ITU-T X.501]: Information of a particular type. Entries are composed of attributes.

3.1.2 attribute type [ITU-T X.501]: That component of an attribute which indicates the class of information given by that attribute.

3.1.3 attribute value [ITU-T X.501]: A particular instance of the class of information indicated by an attribute type.

3.1.4 attribute value assertion [ITU-T X.501]: A proposition, which may be true, false, or undefined, according to the specified matching rules for the type, concerning the presence in an entry of an attribute value of a particular type.

3.1.5 directory information base (DIB) [ITU-T X.501]: The complete set of information to which the Directory provides access, and which includes all of the pieces of information which can be read or manipulated using the operations of the Directory.

3.1.6 directory information tree (DIT) [ITU-T X.501]: The DIB considered as a tree, whose vertices (other than the root) are the Directory entries.

NOTE – The term "DIT" is used instead of "DIB" only in contexts where the tree structure of the information is relevant.

3.1.7 distinguished name (of an entry) [ITU-T X.501]: The name of an entry which is formed from the sequence of the relative distinguished names (RDNs) of the entry and each of its superior entries. Every object entry, alias entry and subentry has precisely one distinguished name.

3.1.8 filter [ITU-T X.511]: An assertion about the presence or value of certain attributes of an entry in order to limit the scope of a search.

3.1.9 filter item [ITU-T X.511]: A filter item is an assertion about the presence or value(s) of attributes in the entry under test.

3.1.10 identifier [b-ITU-T Y.2091]: An identifier is a series of digits, characters and symbols or any other form of data used to identify subscriber(s), user(s), network element(s), function(s), network entity(ies) providing services/applications, or other entities (e.g., physical or logical objects). Identifiers can be used for registration or authorization. They can be either public to all networks, shared between a limited number of networks or private to a specific network (private IDs are normally not disclosed to third parties).

3.1.11 relative distinguished name (RDN) [ITU-T X.501]: A set of one or more attribute type and value pairs, each of which matches a distinct distinguished attribute value of the entry.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 associated information: This is the information external to an AIDC media that is associated with the identifier on the AIDC media. The associated information may be a movie, information about a parcel, manufacture contact information, product specification, or it could be anything else.

3.2.2 automatic identification and data capture (AIDC): Data input without typing. Data is captured automatically usually using equipment such as barcode readers, magnetic-stripe readers, RFID tag readers, etc.

3.2.3 automatic identification and data capture (AIDC) client system: A component that formats information retrieved by an ID terminal into either a DAP request or an LDAP request.

3.2.4 automatic identification and data capture (AIDC) media: A media that may hold information that has been retrieved using AIDC techniques.

3.2.5 electronic product code (EPC): Information typically encoded in a radio frequency identification tag to uniquely identify a physical object (e.g. a unique item identifier (UII).

3.2.6 EPCglobal environment: The context within GS1 EPCglobal specifications for generating radio frequency identification – automatic identification and data capture (RFID-AIDC) media specifications.

3.2.7 identity (ID) terminal: A device with a data reading and optional writing capability which reads (and optionally writes) identifier(s) and possible application data from/to an automatic identification and data capture (AIDC) media. It is a common term for radio frequency identification (RFID) tag readers, barcode readers, etc.

3.2.8 ISO environment: The context established by ISO or ISO/IEC for generating automatic identification and data capture (AIDC) media specification.

3.2.9 ITU-T X.500 system: An infrastructure consisting of one or more interconnected directory system agents (DSAs) and zero or more lightweight directory access protocol (LDAP) servers.

3.2.10 RFID tag: Automatic identification and data capture (AIDC) media that can be queried by means of a suitably modulated inductive or radiating electromagnetic signal to transfer information to a centralized tag-based information system. (in some specifications this is also called an RF tag).

3.2.11 unique item identifier (UII): An identifier as defined in the ISO environment and which is unique within a specific context.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AFI Application Family Identifier

AI	Application Identifier
AIDC	Automatic Identification and Data Capture
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation One
CAGE	Commercial And Governmental Entity
CFR	Code of Federal Regulations
CIN	Company Identification Number
CRC	Cyclic Redundancy Check
DAP	Directory Access Protocol
DI	Data Identifier
DIB	Directory Information Base
DIT	Directory Information Tree
DoD	Department Of Defence
DSA	Directory System Agent
DSFID	Data Storage Format Identifier
DUA	Directory User Agent
EDI	Electronic Data Interchange
EOT	End Of Transmission
EPC	Electronic Product Code
GDTI	Global Document Type Identifier
GIAI	Global Individual Asset Identifier
GID	General Identifier
GRAI	Global Returnable Asset Identifier
GSRN	Global Service Relation Number
IAC	Issuing Agency Code
ID	Identity
LDAP	Lightweight Directory Access Protocol
MDID	Mask-Designer Identification Code
NEN	Nederlands Normalisatie-instituut
NCAGE	NATO Commercial And Governmental Entity
NGN	Next-Generation Network
OID	Object Identifier
RAID	Redundant Arrays of Inexpensive Disks
RDN	Relative Distinguished Name
RFID	Radio Frequency Identification
RTI	Returnable Transport Item
SGLN	Serialized Global Location Number

SGTIN	Serialized Global Trade Item Number
SSCC	Serial Shipping Container Code
SSL	Secure Socket Layer
TID	Tag Identifier
TLS	Transport Layer Security
TMN	Tag Model Number
UII	Unique Item Identifier
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
VIN	Vehicle Identification Number
XML	extensible Markup Language
XPC	extended Protocol Control

5 Conventions

The term ITU-T X.500 system is used to denote an infrastructure consisting of one or more ITU-T X.500 directory service agents (DSAs) and zero or more lightweight directory access protocol (LDAP) servers, where a DSA provides the access point for an accessing client. The term LDAP server is used to denote a system according to [IETF RFC 4510]. A directory system is either an ITU-T X.500 system or a single LDAP server.

NOTE – A DSA is a directory server that follows the specifications in the ITU-T X.500 series of Recommendations [ITU-T X.500-series].

The ITU-T X.500 directory specifications are used to denote the specifications as given by [ITU-T X.500-series].

[ITU-T X.680] notations for character strings, hexadecimal strings and binary strings are used by this Recommendation, i.e., a character string, including a pure numeric string, is surrounded by double quotes (e.g., "string" or "12345"), a hexadecimal string is surrounded by single quotes followed by the letter H (e.g., 'A1'H) and a binary string is surrounded by single quotes followed by the letter B (e.g., '1010 0001'B). A binary number is just given as a figure without quotes.

6 Involved standards organizations

6.1 GS1 EPCglobal

GS1 is a significant organization in the development of AIDC specifications. GS1, formerly called EAN International, adopted the name GS1 in 2005. It is an international not-for-profit association. There are GS1 member organizations in many countries.

EPCglobal is part of GS1 and leads the development of industry-driven standards for the Electronic Product Code™ (EPC) to support the use of radio frequency identification (RFID).

6.2 ISO and ISO/IEC

ISO/IEC JTC 1/SC 31 is the major standards-setting body in AIDC standardization and is responsible for a large number of international standards in the AIDC area.

ISO TC 104 and ISO TC 122 are other committees involved in AIDC standardization.

7 AIDC media and associated information

AIDC is a technique where information is stored on some kind of media called an AIDC media. Examples of AIDC media are radio frequency identification (RFID) tags, barcodes, smart cards, etc.

NOTE – Only RFID tags are the subject of this Recommendation. However, to prepare for other types of tags, the term AIDC media is used in many places as a synonym for RFID tag.

There are two types of specifications for AIDC media: those as defined by GS1 EPCglobal and others as defined by ISO/IEC or ISO.

An AIDC media is affixed or associated with some item and holds a unique identification of that item (or a unique identification of the type of item).

The AIDC media may contain some information beyond the identifier, but more substantial external information may be associated with an AIDC media. Such information could be:

– description of the item
– manufacturing information
– price
– position of the item, such as postal address, coordinates or other location information
– customs information.

This information must be available in some information system and accessed based on the identifier of the AIDC media.

There are other standards that cover similar considerations as made by this Recommendation, such as:

– [b-ITU-T F.771] gives some high-level specification for how information retrieved from an AIDC media may be transformed in such a way that multimedia information may be retrieved from some repository.
– [b-ITU-T H.621] defines a system architecture for multimedia information access. The model defined in this Recommendation can be mapped in the model defined in [b-ITU-T H.621] with the difference that the identification scheme for access can be deducted from the AIDC media information without additional access, as required by [b-ITU-T H.621].
– [b-ITU-T H.642.1] defines an alternative identification scheme for accessing multimedia information.

8 Access to associated information using directory technology

Information associated with an AIDC media may be stored in a directory infrastructure. In some cases, the associated information may be huge, such as a movie, and the directory may then provide the uniform resource locator (URL) for that information.

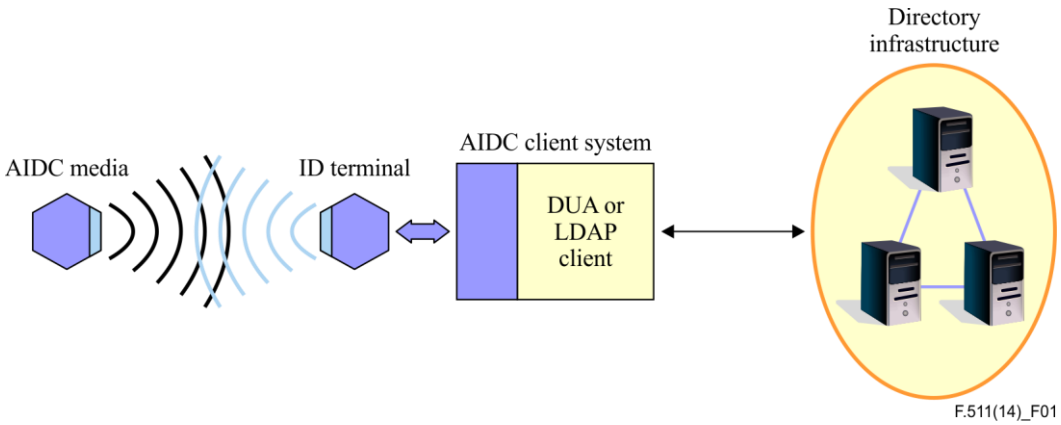

Figure 1 – AIDC use of directory technology

Figure 1 shows how the identifier on an AIDC media may be used to access a directory infrastructure to retrieve the information associated with the AIDC media. The identifier is read from the AIDC media and forwarded to the directory, typically within a search operation, to retrieve the associated information.

The AIDC client system is some intermediate system that takes the information from the identity (ID) terminal (e.g., a RFID tag reader) and converts that information into a directory request acting either as an LDAP client or as a directory user agent (DUA).

NOTE – DUA is a directory client using the directory access protocol (DAP) according to the specifications in [ITU-T X.500-series].

The AIDC client system could be integrated in the ID terminal or it could be a separate system, e.g., a lap-top computer.

A directory system may be an LDAP system consisting of a single LDAP server or it may be an ITU-T X.500 system defined as an interconnection of one of more DSAs and zero or more LDAP servers.

Appendix II provides an introduction to [ITU-T X.500-series]/LDAP and contains sufficient material to obtain a reasonable understanding of the specification principles established by this Recommendation.

9 Unique item identifier (UII) and electronic product code (EPC)

9.1 General

An AIDC media that holds an identifier in the EPCglobal environment is called an electronic product code (EPC) and in the ISO environment it is called a unique item identifier (UII).

As stated in the scope clause, only the RFID tag type defined as ISO/IEC 18000-6C or as ISO/IEC 18000-3 Mode 3 is considered by this Recommendation. Appendix III gives an overview of this tag type.

As described in Appendix III, such a tag type has a bit, called the toggle switch that indicates whether the tag contains an EPC or a UII.

EPCs and UIIs have structures that allow them to be globally unique.

An EPC or a UII is a series of characters or bit fields with an implied hierarchical structure. As an example, some fields may denote a manufacturer; some fields may denote a type of product; and another field may be a serial number.

9.2 Electronic product code (EPC) types

The different EPC formats are described in Appendix IV.

The first eight bits of an EPC is the header and indicates the EPC type. An escape mechanism is defined that allows multi-octet headers to be defined in the future.

9.3 Unique item identifier (UII) types

In the ISO environment many different UII types are defined and others will be further defined in the future. A UII is characterized by having an associated field called the application family identifier (AFI). The AFI field is not part of the actual UII. It is a one-octet field. There is an escape mechanism defined to provide multiple-octet AFIs in the future.

AFIs are allocated to UII types by ISO/IEC JTC 1/SC 31 for items (physical and virtual) not being persons, while ISO/IEC JTC 1/SC 17 is concerned with personal identification.

9.4 Global uniqueness of EPCs and UIIs

An EPC, as identified by the toggle switch (see clause 9.1), is globally unique.

Many different ISO or ISO/IEC committees define UII types. Accordingly, there is no guaranty that a particular UII is globally unique in itself. A UII may only be unique within a specific environment or application. Additional information is necessary to ensure global uniqueness. AFI may be read from the ID-tag or otherwise, as available. AFI together with UII provides a globally unique identification within the context of RFID tags. A reference to the specification defining the UII format is another way to achieve uniqueness.

NOTE – Alternatively to using AFI, an object identifier (OID) may be encoded on the RFID tag. Such an OID together with the UII also provides a globally unique identification.

10 Locating associated information for centralized AIDC identification applications

This Recommendation identifies two distinct scenarios:

a) A particular tag-based identification application may be limited, e.g., to a limited geographical area and it may be under a single management entity. All the information associated with the AIDC media is held by a single directory system (which could be a distributed ITU-T X.500 system). In this environment, the use of directory technology becomes rather straight forward as discussed in the remainder of this clause.

b) The situation becomes considerably more complicated when a tag-based identification application is highly distributed and several different user groups access the information associated with the AIDC media. An example could be goods produced at different location, packed in containers, transferred by trucks or trains to container docks and then carried by ship to some distant location. Containers are then forwarded by trucks or trains to the final destinations, from where the goods are distributed to different recipients. Several user groups may need to access the information associated with the AIDC media, such as the manufacturers, the different shipping companies, the custom authorities, the recipients of the goods, etc. A suggestion on how to handle this more complex information situation is given in clause 11.

10.1 Search using EPC in a centralized environment

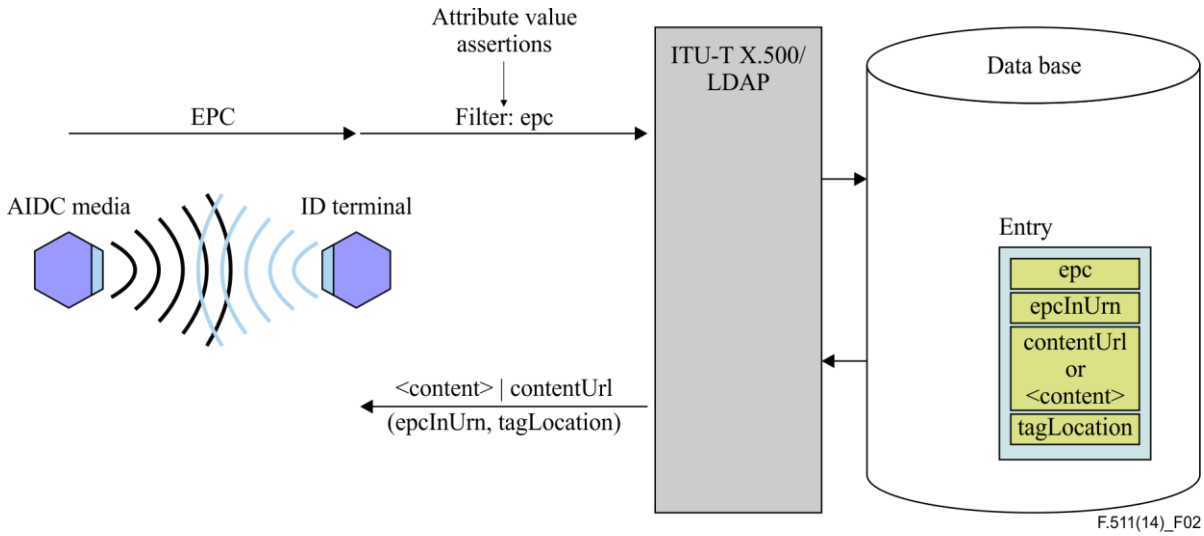

Figure 2 – Search using complete EPC as the search argument

Figure 2 illustrates the case where an ID terminal reads an ISO/IEC 18000-6C RFID tag (GS1 Class 1 Gen 2 tag type) to obtain EPC. EPC is forwarded as the argument of a directory request. The directory will search the database to locate an entry that holds an attribute of type **`epc`** with the same value as presented in the argument. This entry may hold either the information associated with the RFID tag or it may hold a URL in an attribute of type **`contentUrl`** pointing to the location where the associated information might be found. Alternatively, a pointer to another directory location may be provided in the form of a referral, e.g., to another directory server (ITU-T X.500 DSA or LDAP server).

The associated information may take many shapes. The information may be composed of several directory attributes of different types. The directory specifications have defined a large number of different attribute types for general use. New attribute types may be defined as required. This is normal for directories.

If, for some reason, the EPC is wanted in a uniform resource name (URN) format, a ready-made URN could be stored in the located entry and can be returned in an attribute of type **`epcInUrn`**.

Other information may also be available and provided, such as location information in the form of postal address, shipping company or in the form of coordinates provided in an attribute of type **`tagLocation`**.

10.2 Search using UII together with AFI in a centralized environment

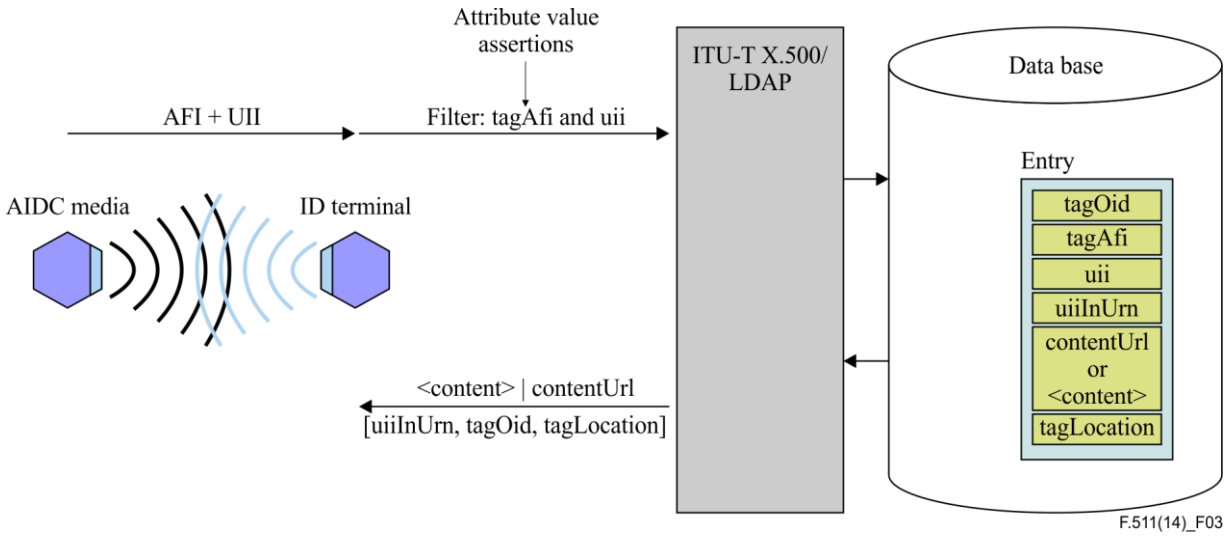

Figure 3 – Retrieval of information using AFI and UII values

Figure 3 illustrates the case where an ID terminal reads an ISO/IEC 18000-3m3 or ISO/IEC 18000-6C RFID tag to get the AFI and the UII. The tuple (AFI, UII) are forwarded in the filter of a search request. The search will locate the entry that holds those two values in the attributes of type **`tagAFI`** and **`uii`**. As discussed clause 8, this entry may hold different types of additional information. As an example, an attribute of type **`uiiInUrn`** may hold a URN representing the UII.

If an object identifier is defined for the tag type in question, the object identifier may be available in the entry in an attribute of type **`tagOid`** and may be retrieved if relevant.

NOTE – If an OID has been read from the RFID tag, it may be used instead of the AFI for the directory search.

11 Issues for a distributed AIDC environment

There are more complex environments than those described in clause 10. In an environment with many organizations involved, it may not be feasible to hold all information in a single directory system. Several companies and organizations might be interested in the same information held by some common information provider, but companies might still want to hold company specific information in their own systems. As an example, manufacturers will keep details on their products in their own database, which may or may not be an ITU-T X.500/LDAP directory system. Different organizations in the supply chain would need to obtain that information. In some environments, a cascade of directory systems may be applied, as is illustrated in Figure 4.

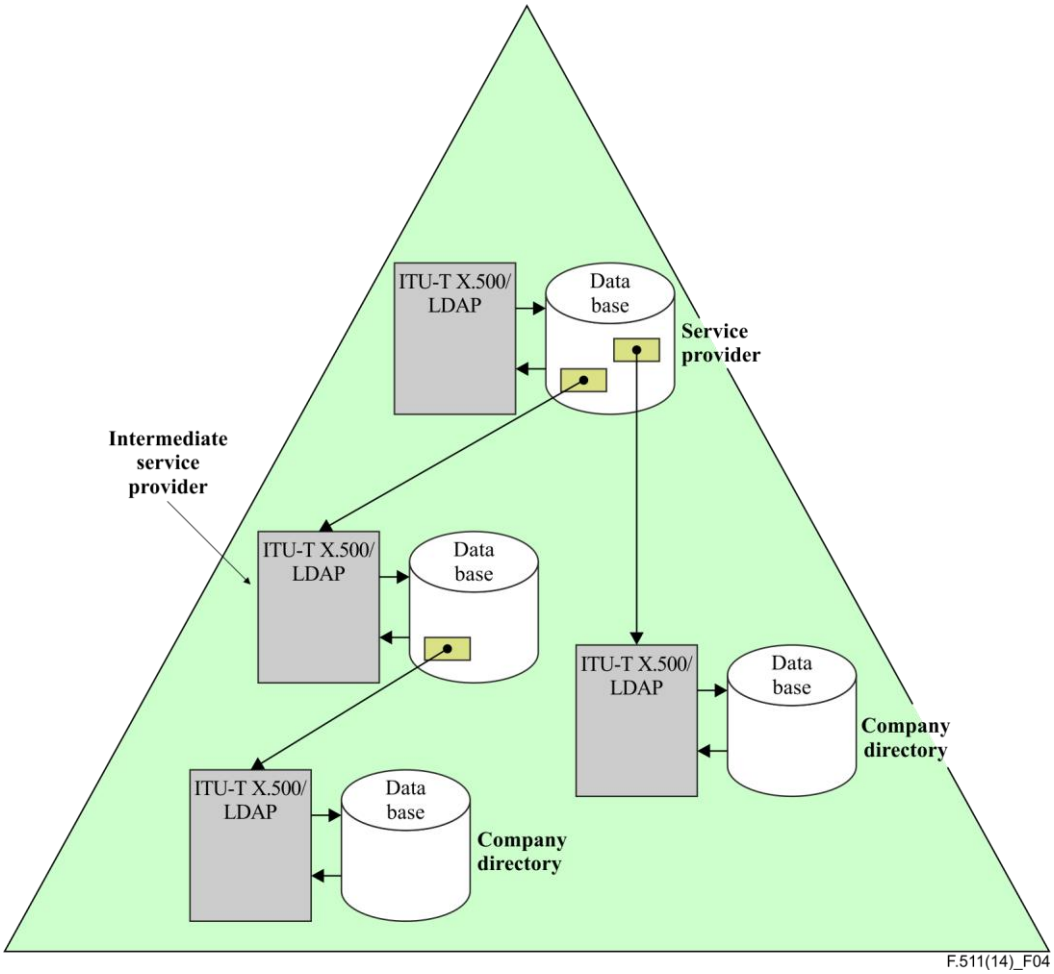

Figure 4 – Directory infrastructure for specific purpose(s)

An AIDC client system reading a specific tag may not know to which company site to access to find information about the item to which the tag is affixed.

It seems quite unlikely that one infrastructure would cover all conceivable applications. For example, defence organizations, the aviation industry or the auto industry, etc. may each create their own separate directory infrastructure. Moreover, more complex situations may not be covered by the architecture illustrated in Figure 4.

Different applications may have different security requirements. Some application may require strong authentication and encryption of the data transmission. Transport layer security (TLS) may be used in some cases and not in others.

The actual distribution of information requires a careful evaluation; however, one possible solution would be to have a directory system run by an independent service provider as illustrated in Figure 4. This directory system would then hold information about EPC/UIIs down to company information, from where a link could be provided to the company site.

To have this distribution, it is necessary to know the exact formats of EPCs/UIIs, as discussed in clause 12.1.1.

12 EPC and UII formatting and URN subtree structure

12.1 EPC and UII formatting information

12.1.1 Formatting information requirements

Formatting information about EPC or UII is required if:

– EPCs and UIIs are used in a distributed environment, as discussed in clause 11;

– a higher level of information, such as manufacturer information, is required rather than information about the particular item;

– there is need to convert the RFID encoded EPC/UII to a printed character representation; and/or

– there is need to convert the RFID encoded EPC/UII to a globally unique URN.

An AIDC client system may have sufficient built-in knowledge of the different EPC and/or UII formats to make such a conversion. However, this requires the AIDC client system to be loaded with that type of information and they have to be periodically updated to keep-up with new or updated EPC and UII types. Storing formatting information in a directory will simplify AIDC client systems and their maintenance.

This Recommendation considers two types of entries:

a) entries representing particular EPCs and UIIs. Such entries form URN subtrees as shown in clause 12.2;

b) entries that hold formatting information about EPC and UII types. Such entries form URN subtrees as shown in clause 13.

12.1.2 EPC formats

EPC types are quite complicated when it comes to converting a bit-encoded EPC to either a character-encoded EPC (a GS1 element string) or a URN representation. This is specified in more details in Appendix IV.

All EPC types have the following in common except for a type called general identifier (GID):

a) An eight-bit header that uniquely identifies the EPC type (the header length may be extended in the future).

b) A three-bit filter field indicating some characteristics of item identified by the EPC, such as a retail trade item, a standard trade item grouping, a single shipping/consumer trade item, etc.

c) A three-bit partition field giving some partitioning information.

d) A company prefix identifying the company associated with the RFID tag.

12.1.3 ISO/IEC 15459 formats

The ISO/IEC 15459 standard (all parts) specify rules for unique item identification by requiring an issuing agency code (IAC) and company identification number (CIN) to be the first part the actual UII.

NOTE – The data identifier (DI) is not part of the UII.

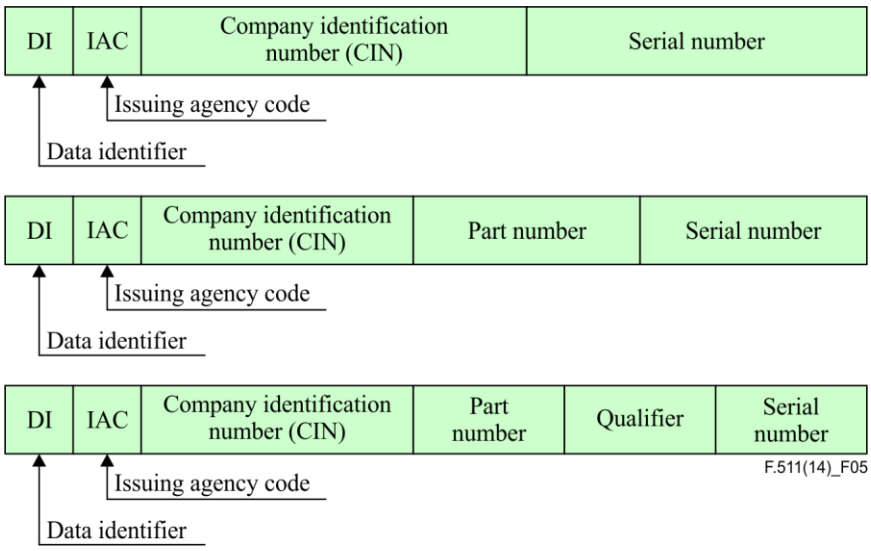

Figure 5 – ISO/IEC 15459 UII formats

Figure 5 illustrates the different formats of UIIs as defined by [ISO/IEC 15459 S]. The fields are described in detail in Appendix VI.

12.2 EPC and UII represented by URN directory subtrees

12.2.1 URN directory subtrees

Annex G of [ITU-T X.520] describes how a URN may be represented in a directory, where each URN component is represented by a directory entry. These entries have a superior and subordinate relationship corresponding to the URN structure. As related URNs have the upper level components in common, they may share entries. A set of URNs may in this way be reflected by a directory subtree (More information on directory information trees (DIT) and directory subtrees can be found in Appendix II.)

The **urnC** attribute type is used for naming of entries representing URNs, i.e., this attribute type is used for the relative distinguished name (RDN) for the entry, where the value is the corresponding URN component.

By using a subtree structure, the information associated with a URN, or possibly a truncated URN, may be obtained by a simple read operation.

NOTE 1 – LDAP does not have a read operation, but it can imitate a read operation using a simplified search operation.

NOTE 2 – A truncated URN is a URN where the last component(s) have been removed allowing reading an entry higher in the URN subtree.

12.2.2 URN subtree for EPCs

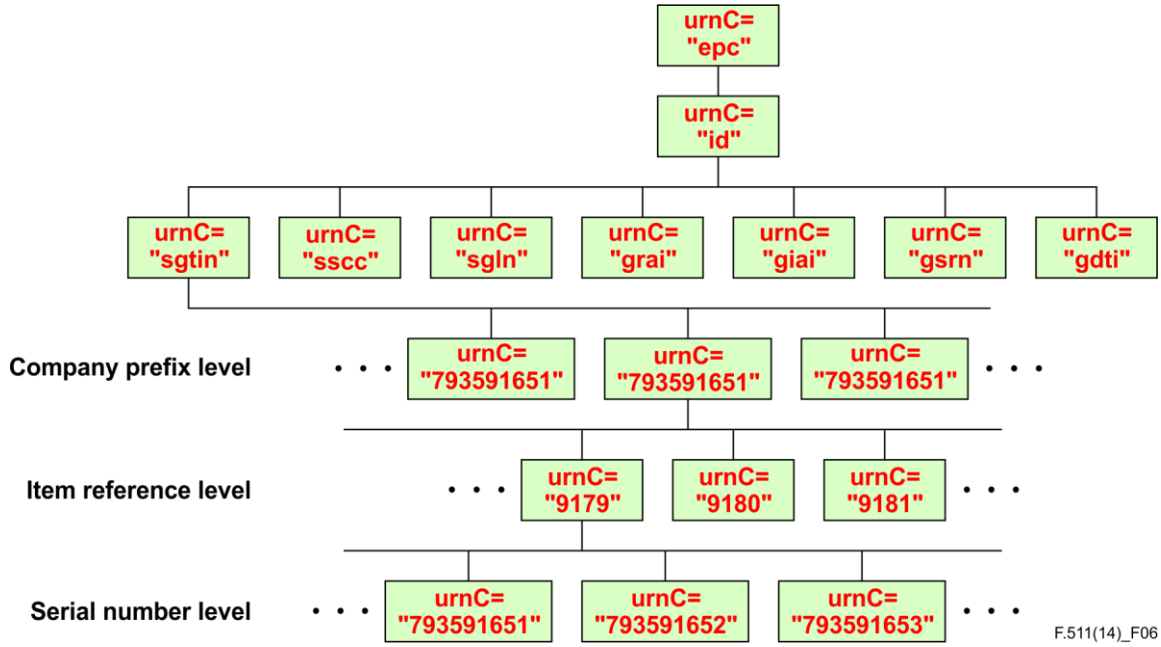

Figure 6 – URN DIT subtree for EPC

The URN structure for EPC is specified in [b-GS1TAGDAT]. Figure 6 illustrates the structure URN subtree structure for EPCs.

a) The root of this subtree is an entry with the RDN value "epc".

b) The second level of the subtree is an entry with the RDN value "id".

c) The third level of the subtree consists of entries each with the RDN value "sgtin", "sscc", "sgln", "grai", "giai", "gsrn" or "gdti", as appropriate (see Appendix IV).

d) The remaining levels of the subtree represent the company prefixes, the item references for each of the company prefix entries, and finally the serial numbers for each of the item reference entries. All these entries have RDNs with appropriate values.

The following is an example of an EPC URN:

"urn:epc:id:sgtin:793591651.2917.ThisIsASerialNumber".

12.2.3 Possible URN DIT subtree for UIIs

The URN structure specified by this Recommendation is primarily intended for directory use, but it could also be a general structure for the URN representation of an UII.

To ensure global uniqueness, it is suggested to use the URN representation as specified in [ITU-T X.520] for URNs for UIIs in accordance with [ISO/IEC 15459 S].

Figure 7 shows a possible URN subtree for an RFID application.

The root of this subtree is an entry with the RDN value "iso" indicating that it is part of the ISO URN name space.

The second level of the subtree is an entry with the RDN value "std" indicating that it the standard subspace that is used.

The third level of the subtree is an entry with the RDN value "iso-iec" indicating that the standard is an ISO/IEC standard.

The fourth level of the subtree is an entry with the RDN value "15459" indicating that we are considering UIIs defined by [ISO/IEC 15459 S].

The fifth level of the example subtree is an entry with the RDN value "25S" identifying the data identifier (see for example clause 12.1.3 and Appendix VI).

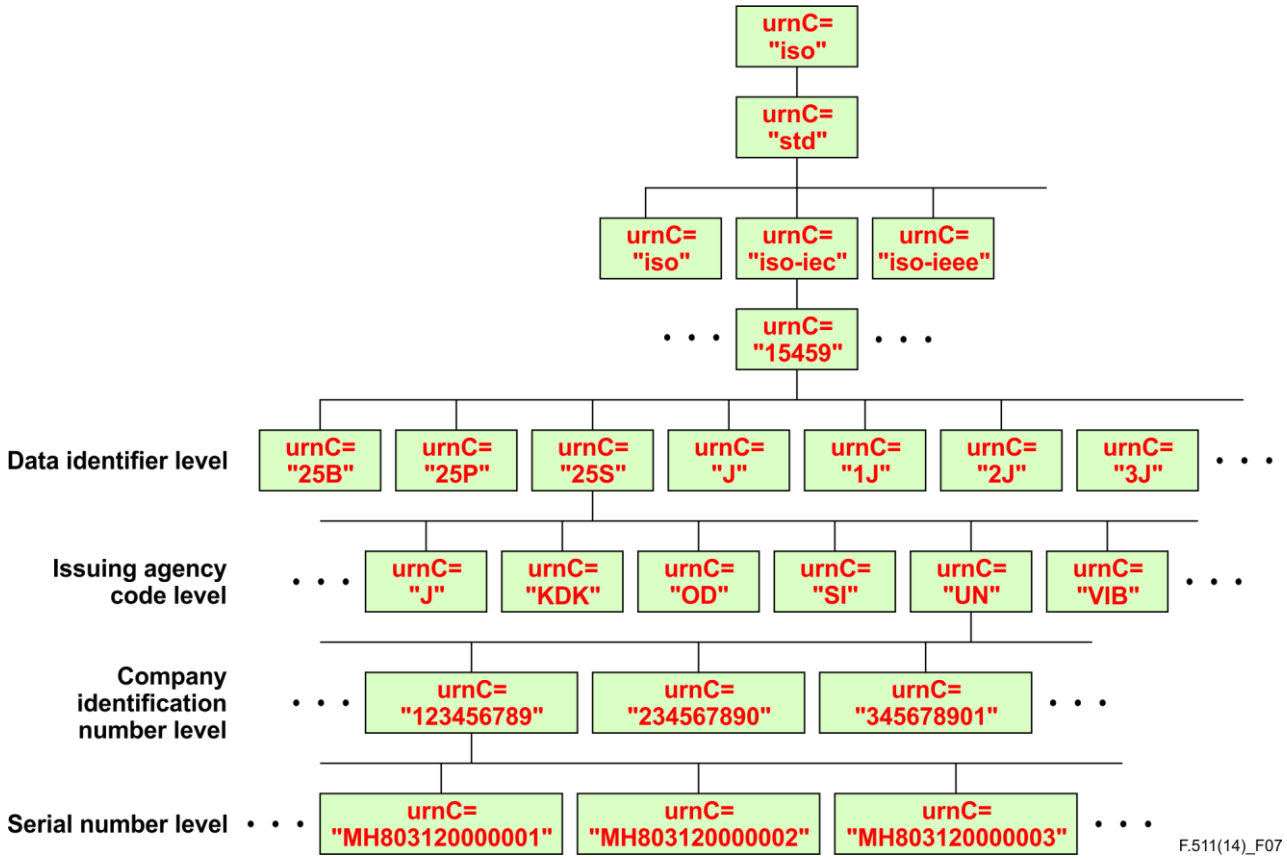

Figure 7 – Example of URN DIT subtree for UIIs

The sixth level of the example URN subtree is the issuing agency code (IAC) level. The RDN has an IAC value as allocated by the issuing agency.

The next level is the company identification number (CIN). The naming attribute **`urnC`** takes as value the IAC allocated to the company in question.

The final levels are for the fields following the CIN level. The RDN for each entry takes as value of the field in question as allocated by the company.

An example is shown below in Figure 8.

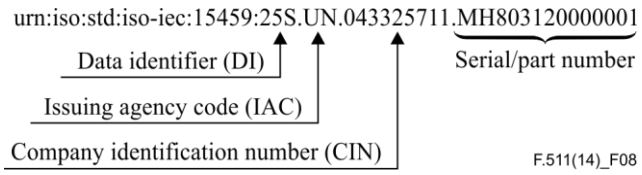

Figure 8 – URN structure for ISO/IEC 15459 UII

An example of a directory distinguished name for such an item could be:

DN = { urnC="iso", urnC="std", urnC="iso-iec", urnC=15459, urnC="25S",urnC="UN", urnC="043325711," urnC="MH80310000001"}

13 EPC/UII format information

13.1 EPC/UII format DIT subtree

In order to access EPC/UII format information, the format information has to be stored in a directory subtree. The format information for EPC is provided using the same subtree structure as for UIIs. This is illustrated in Figure 9.

NOTE – This does not necessarily mean that both sets of entries are in the same directory system.

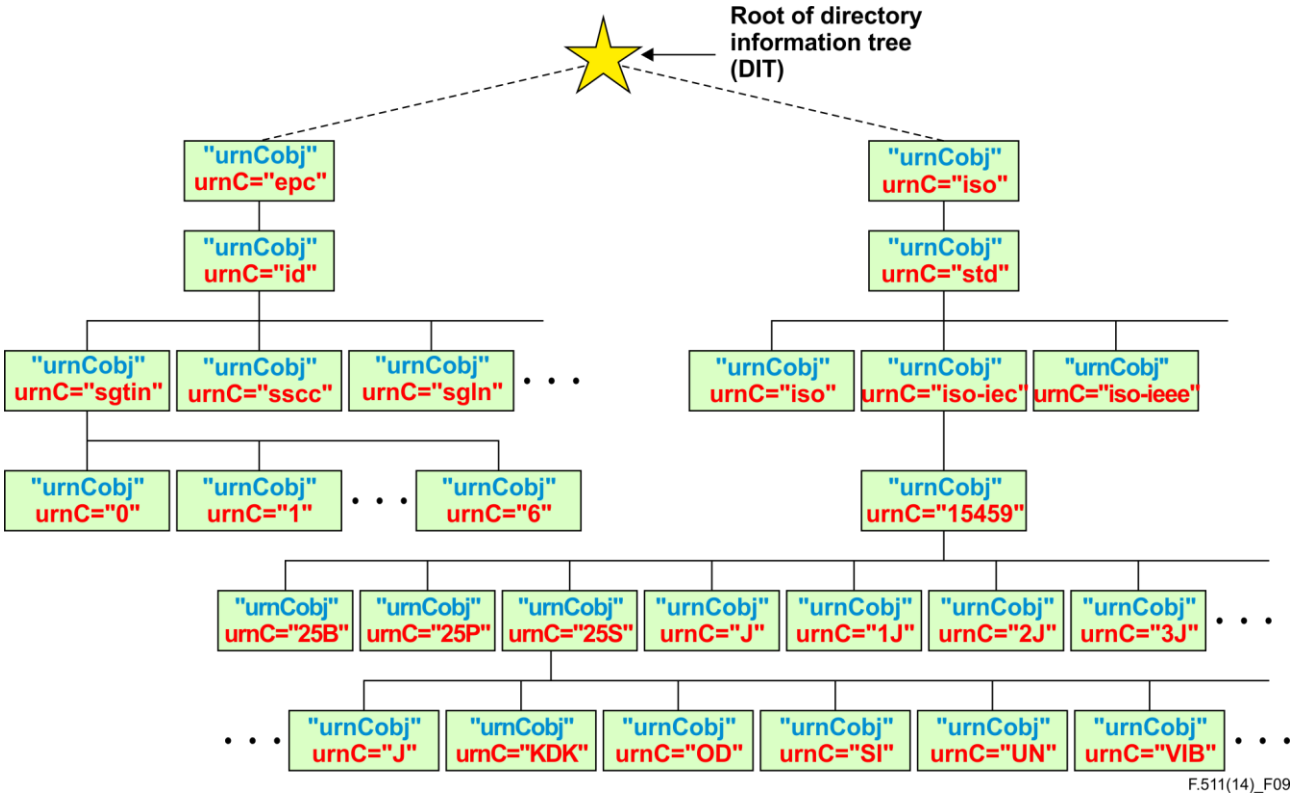

Figure 9 – Possible structure of a DIT subtree for holding format information

13.2 EPC format information

13.2.1 Storing and accessing EPC format information

The EPC format information may then be identified by the following URN:

"urn:spc:id:<header>.<partition>"

where the <header> is the EPC header.

For a given combination of EPC header and partition value the length of the company prefix is fixed. The complete format information may be obtained by a single access and returned in an attribute of type **`epcFormat`**. The **`epcFormat`** attribute type is defined in [ITU-T X.520] and for easy reference is described in Appendix VIII.

There are eleven different EPC types each with seven different partition values. Seventy-seven directory entries will be required for holding all the EPC format information.

13.3 UII format information

While it is possible for an EPC to obtain all the necessary format information using a single access, it becomes more complex for UIIs due to the greater diversity and the many organizations involved. The complete format of an UII is not necessarily determined by a single committee or organization.

It is quite straight forward to isolate the DI and the IAC fields of UII if the encoding is known:

– For AFI values 'A1'H to 'AA'H, as allocated to [ISO/IEC 15459 S], a six-bit encoding is used.

The part of the URN for the UII up to and including the IAC is as follows:

"urn:iso:std:iso-iec:15459:<di>.<iac>"

The length of CIN (or its equivalent) is determined by the IAC. However, it cannot be assumed that the length of CIN is fixed for a particular IAC. As an example, the Universal Postal Union (UPU) specifies the CIN field (called the issuer code) to be either two or three characters long.

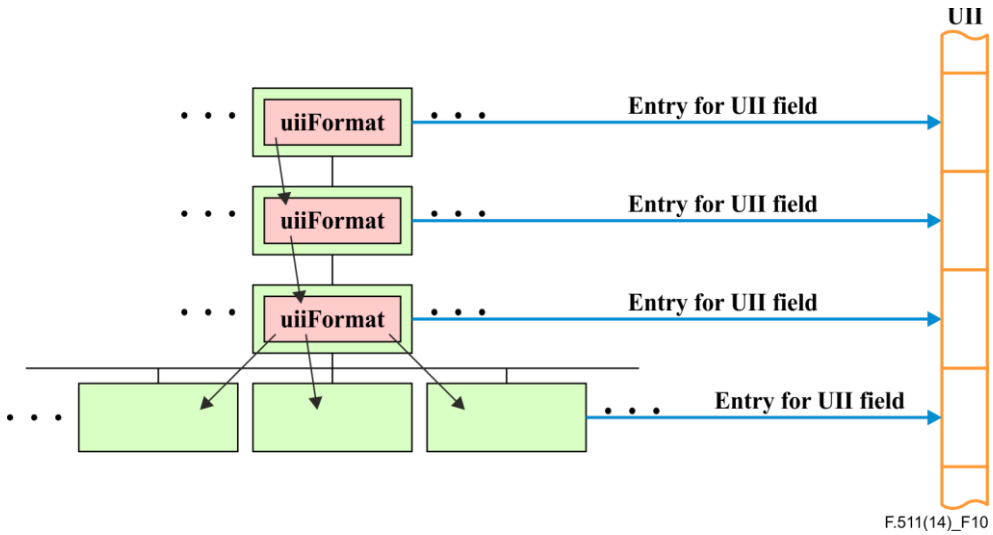

Figure 10 – Retrieving EPC format information

To cope with this situation, a result of a search operation may be an attribute of type **`uiiFormat`** returned to the AIDC client system to give a recommendation on how a subsequent search should be performed. The **`uiiFormat`** attribute type is described in Appendix VIII.

It may not always be possible, or necessary, to get the complete format information. By knowing the DI, IAC and CIN values, the company is identified and it may be possible to obtain a reference, e.g., a URL, to the site of the company, where a complete search may be performed. Alternatively, additional formatting information may be returned in order to navigate to the wanted information.

To ease the following more detailed description, the variable **UIIpointer** is introduced.

1) The first access by the AIDC client system is a search using the distinguished name corresponding to "urn: iso:std:iso-iec:15459:<di>.<iac>" as the **`baseObject`**, a **`scope`** equal to **`baseObject`** and an empty filter (an emulated read). The **UIIpointer** is set to the first character after the IAC in the UII. Two cases are considered depending on the returned result:

   a) If the result is an attribute of type **`uiiFormat`**, a search is performed following the recommendation given in that attribute. The characters to be used in the filter are taken from the UII starting from the **UIIpointer**.

   b) If the issuing agency has its own site, the result in addition to an attribute of type **`uiiFormat`** could include an attribute of type **`ldapURL`** pointing to that site. A search is performed using the recommendations in the attribute of type **`uiiFormat`** against this new site. The characters to be used in the filter are taken from the UII starting from the **UIIpointer**.

2) The result from the latest search may return zero or more entries. Zero entries are returned in the case of an error situation where it not possible to resolve the UII. When entries are returned, the distinguished names are returned together with the entries. Such a distinguished

name minus the last component is the distinguished name of the base object for the performed search. The last component of the distinguished name is the relative distinguished name (RDN) for the entry. The RDN is actually formed by the same attribute that was the target for the search filter (the naming attribute). The RDN can also be used to construct the next component of the corresponding URN for the entry.

If only a single entry is returned and this attribute holds a single value, the next URN component is finally resolved and the **UIIpointer** is updated with the number of characters in that component.

If only a single entry is returned, but the naming attributes holds multiple values, it is checked what values the search filter matched. The longest one of these values determines the number to be added to the **UIIpointer**.

Four cases are considered depending on the returned result from 1) above:

a) If the result is a complete result, the process is complete. This would be the case if the attribute of type **`uiiFormat`** in the previous search specified that the complete UII to be specified in the filter or if the UII has been completely resolved into components (the updated **UIIpointer** points to the end of transmission (EOT)).

b) If the result is an attribute of type **`uiiFormat`**, a search is performed according to the value of that attribute.

c) If the result is both an attribute of type **`ldapURL`** and an attribute of type **`uiiFormat`**, the search filter is constructed as in b), above, and the search is performed against the site identified by the **`ldapURL`**.

3) Step 2) is repeated until a final result is returned.

14 Support of ITU-T Y.2213

[ITU-T Y.2213] specifies different levels of support:

a) Forward identifier resolution, which means resolving the identifier into associated information.

b) Reverse identifier resolution, which means resolving the associated information into the corresponding identifier.

c) One-to-many associations between an identifier and information of different types vs one-to-one association. One-to-many associations allow users to access association information depending on the type of user. Manufacturers, retailers or consumers may access different types of associated information based on the same identifier.

Next-generation networks (NGNs) require protection against a single point of failure.

14.1 General

In the simple case, there is a one-to-one relationship between a tag and its associated information, but there are more complicated cases as outlined in [ITU-T Y.2213].

Different users may want to access different information for the same tag information. For example, manufacturers may use identifiers for production planning while retailers may use the same identifiers for store inventory management, and consumers may use the same identifiers for product information retrieval.

Some users may be interested in information about a particular item; others may be interested type information, while others again may want information about the organization that produced the item.

[ITU-T Y.2213] identifies the requirement to retrieve tag information based on the information associated with the tag (reverse identifier resolution).

[ITU-T Y.2213] also identifies a requirement for having location information as part of the information associated with a tag.

14.2 Multiple user groups for same EPC or UII

Different user groups may require different associated information even for the same EPC or UII. There are different ways this may be achieved:

– Use of access control: Different contents may be held by different attributes. By use of access control, a particular user group will only get access to information relevant for that user group. Likewise, if the returned information is a URL to the content, different URLs may be held by the same attribute. By use of access control down to value level, only the relevant URL will be returned.

  NOTE – LDAP does not have access control specifications, although many LDAP implementations have proprietary access control.

– It is possible in a directory request to specify what attributes are to be returned. Different user groups may specify different attributes to be returned.

– Different user groups may access different directory systems. This will require that some information be duplicated among these systems.

14.3 Reverse identifier resolution

Clause 7.2 of [ITU-T Y.2213] specifies a requirement for reverse resolution i.e., finding the identifier of an object and its location from the associated information. Reverse identification only makes sense if the associated information is only relevant for a single identifier.

If the directory provides the associated information, the entry holding that information may also hold attributes describing the tag information

14.4 Location-based information

Clause 7.6 of [ITU-T Y.2213] specifies the requirements for location-based service support.

The location of a tag may be provided in an attribute of type **`tagLocation`**.

14.5 Avoidance of single-point-of-failure

Avoiding single-point-of-failure has many aspects outside the scope of these directory specifications, such as:

– duplication of networks following different physical routes

– use of redundant arrays of inexpensive disks (RAID) technology

– back-up of databases.

All these techniques are well established, known and deployed by responsible organizations as needed.

Replication is another way to prevent single point of failure. [ITU-T X.525] defines protocols and procedures for replication of information. LDAP does not include a specification for replication, but many LDAP implementation have proprietary replication solutions.

Appendix I

Use of OIDs in an RFID environment

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

I.1 Introduction

[ISO/IEC 15459 S] and [ISO/IEC 15961-1] introduces the possibility to add OIDs to RFID tags and thereby provide an additional technique to create unique identities. Although this possibility exists there is no requirement that OIDs are actually encoded on the RFID tags.

This appendix gives an overview of the RFID OID structure.

I.2 OID used in the ISO environment

As indicated, AFIs are allocated from a flat name space requiring a single authority for the allocation of values for different purposes. OIDs were invented to circumvent this problem. Here the allocation of values may be delegated. Delegation may go down to specific standards activities. An OID together with an ISO UII provides a global unique identification.

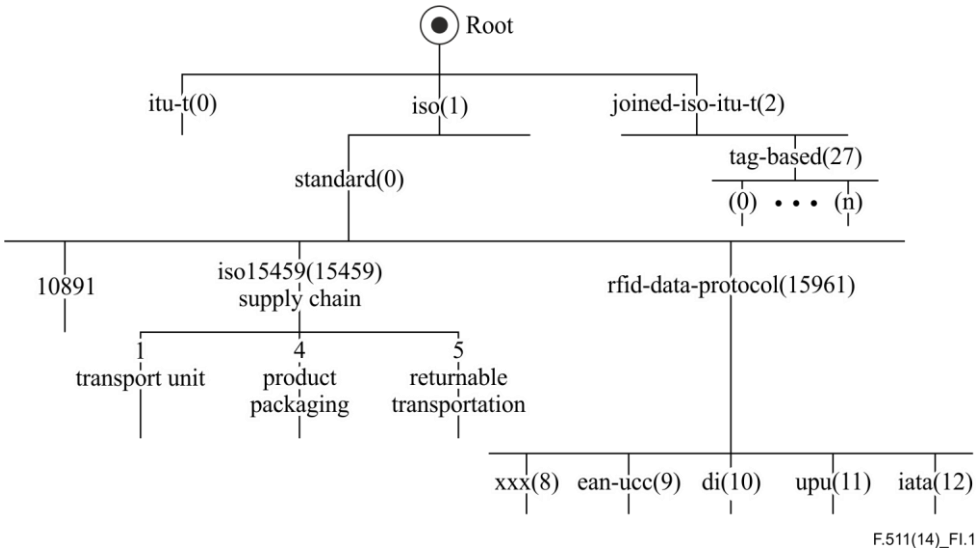

Figure I.1 – ISO/RFID object identifier structure

Figure I.1 shows examples of OID assignments. OIDs, or at least the upper level arcs, are allocated by International Standards and/or ITU-T Recommendations.

[b-ISO/TS 10891] defines OID RFIDs for freight containers.

[ISO/IEC 15459 S] define the following OIDs:

– {1 0 15459 1} is an OID for transport units.

– {1 0 15459 4} is an OID for product tagging.

– {1 0 15459 5} is an OID for returnable transportation.

[ISO/IEC 15961] defines the following OIDs:

– {1 0 15961 8} is an OID for the data format used for libraries according to ISO/IEC 28560-2 "Information and documentation – RFID in libraries – Part 2: Encoding based on ISO/IEC 15962".

– {1 0 15961 9} is an OID for data format used when all the data on the RFID tag complies with GS1 as referred to in ISO/IEC 15418 "Information technology – Automatic identification and data capture techniques – GS1 Application Identifiers and ASC MH 10 Data Identifiers and maintenance".

– {1 0 15961 10} is an OID for data format used when all the data on the RFID tag complies with the Data Identifier standard (as referred to in [b-ISO/IEC 15418]).

– {1 0 15961 11} is an OID for the Universal Postal Union.

– {1 0 15961 12} is an OID for International Air Transport Association (IATA). IATA may further define sub-arcs for specific purposes. As an example, sub-arc 1 is used for IATA baggage identification code.

[b-ITU-T X.668] specifies an OID structure for possible use by tag-based applications. It defines a structure requiring only three arcs to save storage on RFID tags. New RFID applications may choose to apply for an arc within this structure. Current applications may choose to convert to this OID structure.

I.3 ISO/IEC 15459 OID structure vs. data identifiers/application identifiers

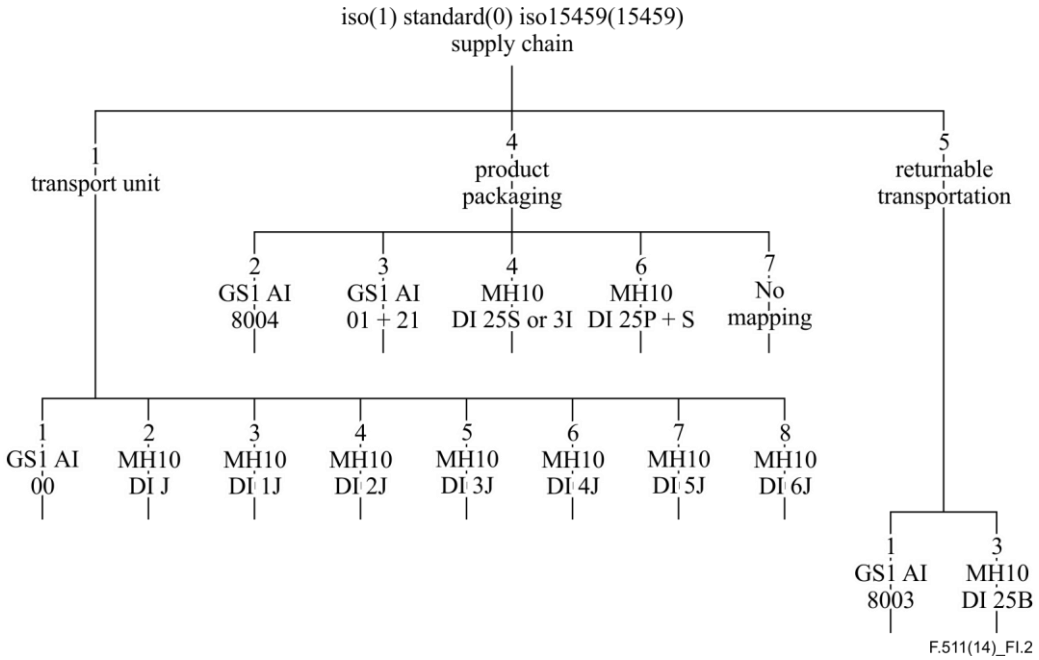

Figure I.2 – ISO/IEC 15459 OID structure with DI and AI indications

Figure I.2 shows how the OID branch for [ISO/IEC 15459] parts is further branching out for different purposes.

For [ISO/IEC 15459 S], an OID is defined for each data identifier (DI) and for each GS1 application identifier (AI).

Appendix II

Introduction to ITU-T X.500-series/LDAP concepts

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

II.1 Directory entries and their content

A directory system holds the information in so-called entries. An entry holds information about a particular physical or logical object. Within this Recommendation an object may be the item to which an AIDC media is affixed or it may be a single field within an EPC or UII.

The information in entries is held by so-called directory attributes (or just attributes). An attribute is of a defined type, where the type is indicated by an OID and the value syntax is part of the attribute type specification.

In addition to an OID, an attribute type has a textual identifier that is a representation of the OID. This textual identifier is used in this Recommendation when referring to a particular attribute type.

Directory systems are open ended systems that can be adapted to handle diverse information types. Neither [ITU-T X.500-series] nor [IETF RFC 4511] (LDAP) need any extension to support new types of information, except for the definition of some new attribute types for holding these types of information. No extensions to the basic protocols are needed.

A directory is built to store and provide many types of information, such postal address, other location information, shipping information, etc. Both [ITU-T X.500-series] and [IETF RFC 4511] (LDAP) define general useable attribute types to be used by any application. Attribute information is returned as required.

The required number of interactions with the directory infrastructure is quite limited and in some cases a single interaction is sufficient.

Directory systems have well-proven security functions allowing, for example, strong authentication of accessing clients, protection of information, etc.

II.2 Directory information tree (DIT)

The directory information within a directory system is conceptually stored in a DIT where the individual vertices are directory entries. The DIT reflects the hierarchical nature of things. As an example, a country is divided into localities and large organizations. Locations may again be subdivided into smaller localities down to smaller companies and private persons. A large organization may be divided into organizational units possibly in several levels down to the individuals.

EPCs and UIIs have also hierarchical structures and the EPC/UII fields may therefore be represented by directory entries.

II.3 Directory subtrees

A directory subtree is a set of entries which themselves constitute a tree structure, but where the subtree root typically is different from the root of the DIT and the leaves of the subtree may not coincide with leaves of the DIT.

II.4 Distinguished names and URNs

[ITU-T X.500-series] and [IETF RFC 4511] (LDAP) both have the concept of distinguished name. A distinguished name reflects the hierarchical nature of objects. Entries in the DIT reflect the hierarchical structure of the objects they represent. A distinguished name is comprised of a number of components. Each level of the DIT adds a new component to a distinguished name. This component for an entry is called the relative distinguished name (RDN) for the entry. The distinguished name for an entry is therefore the sequence of RDNs from the DIT root down to and including the entry in question.

An RDN is an {attribute type, attribute value} pair, i.e., that one of the values of an attribute is used for the RDN for an entry.

NOTE – An RDN may actually consist of multiple {attribute type, attribute value} pairs; however, this is not relevant for this Recommendation.

URNs also have a hierarchical structure, which means that it is rather straight forward to map a URN onto a corresponding distinguished name. This is detailed in Annex G of [ITU-T X.520].

II.5 Directory search operation

II.5.1 Search filter

Directory systems have extensive search capabilities allowing several search criteria to be expressed in a single filter used for interrogating the directory database. Each entry within the scope (see clause II.5.3) is checked against the filter. This filter is composed of filter items, where each filter item is an assertion about, and an attribute of, a specific type. If that assertion is proved satisfied, the filter item yields TRUE. Filter item may combined to an arbitrary complex filter using a recursive definition and by using the logical operators **`and`**, **`or`** and **`not`**. An entry is returned to the requester if the combined filter yields TRUE using the normal rules for Boolean algebra.

Filter items take many forms; however, in this Recommendation it is only necessary to consider attribute value assertions. An attribute assertion is a slightly modified form of an attribute. Only equality assertions are relevant.

II.5.2 Search base object

To avoid searching the whole DIT for every search operation, it is required to specify the distinguished name of an entry from where the search operation is to start. The distinguished name may be empty signalling that the search operation shall start at the DIT root.

II.5.3 Search scope

The search scope specifies the range of the search operation. It takes one of the following values:

– **`baseObject`**, meaning that only the base object is to be search for match. By specifying an empty filter, this corresponds to a read operation. This is how LDAP emulates a read operation.

– **`oneLevel`**, meaning that only the entries immediately subordinate to the base object are to be searched.

– **`wholeSubtree`**, meaning that the base object and all its subordinate entries are to be searched.

II.6 Entry information selection

By default, when a directory entry has been selected for return to the AIDC client system, all information available in that entry will be returned (subject to access control restrictions). However, when issuing an interrogation request, it is possible to specify what attribute types are to be returned. Therefore it is possible to limit the returned result to what the AIDC client system intends to process.

II.7 Pointers to other directories (referrals)

Both [ITU-T X.500-series] and [IETF RFC 4511] (LDAP) have the concept of referrals, which is a way for a directory server to signal that other directory servers are able to further progress a request. This will also include information about how to access that directory server.

An LDAP referral is a URL where the scheme (the first part of a URL) is "ldap".

Appendix III

Introduction to RFID tag structure

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

III.1 Scope of appendix

The purpose of this appendix is to give a short tutorial on the structure of RFID tags and to isolate those information types that are of interest from the point of view of directory support.

III.2 ISO/IEC 18000-6C and ISO/IEC 18000-3m3 RFID tag structure

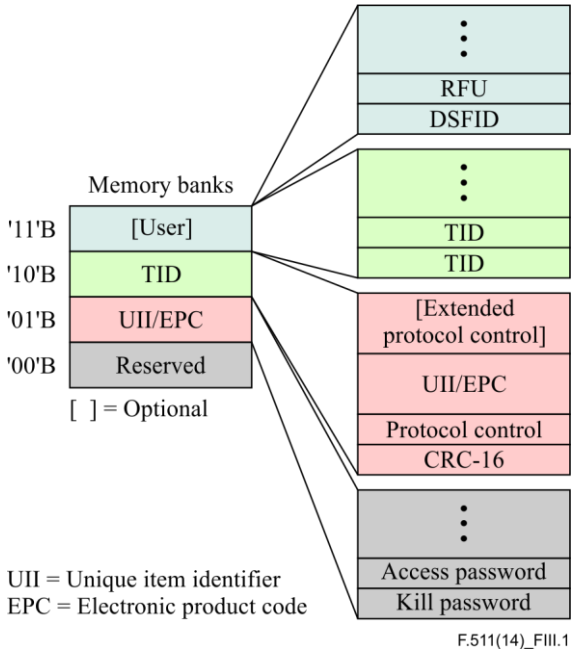

Figure III.1 – ISO/IEC 18000-6C and ISO/IEC 18000-3m3, memory structure

The following RFID tag types are considered:

– The ISO/IEC 18000-6C tag type as defined in [ISO/IEC 18000-6]. This tag type is identical to the GS1 Class 1 Gen 2 tag type.

– The ISO/IEC 18000-3 Mode 3 tag type as defined in [ISO/IEC 18000-3].

The two tag types have the same memory structure, but uses different frequency ranges on the air interface. The tag structure is shown in Figure III.1.

The tag memory is logically separated into four distinct banks as shown in Figure III.1 and is discussed in the following clauses.

III.3 Bank '00'B (reserved memory)

The reserved memory contains the kill and/or access passwords, if passwords are implemented on the tag. The kill password protects against setting the tag out of function or reusing the tag for some new purpose (recommission). The access password controls read and write access to the tag.

III.4 Bank '01'B (UII/EPC memory)

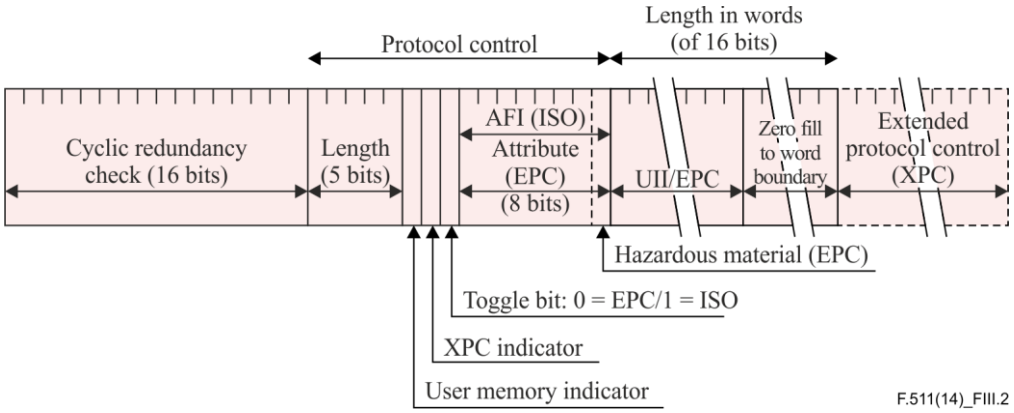

Figure III.2 – Memory bank '01'B (UII/EPC memory)

The UII memory bank is illustrated in Figure III.2 and it has the following fields:

a) The cyclic redundancy check field, which protects the protocol control and the UII/EPC fields.

b) The protocol control field with the following content:

- The length field holds the length of the UII/EPC field in 16 bits words.
- The user memory indicator indicates whether there is any information in the user memory bank. If the bit is set, the user memory bank is present and formatted with information. Otherwise, the indicator is not set.
- The extended protocol control (XPC) indicator is set if the extended protocol control field is present and not all-zero.
- Toggle bit with a value zero indicates that the tag is formatted according to the GS1 specifications. A value one indicates that the tag is formatted according to ISO/IEC or ISO specifications.
- If the toggle bit is one, the AFI/attribute subfield holds an AFI (see clause 10.2). If the toggle bit is zero, the AFI/attribute subfield holds an 8 bit attribute (do not confuse with directory attributes). Only the least significant bit is defined. If that bit is set to '1'B, it indicates that the tag is affixed hazardous material.

c) The UII/EPC field holds an electronic product code (EPC) if the toggle bit is set to '0'B and a unique item identifier (UII) if the toggle bit is set to '1'B.

d) The Extended Protocol Control field is present if the tag supports recommissioning or supports sensors and batteries.

III.5 Bank '10'B (TID memory)

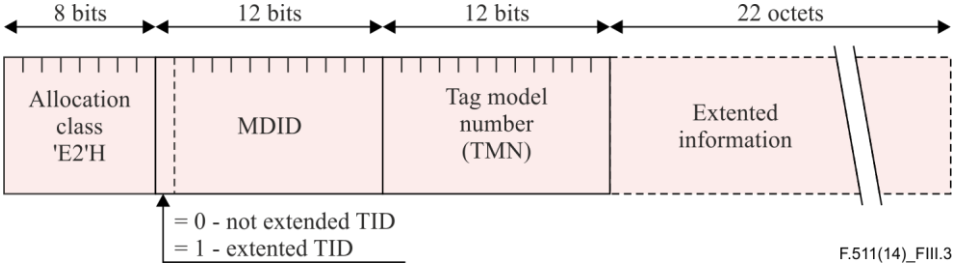

Figure III.3 – Memory bank '10'B (TID memory)

The unique tag identifier (TID) occupies Bank '10' as illustrated in Figure III.3.

The TID provides tag manufacturer information, i.e., it provides information about the tag itself, as opposed to the item to which the tag is affixed.

There are different formats for the TID, e.g., for different types of tags. The TID format to be used for tags according to ISO/IEC 18000-6C and ISO/IEC 18000-3m3 is shown in Figure III.3. The structure of this tag is defined in [b-ISO/IEC 15963].

The first 8 bits of the TID is the allocation class indicating the type of TID. An allocation class for an ISO/IEC 18000-6C and ISO/IEC 18000-3m3 tag is 'E2'H ('1110 0010'B).

The mask-designer identification code (MDID) field holds a code identifying the manufacture of the tag, while the tag model number (TMN) indicates a tag from that manufacture with particular characteristics.

An extended TID holds additional information within the extension (see [b-ISO/IEC 15963]) including a serial number. [b-GS1TAGDATA] specifies that for a particular MDID/TMN combination, all tags with that combination shall have identical values in the extension, except for the serial number.

A TID may be held in a directory by an attribute of type **`tagTID,`** as defined in [ITU-T X.520].

When loading information about a tag into a directory, the TID could also be included. When reading a tag and finding that the TID on the tag is different from the one in the directory, it can be concluded that the item to which the tag is affixed is a counterfeit item.

III.6 Bank '11'B (user memory)

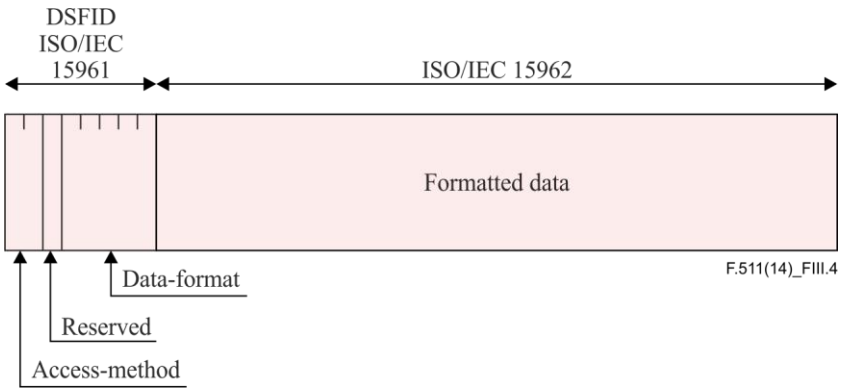

Figure III.4 – Memory bank '11'B (user memory)

User memory is optional.

The first octet is the data storage format identifier (DSFID), while the remaining part of the memory bank holds formatted data.

The first two bits of DSFID indicate the access-method, i.e., how the formatted information is accessed:

0: No-Directory

1: Directory

2: Packed-Objects

3: Tag-Data-Profile

The next bit is reserved.

The remaining five bits are used for specifying the Data-format:

0: Not formatted

1: Data elements identified by full OIDs

2: Root OID encoded on tag

3-28: Open environment – Root OID always {1 0 15961 <data format>}

29: Closed environment – 15961-1/15962 encoded – Root OID always {1 0 15961 0}

30: Closed environment – not 15961-1/15962 encoded

31 Indicates additional octet.

NOTE – There is a mechanism to extend the DSFID to more than one octet, which will allow additional values for the Access-Method and the Data-Format.

The encoding of user memory locations after the DSFID shall be as defined in [b-ISO/IEC 15962].

Appendix IV

Overview of EPC types

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

IV.1 Scope of appendix

IV.2 Structure of serialized global trade item number (SGTIN) EPC types

The SGTIN EPC types are considered in detail to illustrate how to convert a bit-encoded EPC to a URN or to a character encoded representation (GS1 element string) by using knowledge of the format of the bit-encoded EPC.

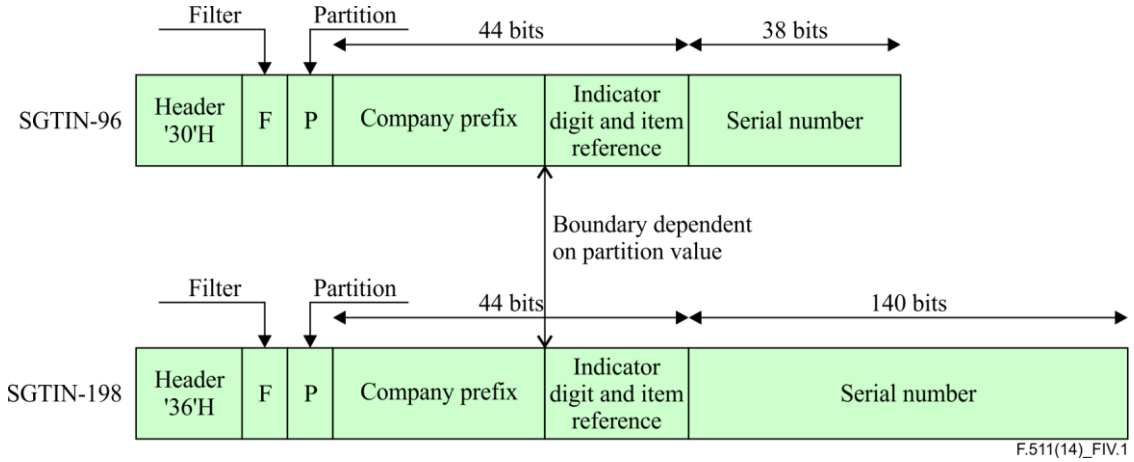

Figure IV.1 – Format of SGTIN EPC types

Figure IV.1 shows the format of an SGTIN EPC. There are two different types of SGTIN, one type with a bit length of 96 (SGTIN-96) and another one with a bit length of 128 (SGTIN-128). An EPC of one of these types identifies a physical object for sale.

The header values in Figure IV.1 indicate that the EPC types are SGTIN-96 and SGTIN-198.

Three fields identify the actual item, namely the company prefix, the indicator digit/item reference (type of item) and serial number. The combined length of company prefix and indicator digit/item reference fields is always 44 bits for an SGTIN, but the boundary between the two fields is dependent on the partition value. These three fields are considered unsigned binary fields that shall be converted to numeric characters in the character-encoded format.

The conversion between the bit-encoding and the character-encoding of an SGTIN EPC is illustrated in Figure IV.2.

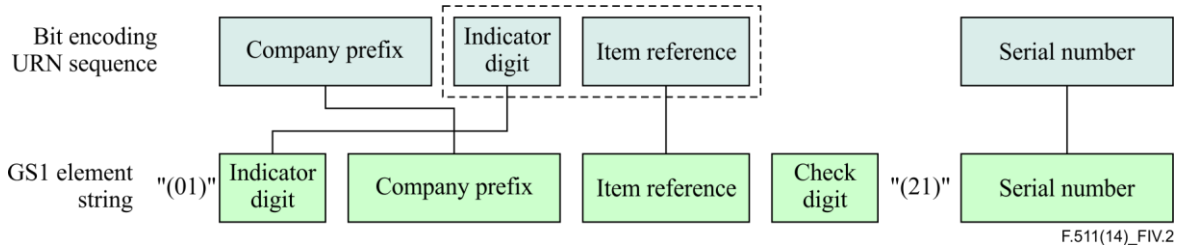

Figure IV.2 – Relationship between bit-encoded and character encoded SGTIN EPC

The partition field indirectly indicates the length of the company prefix, e.g., a zero partition value corresponds to a company prefix length of 40 bits, a partition value of one corresponds to a length of 37 bits, and so on. [b-GS1TAGDATA] provides tables giving the relationship between partition values and length of the company prefix. Varied EPC types have generally different tables, although EPC types of the same "family" like SGTIN may share a single table. Table V.1 gives an example of this.

[b-GS1TAGDATA] also specifies how a binary field is converted to character string. As an example, the bits making up a company prefix are considered an unsigned integer to be converted into a corresponding numeric character representation. In other cases, the serial number for SGTIN-128, for example, is a series of seven bit bytes, each representing a character according to [b-ISO/IEC 646].

When creating a GS1 element string, the characters have to be shifted around, some additional characters and a check digit are added. Figure IV.2 illustrates that the first digit of the indicator digit/item reference field after conversion from bit-encoding to character-encoding shall be moved to the front of the character string.

In front of some fields, additional information is added indicating the types of fields when converted into GS1 element string.

NOTE – 01 is a so-called application identifier (AI). This particular AI signals that what follows is a global trade item number (GTIN), while the AI 21 signals that what follows is a serial number. The AIs are surrounded by parenthesis.

When creating a URN, the indicator digit shall not be moved and the check digit shall not be created, i.e., the bit encoding and the URN have the same components in the same order.

Other EPC UII types have different formats, but it is common for all of the types that the formatting information may be conveyed in an attribute of type **`epcFormat`** as defined in [ITU-T X.520]. A description of this attribute type is given in Appendix V.

IV.3 Structure of serial shipping container code (SSCC) EPC types

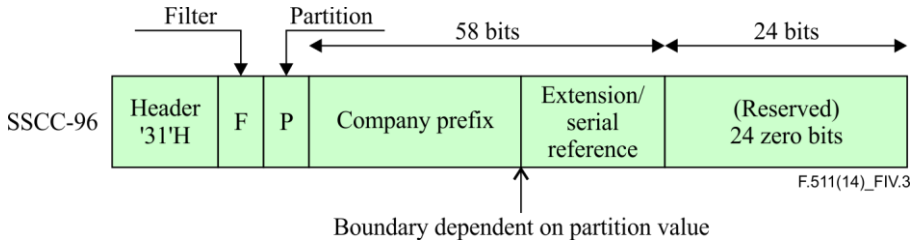

Figure IV.3 – Format of SSCC EPC types

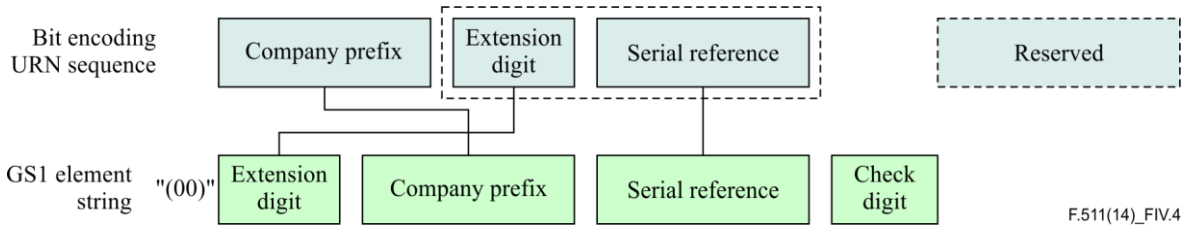

Figure IV.4 – Relationship between bit-encoded and character encoded SSCC EPC

The serial shipping container code (SSCC) EPC type identifies a shipping container. One EPC type is defined as SSCC-96. Figures IV.3 and IV.4 illustrate how conversion is performed for a SSCC EPC type. It follows the same principles as described in clause 13.1.1.

IV.4 Structure of serialized global location number (SGLN) EPC types

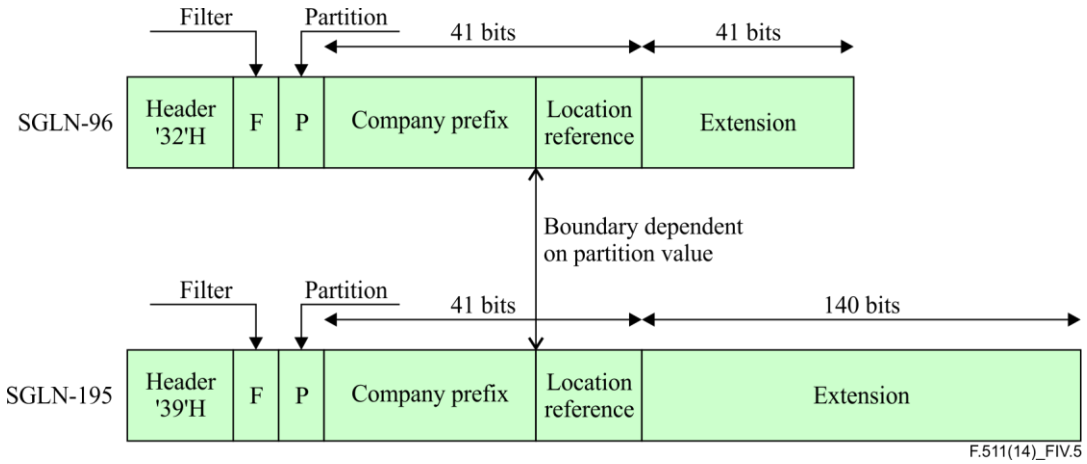

Figure IV.5 – Format of SGLN EPC type

Two EPC types are defined for SGLN as SGLN-96 and SGLN-195. There is no digit shifting. The GS1 element string is prefixed with the AI value "(414)". The extension field may be omitted in the GS1 element string, in which case the check digit is added to the end of the string. If the extension field is included, it is prefixed with the AI value "(254)". The check digit is inserted between the location reference field and the extension field.

IV.5 Structure of global returnable asset identifier (GRAI) EPC types

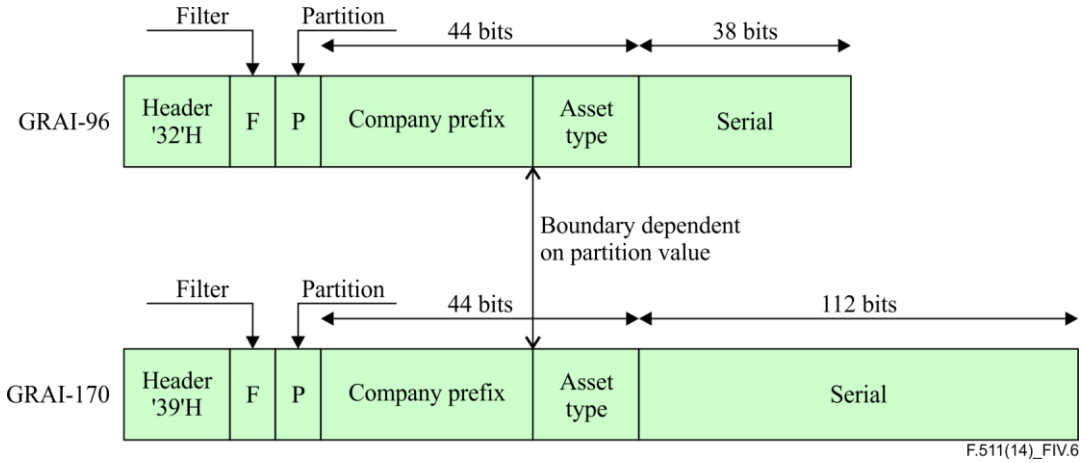

Figure IV.6 – Format of GRAI EPC types

Two EPC types are defined for GRAI as GRAI-96 and GRAI-170. There is no digit shifting. The GS1 element string is prefixed with the AI value "(8003)" followed by an additional "0" numeric character. The check digit is inserted between the asset type field and the serial field.

IV.6 Structure of global individual asset identifier (GIAI) EPC types

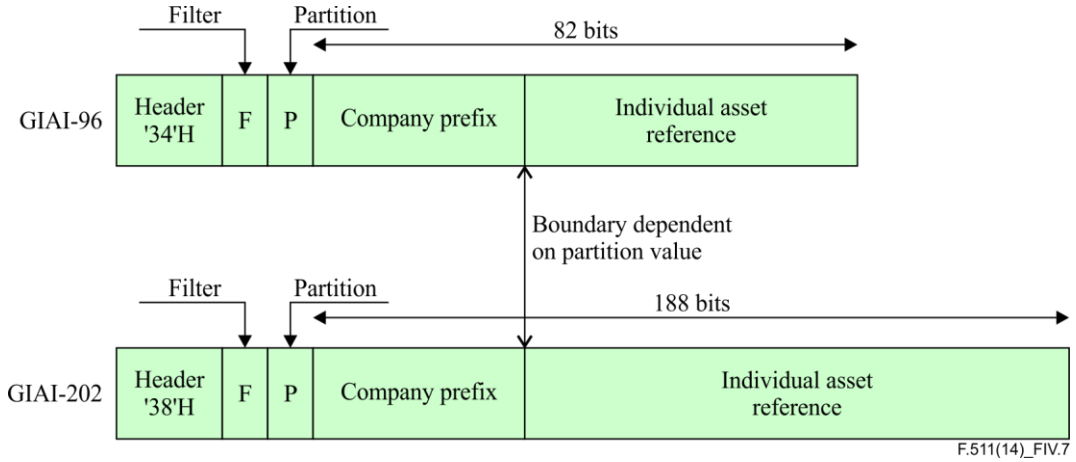

Figure IV.7 – Format of GIAI EPC types

Two EPC types are defined for global individual asset identifier (GIAI) as GIAI-96 and GIAI-202. There is no digit shifting. The GS1 element string is prefixed with the AI value "(8004)".

IV.7 Structure of global service relation number (GSRN) EPC types

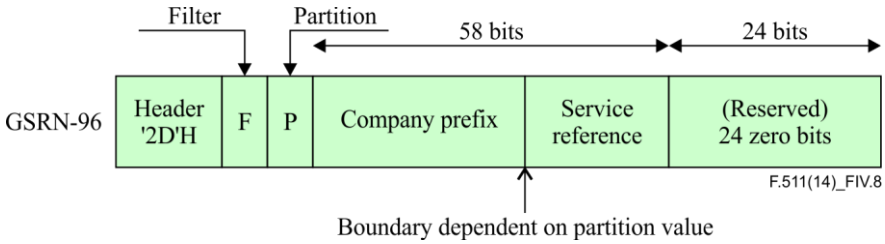

Figure IV.8 – Format of GSRN EPC types

One EPC type is defined for global service relation number (GSRN) as GSRN-96. There is no digit shifting. The GS1 element string is prefixed with the AI value "(8018)".

IV.8 Structure of global document type identifier (GDTI) EPC types

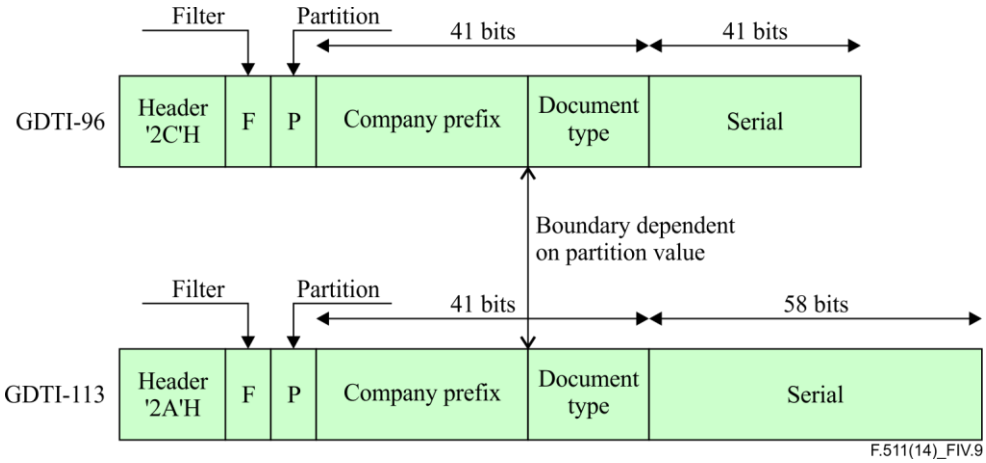

Figure IV.9 – Format of GDTI EPC types

Two EPC types are defined for global document type identifier (GDTI) as GDTI-96 and GDTI-113. There is no digit shifting. The GS1 element string is prefixed with the AI value "(8004)". A check digit is added between the document type field and the serial field.

Appendix V

Example of retrieving EPC format information

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

An example is used to illustrate how EPC format information may be retrieved. The EPC type SGTIN-198 with header value '36'H and partition value three is used in this example. The URN for the format information for that EPC type may be expressed as:

"urn:spc:id:sgtin.3"

An entry addressed by this URN will have an attribute of type **epcFormat** with the following value expressed in extensible markup language (XML) notation.

```
<fields>
  <field>
    <bits>30</bits>
    <charField>
      <characters>9</characters>
    </charField>
    <result><numericPad/></result>
    <gs1Key>(01)</gs1Key>
  </field>
  <field>
    <bits>14</bits>
    <charField>
      <characters>4</characters>
    </charField>
      <result><numericPad/></result>
  </field>
  <field>
    <bits>140</bits>
    <charField>
      <characters>20</characters>
    </charField>
    <result><alpha7bits></result>
    <gs1Key>(21)</gs1Key>
  </field>
</fields>
<digitShift>10</digitShift>
<checkCalc>13</checkCalc>
<urnPrefix>urn:epc:id:sgtin:</urnPrefix>
```

This value states that the EPC consists of three fields beyond the header, filter and the partition fields. According to [b-GS1TAGDATA] (see Table V.1) a partition field value of three means:

– The first field is 30 bits and once converted represents nine numeric characters padded in the front with zeroes if necessary. The GS1 Key "(01)" has to be added in front if a GS1 element string is produced.

– The second field is 14 bits long and once converted represents four numeric characters padded in the front with zeroes, if necessary.

– The third field is always 140 bits long and consists of seven bits bytes each representing a seven bits character as defined by [b-ISO/IEC 646].

As the indicator digit is the first digit of the second field, it has number 10. In case a GS1 element string is produced, this digit has to be moved to the front of field one.

The check digit is placed after the first two fields and therefore gets position number 14 and is calculated over all the characters in front of it (excluding the GS1 key).

In case a GS1 element string is produced, the GS1 key "(21)" has to be added in front of the third field.

In case a pure identity EPC URI is produced, it shall be prefixed with "urn:epc:id:sgtin:".

The following is an example of EPC in binary encoding to be converted to a GS1 element string and to a pure identity EPC URI:

'00110110000011101111010011010011111101100011001101011001011010100110100011010011110011100100111100111000001101001111001011110010110100111000011101100100111011101011100010110010111100100000000'B

The first eight bits are the header. They have the value '0011 0110'B, which indicates that the EPC is an SGTIN-198.

The next three bits field is the filter field and has the value 000 followed by the partition field with the value '011'B (decimal 3).

As the partition field specifies the partitioning of the two first fields, it is necessary to consult the partition table for SGTIN as shown in Table V.1.

Table V.1 – Partition table for SGTIN

Partition value	Company prefix		Indicator digit and item reference	
	Bits (M)	Digits (L)	Bits (N)	Digits
0	40	12	4	1
1	37	11	7	2
2	34	10	10	3
3	30	9	14	4
4	27	8	17	5
5	24	7	20	6
6	20	6	24	7

The serial number for SGTIN-96 has a fixed length of 38 bits. When converted to numeric characters, the maximum value is 274.877.906.943. If the result exceeds that value, the UII is invalid.

The serial number for SGTIN-198 has a fixed length of 140 bits allowing for 20 seven-bit ASCII characters.

Having removed the header, the filter and the partition bits, the remaining bit stream has to be converted based on the format information from the **`epcFormat`** attribute value shown in Appendix VIII.

'10111101001101001111111011000110011010110010110101001101000110100111100111001001111100111000001101001111001011110010110100111000011101100100111011101011101101110001011001011110010000000'B

a) The first field is 30 bits long ('101111010011010011111101100011'B). When treated as an unsigned integer, this bit string is converted into the numeric characters: 793591651. As this field now consists of nine numeric characters, it does not have to be padded in the front with zeroes.

b) The second field is 14 bits long ('00110101100101'B). When treated as an unsigned number, this bit string is converted into numeric characters: 2917. As this field now consists of four numeric characters, it does not have to be padded in the front with zeroes.

c) The third field is 140 bits long or 20 bytes of seven bits ('1010100'B '1101000'B' 1101001'B '1110011'B '1001001'B '1110011'B '1000001'B '1010011'B '1100101'B '1110010'B '1101001'B '1100001'B '1101100'B '1001110'B '1110101'B '1101101'B '1100010'B '1100101'B '1110010'B '0000000'B). The first nineteen blocks are converted into alphanumeric characters (yielding "ThisIsASerialNumber"). The last block is all zero bits and therefore ignored.

When concatenated, the converted fields consist of 32 characters numbered from 1 to 32:

7935916512917ThisIsASerialNumber.

The 10th character has to be move to the front:

2793591651917ThisIsASerialNumber.

The 13 first characters are now used to yield the check digit:

(3 × 2 + 7 + 3 × 9 + 3 + 3 × 5 + 9 + 3 × 1 + 6 + 3 × 5 + 1 + 3 × 9 + 1 + 3 × 7) = 141 subtracted from 150 equals nine, which is the check digit (see [b-GS1TAGDATA]).

The resulting GS1 element string is "(01) 27935916519179 (21) ThisIsASerialNumber"

If the pure identity URI is to be generated, the **`urnPrefix`** is concatenated with the fields as generated within a) b) and c) above with full stops inserted, which results in:

"urn:epc:id:sgtin:793591651.2917.ThisIsASerialNumber"

Appendix VI

Overview of ISO/IEC 15459 UII types

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

VI.1 Overview

This appendix provides a short description of the different fields of ISO/IEC 15459 UII types for the purpose this Recommendation.

All the fields are in 6 bits character encoding using a limited character repertoire from [b-ISO/IEC 646] consisting of 0-9 and A-Z.

VI.2 Data identifier field

A data identifier (DI) is a string of characters that define the general category or intended use of the data that follows. The string consists of either a single character alone or a character prefixed by one to three numeric characters.

The American National Standards Institute (ANSI) registers DIs as part of the [b-MH10.8.2] standardization and they are listed in [b-ISO/IEC 15418].

The following is a list of DIs relevant for [ISO/EC 15459 S].

I: Exclusive assignment – Vehicle identification number (VIN) as defined in the U.S. under 49 code of federal regulations (CFR), §§ 565 and internationally by [b-ISO 3779]. (These are completely compatible data structures)

J: Unique licence plate number. Maximum field is 35 alphanumeric characters.

1J: Unique licence plate number assigned to a transport unit which is the lowest level of packaging, the unbreakable unit. Maximum field is 35 alphanumeric characters.

2J: Unique licence plate number assigned to a transport unit which contains multiple packages. Maximum field is 35 alphanumeric characters.

3J: Unique licence plate number assigned to a transport unit which is the lowest level of packaging, the unbreakable unit and which has electronic data interchange (EDI) data associated with the unit. Maximum field is 35 alphanumeric characters.

4J: Unique licence plate number assigned to a transport unit which contains multiple packages and which is associated with EDI data. Maximum field is 35 alphanumeric characters.

5J: Unique licence plate number assigned to a mixed transport unit containing different items on a single customer transaction and may or may not have associated EDI data. Maximum field is 20 alphanumeric characters.

6J: Unique licence plate number assigned to a master transport unit containing like items on a single customer transaction and may or may not have associated EDI data. Max field 20 alphanumeric characters.

1P Item identification code assigned by the supplier

25B: Identification of a party to a transaction in which the data format consists of two concatenated segments. The first segment is the unique code assigned to an issuing agency by the Netherlands Normalisatie-instituut (NEN) in accordance with [ISO/IEC 15459]; the second segment is a unique entity identification assigned in accordance with rules established by the issuing agency. This identification is followed by a supplier assigned serial number to a returnable transport item (RTI). The Maximum field is 35 alphanumeric characters.

25K: Global unique identification of groupings of transport units assigned by the carrier, defined as:

Identification of a party to a transaction as identified in 18V, followed by the bill of lading or waybill or shipment identification code assigned by that party.

26K: Global unique identification of groupings of transport units assigned by the shipper, defined as:

Identification of a party to a transaction as identified in 18V, followed by the Bill of Lading or Waybill or Shipment Identification Code assigned by that party.

25P: Identification of a party to a transaction in which the data format consists of two concatenated segments. The first segment is the unique code assigned to an issuing agency by NEN in accordance with [ISO/IEC 15459-S], the second segment is a unique entity identification assigned in accordance with rules established by the issuing agency. This identification is followed by a supplier assigned a part number. The maximum field is 35 alphanumeric characters.

S: A serial number or code assigned by the supplier to an entity for its lifetime, (e.g., a computer serial number, a traceability number, a contract tool identification).

25S: Identification of a party to a transaction in which the data format consists of two concatenated segments. The first segment is the unique code assigned to an issuing agency by NEN in accordance with [ISO/IEC 15459 S], the second segment is a unique entity identification assigned in accordance with rules established by the issuing agency. This identification is followed by a supplier assigned serial number. The max field 35 is alphanumeric characters.

25T: Identification of a party to a transaction as identified in 18V, followed by the supplier assigned a traceability number.

17V: U.S. DoD CAGE Code

18V: Identification of a party to a transaction in which the data format consists of two concatenated segments. The first segment is the unique code assigned to an issuing agency by NEN in accordance with [ISO/IEC 15459 S], the second segment is a unique entity identification assigned in accordance with rules established by the issuing agency.

VI.3 Issuing agency codes (IAC)

Different agencies, called issuing agencies, may define codes for organizations, companies, institutions, etc. Each issuing agency is allocated an issuing IAC. Such IACs are allocated by a registration authority, which currently is NEN (Nederlands Normalisatie-instituut).

The rules for registering IACs are specified in [ISO/IEC 15459-2].

The following is a short description of the rules for defining IACs.

A single character A to J may be assigned to non-profit international organizations. Examples are:

– D – NATO AC/135

– J – Universal Postal Union

A single letter K followed by a two-character country code for national public administrations. Examples are:

– KKR – Korea Institute of Distribution and Logistics

A two-letter code starting with L to U for other agencies. Examples are:

– OD – ODETTE EUROPE (auto industry)

A three-letter code starting with V to Z for other agencies. Examples are:

– VIB – IBM

VI.4 Company identification number (CIN)

The issuing agency allocates CINs to organizations. The issuing agency determines the length of the CINs within its authority. CIN is the common term for companies or organizations responsible for the remaining fields (by NATO, this is called CAGE or NCAGE Code; by the Universal Postal Union (UPU) called the issuer code; etc.).

VI.5 Remaining fields

The remaining fields are allocated by the organization to which a CIN is allocated. There is no general rule for how the field lengths relate to each other.

Appendix VII

Examples of retrieving UII format information

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

VII.1 Scope of appendix

This appendix illustrates how the UII format is retrieved.

VII.2 IAC=UN (Dun and Bradstreet)

Let us assume the following:

UII = "UN043325711MH803120000001"

where the UII is encoded using six-bit characters.

As the IAC starts with "U", it is known that the IAC consists of two characters and has therefore the value "UN".

The initial part of the URN for IAC=UN is:

"urn:iso:std:iso-iec:15459.25S.UN"

The corresponding distinguished name is:

{ urnC="iso", urnC="std", urnC="iso=iec", urnC="15459", unrC="25S", urnC="UN" }

This distinguished name is used to read the addressed entry and it is assumed that the result is an attribute of type **`uiiFormat`** with the following value.

```
<UiiFormat>
  <subset>baseObject</subset>
  <next>
    <length>9</length>
  </next>
</UiiFormat>
```

This value indicates that the next level (the CIN) is always nine characters long. Accordingly, the CIN is "043325711".

The URN is updated to include the value of the CIN:

"urn:iso:std:iso-iec:15459:UN. 043325711"

As no filter is supplied, and the **`subset`** is suggested to be **`baseObject`**, the next operation should be a search with an empty filter, a subset equal to **`baseObject`** and a base object name corresponding to the above URN.

An **`ldapUrl`** may be returned together with an attribute of type **`uiiFormat`** with the following value:

```
<UiiFormat>
  <subset>wholeSubtree</subset>
  <next>
    <filter>
      <item>
        <type>1.2.3.4.5</type>
      </item>
    </filter>
  </next>
</UiiFormat>
```

It can be assumed that the result also included an attribute of type **`ldapURL`** pointing to the site of the identified organization. A complete search is performed at that site, i.e., an empty distinguished name

is used as a base object and the filter is an attribute value assertion of type **uii** with the complete UII as value. The result is the return of information about the item to which the RFID tag is attached.

VII.3 IAC=J example (UPU)

In the case of an IAC equals J, the issuing agency is the Universal Postal Union (UPU). In this case, the CIN (or the issuer code according to UPU) field is either two or three characters long.

Let us assume the following:

UII = "JDKA01NR0C09602291710C0AB5A4"

As the IAC starts with "J", it is known that the IAC consists of one character.

The initial part of the URN for IAC=J is:

"urn:iso:std:iso-iec:15459:J"

This distinguished name corresponding to this URN is used to read the addressed entry. It can then be assumed that the result is an attribute of type **uiiFormat** with the following value.

```
<UiiFormat>
  <subset>oneLevel</subset>
  <next>
    <filter>
      <or>
        <item>
          <type>2.5.4.89</type>
          <length>2</length>
        </item>
        <item>
          <type>2.5.4.89</type>
          <length>3</length>
        </item>
      </or>
    </filter>
</UiiFormat>
```

Based on that value, a search is generated with the base object as the same as for the previous search and with a subset stating **oneLevel**. A filter is generated with two attribute value assertions against an attribute of type **urnC**, one with the two characters "DK" and the other with the three characters "DKA". These two attribute value assertions are OR'ed, which means that an entry will be returned if an entry has an attribute of type **urnC** with either the value "DK" or "DKA". If there is an entry immediately subordinate to the base object with an attribute of type urnC with two values, which happen to be "DK" and "DKA" and no other immediately subordinate entry has any of those values, the search will then return the single entry that matches the filter together with the attribute with the two values and the distinguished name of the entry. The last component of that distinguished name has the value "DKA", which then goes into the URN corresponding to the entry, this means that the URN is:

"urn:iso:std:iso-iec:15459:J.DKA"

The used filter is now checked against the returned attribute of type **urnC** and both values satisfy the filter. The **UIIpointer** is now updated with the length of the longest attribute value, which is three.

If the returned entry also contains an attribute of type **uiiFormat** with the following value:

```
<UiiFormat>
  <subset>baseObject</subset>
  <next>
    <length>2</length>
  </next>
</UiiFormat>
```

The next component of the URN is two characters long with the value "01". The URN is now:

"urn:iso:std:iso-iec:15459:J.DKA.01"

This distinguished name corresponding to this URN name is now used to read the addressed entry and it will be assumed that the result is an attribute of type **ldapURL** pointing to a new site and an attribute of type **uiiFormat** with the following value.

```
<UiiFormat>
  <baseObject>urn:abc.def.ghi</baseObject>
  <subset>wholeSubtree</subset>
  <next>
    <filter>
      <item>
        <type>1.2.3.4.5</type>
      </item>
    </filter>
  </next>
</UiiFormat>
```

A search is now performed against the new site using the base object as specified and using a subset that dictates searching all entries in the subtree having its root in the entry corresponding to the specified URN. An attribute value assertion has to be generated against the attribute of type {1.2.3.4.5} using the whole UII as value. The result is the information about the item to which the RFID tag is affixed.

Appendix VIII

The format attribute type

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

VIII.1 Introduction

This appendix provides a tutorial overview of the directory attribute types used to provide RFID format information.

VIII.2 EPC format information

An attribute of the **epcFormat** attribute type specifies how an EPC bit string may be partitioned into components.

An EPC consists of a number of fields. Only the formats of the fields following the header, the filter and the partition fields are considered. For each of the subsequent fields the following information is provided:

a) The length of the field in number of bits.

b) The limitation on the resulting character string:
   - the maximum number of characters to which the bit string may be converted; or
   - if the string consists of all numeric characters, the value shall not exceed a specified value.

c) The specifications for the resulting string shall result in either:
   - the string shall consist of numeric characters and shall be prefixed with '0's to pad the string to the maximum length, as specified in b):

     the string shall consist of numeric characters without padding; or
   - the string shall consist of alphanumeric characters limited to the ASCII character set.

The following information is independent of the individual fields:

a) The number of the resulting octet that needs to be shifted to the front in case of a character representation of the EPC.

b) A check digit to be added.

c) To make a URN globally unique a prefix has to be added. The prefix has the format "urn:epc:id:<epc scheme>:", where "<epc scheme>" is either "sgtin", "sscc", "sgln", "grai", "giai", "gdti", "gsrn", "gid" or "usdod".

VIII.3 UII format information

An attribute of the **uiiFormat** attribute type specifies how to return information related to an UII string by specifying how a subsequent search should be performed. It is assumed that the UII has been resolved down to a specific point, and that some of the subsequent characters constitute the next field. This means that the distinguished name of the entry corresponding to the latest UII field is resolved and that the RDN corresponding to the subsequent UII field is to be determined. This is done by finding the value of the subsequent field to be used as the RDN value for the entry representing the subsequent field.

A value of this attribute type has the following syntax expressed in ASN.1 terms:

```
UiiFormat ::= SEQUENCE {
  baseObject  URI,
  subset      ENUMERATE {
    baseObject   (0),
    oneLevel     (1),
    wholeSubtree (2) } DEFAULT baseObject,
  next        CHOICE {
    length      INTEGER,
    filter      UiiFilter } }

UiiFilter ::= CHOICE {
  item  [0]  UiiItem,
  and   [1]  SET OF UiiFilter,
  or    [2]  SET OF UiiFilter,
  not   [3]  UiiFilter }

UiiItem ::= SEQUENCE {
  type   ATTRIBUTE.&id OPTIONAL,
  length INTEGER       OPTIONAL }
```

The **baseObject** component shall contain the URN corresponding to the base object of the search. If this component is absent, the search is suggested to start at the root.

The **subset** component gives a recommendation as to how to fill the **subset** component of the **Filter** in a subsequent search operation. **baseObject** specifies that the search shall only be performed against the base object. **oneLevel** specifies that only the entries immediately subordinate to the base object are to be searched. **wholeSubtree** specifies that all entries in the subtree having the base object as root are to be searched.

The **next** component gives some information about the next field of the UII:

– The choice **length** subcomponent shall be taken if the length of the following UII field has a fixed length and the subcomponent signals the length in characters.

– The choice **filter** shall be taken if the following UII field does not have a fixed length. It provides guidance on how filter items should be constructed to explore the actual length of the next UII field. The **UiiFilter** data type has a recursive structure similar to the structure of the **Filter** data type as defined by [ITU-T X.511] allowing for specification for a filter of arbitrary complexity.

  The recommendation for a particular filter item is given by the **UiiItem** data type

  i) the **type** subcomponent specifies the attribute type to be used in the attribute value assertion; and

  ii) the **length** subcomponent specifies how many characters are to be used as the value of the attribute value assertion.

NOTE – Only in rare circumstances will the full power of this rather complicated attribute type be used. It will only come in use when there is no, or little, discipline on how UIIs are constructed. In the case the next UII field has a fixed length, the value of the attribute gives the number of characters to be used for the next RDN. By adding this RDN to the distinguished name at hand, the entry representing the entry of the subsequent field may be read. As another example: If the length of the subsequent field is either two or three characters long, the value of the **uiiFormat** attribute indicates that two filter items should be constructed, one filter item consisting of two characters and another of three characters. These two filter items should then be OR'ed in the filter for locating the entry that corresponds to the subsequent field.

Bibliography

[b-ITU-T F.771] Recommendation ITU-T F.771 (2008), *Service description and requirements for multimedia information access triggered by tag-based identification*.

[b-ITU-T H.621] Recommendation ITU-T H.621 (2008), *Architecture of a system for multimedia information access triggered by tag-based identification*.

[b-ITU-T H.642.1] Recommendation ITU-T H.642.1 (2012), *Multimedia information access triggered by tag-based identification – Identification scheme*.

[b-ITU-T Y.2091] Recommendation ITU-T Y.2091 (2007), *Next Generation Networks – Frameworks and functional architecture models Terms and definitions for Next Generation Networks*.

[b-ISO/IEC 646] ISO 646:1991, *Information processing – ISO 7-bit coded character set for information interchange*.

[b-ISO 3779] ISO 3779:2009, *Road vehicles – Vehicle identification number (VIN) – Content and structure*.

[b-ISO/IEC 15418] ISO/IEC 15418:2009, *Information technology – Automatic identification and data capture techniques – GS1 Application Identifiers and ASC MH10 Data Identifiers and maintenance*.

[b-ISO/IEC 15962] ISO/IEC 15962:2013, *Information technology – Radio frequency identification (RFID) for item management – Data protocol: data encoding rules and logical memory functions*.

[b-ISO/IEC 15963] ISO/IEC 15963:2009, *Information technology – Radio frequency identification for item management – Unique identification for RF tags*.

[b-GS1TAGDATA] GS1 EPCglobal, *EPC Tag Data Standard*, Version 1.5.

[b-GS1GS] GS1 General Specification, Version 10, Issue 1, Jan-2010.

[b-MH10.8.2] MH10.8.2, *Data Identifier & Application Identifier Standard*.

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	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

Printed in Switzerland
Geneva, 2015