

International Telecommunication Union

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**H.350**

(05/2011)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Directory services  
architecture for audiovisual and multimedia services

---

**Directory services architecture for multimedia  
conferencing**

Recommendation ITU-T H.350



ITU-T H-SERIES RECOMMENDATIONS  
**AUDIOVISUAL AND MULTIMEDIA SYSTEMS**

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
<b>Directory services architecture for audiovisual and multimedia services</b>	<b>H.350–H.359</b>
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	H.700–H.719
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779

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

# Recommendation ITU-T H.350

## Directory services architecture for multimedia conferencing

### Summary

Recommendation ITU-T H.350 describes a directory services architecture for multimedia conferencing using lightweight directory access protocol (LDAP). Standardized directory services can support the association of persons with endpoints, searchable white pages, and clickable dialling. Directory services can also assist in the configuration of endpoints, and user authentication based on authoritative data sources. This Recommendation describes a standardized LDAP schema to represent endpoints on the network and associate those endpoints with users. It discusses design and implementation considerations for the interrelation of video and voice-specific directories, enterprise directories, call servers and endpoints.

This revised version of Recommendation ITU-T H.350 introduces several enhancements and clarifications to the previous version, primarily the addition of ITU-T X.500 directories support and authentication across multiple directories, and clarifications of commPrivate and commOwner.

This Recommendation includes an electronic attachment containing two schema configuration files for commURIObject and CommObject.

### History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T H.350	2003-08-06	16
1.0	ITU-T H.350 attachment	2003-08-06	16
2.0	ITU-T H.350	2011-05-14	16

### Keywords

Directory services, ITU-T H.235.0, ITU-T H.320, ITU-T H.323, LDAP, SIP, ITU-T X.500.

## 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/>.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## Table of Contents

	<b>Page</b>
1 Scope .....	1
1.1 Design goals .....	2
1.2 Extending the schema.....	3
2 References.....	4
3 Definitions .....	5
4 Abbreviations.....	5
5 Conventions .....	6
6 commURIObject definition .....	6
6.1 commURIObject.....	6
6.2 commURI .....	6
7 CommObject definition .....	7
7.1 commObject .....	7
7.2 commUniqueId .....	7
7.3 commOwner .....	8
7.4 commPrivate.....	8
8 CommObject LDIF files .....	9
8.1 LDIF for commURIObject.....	9
8.2 LDIF for commObject.....	10
9 Using ITU-T H.350 with ITU-T X.500 directories .....	12
9.1 IMPORTS of ITU-T X.500 ASN.1 .....	12
9.2 commObjectASN1.asn .....	12
9.3 commURIASN1.asn.....	14
Annex A – Indexing profile .....	16
Appendix I – Implementation considerations .....	17
I.1 Relationship of enterprise directories to commObject directories.....	17
Appendix II – Call flows.....	21
II.1 Call flow scenarios .....	21
Appendix III – Electronic attachments: schema configuration files for commURIObject and commObject.....	24
Bibliography.....	25



## **Recommendation ITU-T H.350**

### **Directory services architecture for multimedia conferencing**

#### **1 Scope**

This Recommendation<sup>1</sup> describes a directory services architecture for multimedia conferencing using lightweight directory access protocol (LDAP). Standardized directory services can support the association of persons with endpoints, searchable white pages, and clickable dialling. Directory services can also assist in the configuration of endpoints, and user authentication based on authoritative data sources. This Recommendation describes a standardized LDAP schema to represent endpoints on the network and associate those endpoints with users. It discusses design and implementation considerations for the interrelation of video and voice-specific directories, enterprise directories, call servers and endpoints.

The use of a common, authoritative data source for call server, endpoint, user, authentication and white pages information is an important aspect of large scale multimedia conferencing environments. Without a common data source, service providers must create separate processes to manage each of these functions. By standardizing the LDAP schema used to represent the underlying data, products from different system vendors can be deployed together to create an overall application environment. For example, a white pages search engine developed by one provider could serve directory information to IP telephones produced by a second provider, with signalling managed by a call server produced by yet a third provider. Each of these disparate systems can access the same underlying data source, reducing or eliminating the need to coordinate the separate management of each system. A significant benefit to the user is that the management of this data can be incorporated into existing customer management tools, allowing for quick and flexible scaling up of applications. Indeed, many technology providers have already incorporated LDAP into their products, but have been forced to do so without the benefit of a standardized schema. This Recommendation represents an effort to standardize those representations to improve interoperability and performance.

While URLs are already standardized for several conferencing protocols, their representation in a directory is not. This Recommendation supports a standardized way for URLs to be searched and located. This is a necessary step to support 'clickable dialling'.

Management of endpoint configurations can be improved if the correct settings are stored by the service provider in a location that is accessible to both service provider and endpoint. LDAP provides a convenient storage location that can be accessed by both call server and endpoint; thus it is possible to use the directory to support the endpoint configuration, which is important for simplified operation and supporting user mobility. Note that other technologies also support endpoint configuration, notably the use of SNMP for complete configuration and DNS SRV resource records for obtaining registration server addresses. Therefore, ITU-T H.350 should be viewed not as an authoritative endpoint configuration architecture, but rather one tool that can assist with this task. Note that the use of ITU-T H.350 has as a feature endpoint specific configuration, where it is desirable that each endpoint have a unique configuration.

This architecture uses a generic object class, called commObject, to represent attributes common to any video or voice protocol. Auxiliary classes represent specific protocols, such as ITU-T H.323, ITU-T H.235.x, or ITU-T H.320, as described in the ITU-T H.350.x series of Recommendations. Multiple ITU-T H.350.x classes can be combined to represent endpoints that support more than one protocol. For example, endpoints that support ITU-T H.323, ITU-T H.235.x and ITU-T H.320

---

<sup>1</sup> This Recommendation includes an electronic attachment containing schema configuration files for commURIObject and commObject.

would include ITU-T H.350, ITU-T H.350.1, ITU-T H.350.2, and ITU-T H.350.3 in their LDAP representations. Furthermore, each entry should contain commObject to serve as the entry's structural object class.

There are two basic components in the architecture. The commURI object is a class whose only purpose is to link a person or resource to a commObject. By placing a commURI 'pointer' in an individual's directory entry, that individual becomes associated with the particular targeted commObject. Similarly, each commObject contains a pointer, called commOwner, which points to the individual or resource that is associated with the commObject. In this way, people or resources can be associated with endpoints. The only change required to the enterprise directory is the addition of the simple object class commURI. CommObject data may be instantiated in the same or an entirely separate directory, thus allowing flexibility in implementation.

## **1.1 Design goals**

Large-scale deployments of IP video and voice services have demonstrated the need for complementary directory services middleware. Service administrators need call servers that are aware of enterprise directories to avoid duplication of account management processes. Users need 'white pages' to locate other users with whom they wish to communicate. All of these processes should pull their information from canonical data sources in order to reduce redundant administrative processes and ensure information accuracy. The following design criteria are established for this architecture. The architecture will:

- 1) enable endpoint information to be associated with people. Alternatively, it enables endpoint information to be associated with resources such as conference rooms or classrooms;
- 2) enable online searchable "white pages" where dialling information (e.g., endpoint addresses) can be found, along with other "traditional" directory information about a user, such as name, address, telephone, e-mail, etc.;
- 3) enable all endpoint information to be stored in a canonical data source (the Directory), rather than local to the call server, so that endpoints can be managed through manipulations of an enterprise directory, rather than by direct entry into the call server;
- 4) support the creation of very large scale distributed directories. These include white pages "portals" that allow searching for users across multiple institutional directories. In this application, each enterprise directory registers itself with (or is unknowingly discovered by) a directory of directories that is capable of searching across multiple LDAP directories;
- 5) be able to support multiple instances of endpoints per user or resource;
- 6) represent endpoints that support more than one protocol, for example, endpoints that are both ITU-T H.320 and ITU-T H.323;
- 7) store enough information about endpoint configuration so that correct configuration settings can be documented to end users on a per-endpoint basis, as a support tool, or loaded automatically into the endpoint;
- 8) be extendable as necessary to allow implementation-specific attributes to be included;
- 9) be non-invasive to the enterprise directory, so that support for multimedia conferencing can be added in a modular fashion without significant changes to the enterprise directory.

The scope of this Recommendation does not include extensions of functionality to protocols as defined within the protocols themselves. It is not the intent of this Recommendation to add features, but merely to represent existing protocol attributes. The exception to this case is when functionality is implied by the directory itself, such as the commPrivate attribute.

## 1.2 Extending the schema

ITU-T H.350 object classes may be extended as necessary for specific implementations. For example, a class may be extended to support billing reference codes. Extensions to the schema are not considered as part of this Recommendation and do not signify compliance.

In some cases it may be necessary to extend the ITU-T H.350 schemas in order to represent more information than is supported by the Recommendations. This may be important for developers that implement proprietary endpoint functionality that needs to be represented by attributes in the directory. It may also be important for enterprise applications. For example 'modelNumber', and 'accountNumber' are examples of attributes that are not defined in this Recommendation, but may be useful if implemented. Adding attributes to this architecture must be done in a way that does not break compatibility with this Recommendation.

A full discussion of schema design and extension is beyond the scope of this Recommendation. See [IETF RFC 4510] for details. Two basic approaches to schema extension that do not break compatibility with this Recommendation are extension through subclass and extension through the use of auxiliary classes.

### 1.2.1 Extension through subclass

It is possible to create a subclass of an existing predefined object class in order to add new attributes to it. To create a subclass, a new object class must be defined, that is a subclass of the existing one, by indicating in the definition of the new class that the existing class is its superior. Once the subclass is created, new attributes can be defined within it.

The following example shows how the commObject class can be subclassed in order to add an attribute to represent a billing account and a billing manager.

```
objectclass ( BillingInfo-OID
NAME 'BillingInfo'
DESC 'Billing Reference Information'
SUP commObject STRUCTURAL
MAY ( BillingAccount $ BillingManager $ )
)
```

Note that BillingInfo-OID must be replaced by an actual OID. Also note that, whenever a structural class is extended, its subclass must also be structural.

The following sample entry shows the newly created attributes. This example also uses [ITU-T H.350.1] for h323Identity.

```
dn: commUniqueId=2000,ou=h323Identity, dc=company, dc=com
objectclass: top
objectclass: commObject
objectclass: h323Identity
objectclass: BillingInfo
commUniqueId: 2000
BillingAccount: 0023456
BillingManager: John Smith
```

Note that this example and approach demonstrate an extension of the general commObject object class, and not any individual ITU-T H.350.x classes. If it is desired to extend an ITU-T H.350.x auxiliary class, then that should be accomplished through the definition of additional auxiliary classes that support the desired attributes, as described in clause 1.2.2.

### 1.2.2 Extension through the use of auxiliary classes

It is possible to add attributes to an LDAP entry by defining an auxiliary class containing the new attributes and applying those attributes to instantiated values in the directory. The auxiliary class will not be subclassed from any existing object class. Note that it should have the special class top as its superior. The following example creates the same billing account and billing manager attributes as the previous example, but does so by defining them in their own auxiliary class.

```
objectclass ( BillingInfo-OID
NAME 'BillingInfo'
DESC 'Billing Reference Information'
SUP top AUXILIARY
MAY ( BillingAccount $ BillingManager $ )
)
```

Note how the superior was changed from commObject to top and the object class changed from being a structural to auxiliary.

It is recommended that all attributes in the auxiliary class be optional rather than mandatory. In this way, the auxiliary object class itself can be associated with an entry regardless of whether any values for its attributes are present.

The following example shows a sample endpoint that utilizes the new auxiliary class and attributes. This example also uses [ITU-T H.350.1] for h323Identity.

```
dn: commUniqueId=2000,ou=h323Identity, dc=company, dc=com
objectclass: top
objectclass: commObject
objectclass: BillingInfo
commUniqueId: 2000
BillingAccount: 0023456
BillingManager: John Smith
```

### 1.2.3 Object identifiers

An attribute's object identifier (OID) is a unique numerical identifier usually written as a sequence of integers separated by dots. For example, the OID for the commUniqueId is 0.0.8.350.1.1.2.1.1. All attributes must have an OID. OIDs can be obtained from anyone who has one and is willing to delegate a portion of it as an arc, keeping a record of the arc to avoid duplication. Furthermore, the Internet Assigned Numbers Authority (IANA) gives out OIDs to any organization that asks.

## 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 H.350.x] Recommendations ITU-T H.350.x-series (in force), *Directory services architecture for H.323*.

[ITU-T X.500] Recommendation ITU-T X.500 (2008) | ISO/IEC 9594-1:2008, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services*.

[ITU-T X.501] Recommendation ITU-T X.501 (2008) | ISO/IEC 9594-2:2008, *Information technology – Open Systems Interconnection – The Directory: Models*.

- [ITU-T X.509] Recommendation ITU-T X.509 (2008) | ISO/IEC 9594-8:2008, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks.*
- [ITU-T X.511] Recommendation ITU-T X.511 (2008) | ISO/IEC 9594-3:2008, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition.*
- [ITU-T X.518] Recommendation ITU-T X.518 (2008) | ISO/IEC 9594-4:2008, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation.*
- [ITU-T X.519] Recommendation ITU-T X.519 (2008) | ISO/IEC 9594-5:2008, *Information technology – Open Systems Interconnection – The Directory: Protocol specifications.*
- [ITU-T X.520] Recommendation ITU-T X.520 (2008) | ISO/IEC 9594-6:2008, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types.*
- [ITU-T X.525] Recommendation ITU-T X.525 (2008) | ISO/IEC 9594-9:2008, *Information technology – Open Systems Interconnection – The Directory: Replication.*
- [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.*

### 3 Definitions

This Recommendation defines the following terms:

**3.1 call server:** A protocol-specific signalling engine that routes video or voice calls on the network. In [b-ITU-T H.323], this entity is a gatekeeper. In session initiation protocol (SIP), this entity is a SIP proxy server. Note that not all signalling protocols use a call server.

**3.2 endpoint:** A logical device that provides video and/or voice media encoding/decoding, and signalling functions. Examples include:

- 1) a group teleconferencing appliance that is located in a conference room;
- 2) an IP telephone;
- 3) a software program that takes video and voice from a camera and microphone, encodes it and applies signalling using a host computer.

**3.3 enterprise directory:** A canonical collection of information about users in an organization. Typically this information is collected from a variety of organizational units to create a whole. For example, Human Resources may provide name and address, Telecommunications may provide the telephone number, Information Technology may provide the e-mail address, etc. For the purposes of this architecture, it is assumed that an enterprise directory is accessible via LDAP.

**3.4 white pages:** An application that allows end users to look up the address of another user. This may be web-based or use some other user interface.

### 4 Abbreviations

This Recommendation uses the following abbreviations:

CN Common Name

DN	Distinguished Name
LDAP	Lightweight Directory Access Protocol
	NOTE – This is consistent with [IETF RFC 4510].
LDIF	LDAP Data Interchange Format
RDN	Relative Distinguished Name
SIP	Session Initiation Protocol
SRV	Service

## 5 Conventions

In this Recommendation, the following conventions are used:

"Shall" indicates a mandatory requirement.

"Should" indicates a suggested but optional course of action.

"May" indicates an optional course of action rather than a recommendation that something takes place.

References to clauses, subclauses, annexes and appendices refer to those items within this Recommendation unless another specification is explicitly listed.

## 6 commURIObject definition

An auxiliary object class that contains the commURI attribute. This attribute is added to a person or resource object to associate one or more commObject instances with that object. Its values are LDAP URIs that point to the associated commObject instances, for example, to a user's ITU-T H.323 conferencing station and SIP IP phone. Note that multiple instances of commURI need not point to the same commObject directory. In fact, each commURI instance could point to an endpoint managed by a different service provider.

### 6.1 commURIObject

```

OID: 0.0.8.350.1.1.1.2.1
objectclasses: (0.0.8.350.1.1.1.2.1
NAME 'commURIObject'
DESC 'object that contains the URI attribute type'
SUP top AUXILIARY
MAY ( commURI )
)

```

### 6.2 commURI

```

OID: 0.0.8.350.1.1.1.1.1
attributetypes: ( 0.0.8.350.1.1.1.1.1
NAME 'commURI'
DESC 'Labelled URI format to point to the distinguished name of the commUniqueId'
EQUALITY caseExactMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

```

Application utility class

Standard

Number of values

multi

## Definition

Labelled URI containing an LDAP URL identifying the directory containing the referenced commObject instance. The search filter specified by this LDAP URL shall specify an equality search of the commUniqueId attribute of the commObject class.

## Permissible values (if controlled)

## Notes

Used to find the endpoint of the user in question. The label field may be used to represent the function of the endpoint, such as 'home IP phone' or 'desktop video' for user interface display purposes.

Note that the label portion of the field may contain spaces as in the example below showing 'desktop video'.

## Semantics

Example applications for which this attribute would be useful

## Example (LDIF fragment)

```
commURI: ldap://directory.acme.com/dc=acme,dc=com??sub?(commUniqueId=bob) desktop
video
```

## 7 CommObject definition

Abstraction of video or voice over IP device. The commObject class permits an endpoint (ITU-T H.323 endpoint or SIP user agent or other protocol endpoint) and all their aliases to be represented by a single entry in a directory. Note that every directory entry should contain commObject as the entry's structural object class. That entry may also contain ITU-T H.350.x auxiliary classes.

### 7.1 commObject

```
OID: 0.0.8.350.1.1.2.2.1
objectclasses: (0.0.8.350.1.1.2.2.1
NAME 'commObject'
DESC 'object that contains the Communication attributes'
SUP top STRUCTURAL
MUST commUniqueId
MAY ( commOwner $ commPrivate )
)
```

### 7.2 commUniqueId

```
OID: 0.0.8.350.1.1.2.1.1
attributetypes: (0.0.8.350.1.1.2.1.1
NAME 'commUniqueId'
DESC 'To hold the endpoints unique Id'
EQUALITY caseIgnoreIA5Match
SUBSTR caseIgnoreIA5SubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

## Application utility class

standard

## Number of values

multi

## Definition

The endpoint's unique ID.

Permissible values (if controlled)

Notes

This is the RDN of this object. In practice, there will always be one and only one `commUniqueId` for every endpoint. This attribute uniquely identifies an endpoint in the `commObject` directory. It must be unique within that directory, but need not be unique globally. This attribute has no relationship to the enterprise directory.

Semantics

Example applications for which this attribute would be useful

Example (LDIF fragment)

```
commUniqueId: bob
```

### 7.3 commOwner

```
OID: 0.0.8.350.1.1.2.1.2
attributetypes: 0.0.8.350.1.1.2.1.2
NAME 'commOwner'
DESC 'Labelled URI to point back to the original owner'
EQUALITY caseExactMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

Application utility class

Standard

Number of values

multi

Definition

Labelled URI format to point back to the person or resource object associated with this entry.

Permissible values (if controlled)

Notes

Used as a reverse entry finder of the owner(s). This attribute may point to groups. Note that this URI can point to a `cn`, but in applications where it is desired to bind authentication information across both the `commObject` and enterprise directories, it may be desirable that `commOwner` points to a `dn` rather than a `cn`, thus uniquely identifying the owner of the `commObject`.

Semantics

Example applications for which this attribute would be useful

Example (LDIF fragment)

```
commOwner: ldap://directory.acme.com/dc=acme,dc=com??sub?(cn=bob%20smith)
commOwner: uid=bob,ou=people,dc=acme,dc=com
```

### 7.4 commPrivate

```
OID: 0.0.8.350.1.1.2.1.3
attributetypes: (0.0.8.350.1.1.2.1.3
NAME 'commPrivate'
DESC 'To decide whether the entry is visible to world or not'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

Application utility class

Standard

Number of values

multi

Definition

To be used by the user and indicate privacy options for an endpoint, i.e., unlisted number.

Permissible values (if controlled)

Notes

This attribute is defined as String. The commPrivate attribute has the string value of either 'true' or 'false'. Future versions of this Recommendation may develop a larger controlled vocabulary for this attribute to accommodate multiple types of privacy. For compatibility purposes, values defined as 'false' or null (commPrivate absent of the commObject) shall be assumed to be non-private.

Semantics

Example applications for which this attribute would be useful

Example (LDIF fragment)

```
commPrivate: true
```

## 8 CommObject LDIF files

This clause contains a schema configuration file for commURIObject and commObject that can be used to configure an LDAP server to support these classes.

### 8.1 LDIF for commURIObject

```
# Communication Object Schema
#
# Schema for Representing Communication Objects in an LDAP Directory
#
# Abstract
#
# This Recommendation defines the schema for representing communication
# objects in an LDAP directory [LDAPv3]. It defines schema elements
# to represent a communication object URI [commURIObject].
#
#
#
#           .1 = Communication related work
#           .1.1 = commURIObject
#           .1.1.1 = attributes
#           .1.1.2 = objectclass
#           .1.1.3 = syntax
#
# Attribute Type Definitions
#
#   The following attribute types are defined in this Recommendation:
#
#       commURI
dn: cn=schema
changetype: modify
#
# if you need to change the definition of an attribute,
#       then first delete and re-add in one step
#
# if this is the first time you are adding the commObject
# objectclass using this LDIF file, then you should comment
```

```

# out the delete attributetypes modification since this will
# fail. Alternatively, if your ldapmodify has a switch to continue
# on errors, then just use that switch -- if you are careful
#
delete: attributetypes
attributetypes: (0.0.8.350.1.1.1.1.1 NAME 'commURI' )
-
#
# re-add the attributes -- in case there is a change of definition
#
#
add: attributetypes
attributetypes: (0.0.8.350.1.1.1.1.1
    NAME 'commURI'
    DESC 'Labelled URI format to point to the distinguished name of the commUniqueId'
    EQUALITY caseExactMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
-
# Object Class Definitions
#
# The following object classes are defined in this Recommendation:
#
#     commURIObject
#
# commURIObject
#
# This auxiliary object class represents a URI attribute type
#
#
delete: objectclasses
objectclasses: (0.0.8.350.1.1.1.2.1 NAME 'commURIObject' )
-
add: objectclasses
objectclasses: (0.0.8.350.1.1.1.2.1
    NAME 'commURIObject'
    DESC 'object that contains the URI attribute type'
    SUP top AUXILIARY
    MAY ( commURI )
    )
-
#
# end of LDIF
#

```

## 8.2 LDIF for commObject

```

# Communication Object Schema
#
# Schema for Representing Communication Objects in an LDAP Directory
#
# Abstract
#
# This Recommendation defines the schema for representing Communication
# objects in an LDAP directory [LDAPv3]. It defines schema elements
# to represent a communication object [commObject].
#
#
#         .1 = Communication related work
#         .1.2 = commObject
#         .1.2.1 = attributes
#         .1.2.2 = objectclass
#         .1.2.3 = syntax
#
#
# Attribute Type Definitions
#
# The following attribute types are defined in this Recommendation:
#
#     commUniqueId
#     commOwner

```

```

#         commPrivate
dn: cn=schema
changetype: modify
#
# if you need to change the definition of an attribute,
#         then first delete and re-add in one step
#
# if this is the first time you are adding the commObject
# objectclass using this LDIF file, then you should comment
# out the delete attributetypes modification since this will
# fail. Alternatively, if your ldapmodify has a switch to continue
# on errors, then just use that switch -- if you are careful
#
delete: attributetypes
attributetypes: (0.0.8.350.1.1.2.1.1 NAME 'commUniqueId' )
attributetypes: (0.0.8.350.1.1.2.1.2 NAME 'commOwner' )
attributetypes: (0.0.8.350.1.1.2.1.3 NAME 'commPrivate' )
-
#
# re-add the attributes -- in case there is a change of definition
#
#
add: attributetypes
attributetypes: (0.0.8.350.1.1.2.1.1
    NAME 'commUniqueId'
    DESC 'To hold the endpoints unique Id'
    EQUALITY caseIgnoreIA5Match
    SUBSTR caseIgnoreIA5SubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
attributetypes: (0.0.8.350.1.1.2.1.2
    NAME 'commOwner'
    DESC 'Labelled URI to point back to the original owner'
    EQUALITY caseExactMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
attributetypes: (0.0.8.350.1.1.2.1.3
    NAME 'commPrivate'
    DESC 'To decide whether the entry is visible to world or not'
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
-
# Object Class Definitions
#
#     The following object classes are defined in this Recommendation:
#
#         commObject
#
# commObject
#
delete: objectclasses
objectclasses: (0.0.8.350.1.1.2.2.1 NAME 'commObject' )
-
add: objectclasses
objectclasses: (0.0.8.350.1.1.2.2.1
    NAME 'commObject'
    DESC 'object that contains the Communication attributes'
    SUP top STRUCTURAL
    MUST commUniqueId
    MAY ( commOwner $ commPrivate )
    )
-
#
# end of LDIF
#

```

## 9 Using ITU-T H.350 with ITU-T X.500 directories

LDAP object classes are based upon the ITU-T X.500 directory architecture. While ITU-T H.350 specifies use with LDAP, it is also possible to use ITU-T H.350 with ITU-T X.500 directories. To do so requires ASN.1 definitions of the object classes and attributes, rather than the LDAP definitions and LDIF files included in the original ITU-T H.350.x-series Recommendations.

Note that ASN.1 definitions found in ITU-T H.350.x-series Recommendations may rely on the ASN.1 definitions given here for basic data structures.

### 9.1 IMPORTS of ITU-T X.500 ASN.1

To satisfy all the IMPORTS clauses, the following modules are needed:

- BasicAccessControl ([ITU-T X.501])
- DSAOperationalAttributeTypes ([ITU-T X.501])
- EnhancedSecurity ([ITU-T X.501])
- InformationFramework ([ITU-T X.501])
- OperationalBindingManagement ([ITU-T X.501])
- ServiceAdministration ([ITU-T X.501])
- UsefulDefinitions ([ITU-T X.501])
- AttributeCertificateDefinitions ([ITU-T X.509])
- AuthenticationFramework ([ITU-T X.509])
- CertificateExtensions ([ITU-T X.509])
- MTSAbstractService ([ITU-T X.509])
- PKIX1Implicit93 ([ITU-T X.509])
- DirectoryAbstractService ([ITU-T X.511])
- SpkmGssTokens ([ITU-T X.511])
- DistributedOperations ([ITU-T X.518])
- HierarchicalOperationalBindings ([ITU-T X.518])
- CommonProtocolSpecification ([ITU-T X.519])
- DirectoryOSIProtocols ([ITU-T X.519])
- DirectoryOperationalBindingTypes ([ITU-T X.519])
- OSIProtocolSpecification ([ITU-T X.519])
- SelectedAttributeTypes ([ITU-T X.520])
- DirectoryShadowAbstractService ([ITU-T X.525])
- ldap ([IETF RFC 4511])

It is noted that these modules can be downloaded from the ITU-T ASN.1 module database.

### 9.2 commObjectASN1.asn

```
CommObject { itu-t(0) recommendation(0) h(8) 350 1 cr(1) commObject(2) module(4) }
DEFINITIONS ::=
BEGIN
-- Communication Object Schema

-- Schema for Representing Communication Objects in an LDAP Directory

-- Abstract

-- This Recommendation defines the schema for representing Communication
```

```

-- objects in an LDAP directory [LDAPv3]. It defines schema elements
-- to represent a communication object [commObject].

--          .1 = Communication related work
--          .1.2 = commObject
--          .1.2.1 = attributes
--          .1.2.2 = objectclass
--          .1.2.3 = syntax

-- Attribute Type Definitions

-- The following attribute types are defined in this Recommendation:

--      commUniqueId
--      commOwner
--      commPrivate

IMPORTS

-- from Rec. ITU-T H.350

h350-cr, caseIgnoreIA5Match, caseIgnoreIA5SubstringsMatch
    FROM CommURI { itu-t(0) recommendation(0) h(8) 350 1 cr(1) commURI(1) module(4) }

-- from Rec. ITU-T X.501 | ISO/IEC 9594-2

ATTRIBUTE, OBJECT-CLASS, top
    FROM InformationFramework {joint-iso-itu-t ds(5) module(1) informationFramework(1)
6}

-- from Rec. ITU-T X.520 | ISO/IEC 9594-6

UnboundedDirectoryString, caseExactMatch
    FROM SelectedAttributeTypes {joint-iso-itu-t ds(5) module(1)
selectedAttributeTypes(5) 6} ;

commUniqueId ATTRIBUTE ::= {
    WITH SYNTAX IA5String
    EQUALITY MATCHING RULE caseIgnoreIA5Match
    SUBSTRINGS MATCHING RULE caseIgnoreIA5SubstringsMatch
    ID { at 1 } }

commOwner ATTRIBUTE ::= {
    WITH SYNTAX UnboundedDirectoryString
    EQUALITY MATCHING RULE caseExactMatch
    ID { at 2 } }

commPrivate ATTRIBUTE ::= {
    WITH SYNTAX IA5String
    ID { at 3 } }

-- Object Class Definitions

-- The following object classes are defined in this Recommendation:

--      commObject

-- commObject

commObject OBJECT-CLASS ::= {
    SUBCLASS OF { top }
    MUST CONTAIN { commUniqueId }
    MAY CONTAIN { commOwner | commPrivate }
    ID { oc 1 } }

co          OBJECT IDENTIFIER ::= { h350-cr co(2) }
at          OBJECT IDENTIFIER ::= { co at(1) }
oc          OBJECT IDENTIFIER ::= { co oc(2) }

END -- end of ASN.1

```

### 9.3 commURIASN1.asn

```
CommURI { itu-t(0) recommendation(0) h(8) 350 1 cr(1) commURI(1) module(4) }
DEFINITIONS ::=
BEGIN

-- Communication Object Schema

-- Schema for Representing Communication Objects in an LDAP Directory

-- Abstract

-- This Recommendation defines the schema for representing Communication
-- objects in an LDAP directory [LDAPv3]. It defines schema elements
-- to represent a communication object URI [commURIObject].

--           .1 = Communication related work
--           .1.1 = commURIObject
--           .1.1.1 = attributes
--           .1.1.2 = objectclass
--           .1.1.3 = syntax

IMPORTS

-- from Rec. ITU-T X.501 | ISO/IEC 9594-2
ATTRIBUTE, OBJECT-CLASS, MATCHING-RULE, top
    FROM InformationFramework {joint-iso-itu-t ds(5) module(1) informationFramework(1)
6}

-- from Rec. ITU-T X.520 | ISO/IEC 9594-6
UnboundedDirectoryString, caseExactMatch
    FROM SelectedAttributeTypes {joint-iso-itu-t ds(5) module(1)
selectedAttributeTypes(5) 6} ;

-- Attribute Type Definitions

-- The following attribute types are defined in this Recommendation:

-- commURI

commURI ATTRIBUTE ::= {
    WITH SYNTAX UnboundedDirectoryString
    EQUALITY MATCHING RULE caseExactMatch
    ID { at 1 } }

-- Object Class Definitions

-- The following object classes are defined in this Recommendation:

-- commURIObject

-- commURIObject
-- This auxiliary object class represents a URI attribute type

commURIObject OBJECT-CLASS ::= {
    SUBCLASS OF { top }
    MAY CONTAIN { commURI }
    ID { oc 1 } }

h350-cr OBJECT IDENTIFIER ::= { itu-t(0) recommendation(0) h(8) 350 1 cr(1) }
cu OBJECT IDENTIFIER ::= { h350-cr cu(1) }
at OBJECT IDENTIFIER ::= { cu at(1) }
oc OBJECT IDENTIFIER ::= { cu oc(2) }

caseIgnoreIA5Match MATCHING-RULE ::= {
    SYNTAX IA5String
    ID { 1 3 6 1 4 1 1466 109 114 2 } }
```

```
caseIgnoreIA5SubstringsMatch MATCHING-RULE ::= {  
    SYNTAX IA5String  
    ID { 1 3 6 1 4 1 1466 109 114 3 } }  
  
userSMIMECertificate ATTRIBUTE ::= {  
    WITH SYNTAX OCTET STRING  
    ID { 2 16 840 1 113730 3 1 40 } }  
  
END -- end of ASN.1
```

## **Annex A**

### **Indexing profile**

(This annex forms an integral part of this Recommendation.)

Indexing of attributes is an implementation-specific activity and depends upon the desired application. Non-indexed attributes can result in search times sufficiently long to render some applications unusable. Notably, user and alias lookup should be fast. This annex indexing profile describes an indexing configuration for commObject directories that will be optimized for use in directory of directories applications. Use of this profile is optional.

commURI: no recommendation

commUniqueId: equality

commOwner: presence

commPrivate: presence

# Appendix I

## Implementation considerations

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

### I.1 Relationship of enterprise directories to commObject directories

CommObject information is located separately from person or resource information. Its location may be a sub-tree of the larger enterprise directory or on a separate logical server. The person directory will continue to host traditional person or resource information such as name, telephone, address, etc. Additionally, it will have a commURI link to the commUniqueId attribute in commObject. Rather than extending the enterprise directory's person object class, this linking provides the following advantages:

- 1) Changes to the enterprise directory are not to be undertaken lightly and are often not under the administrative control of the video/voice over IP service provider.
- 2) Elements associated with video and voice over IP communications are very dynamic. The technology itself is changing quickly in relation to the enterprise directory. For example, changes to a specific protocol would require changes to the enterprise directory if its representation were inherited from an enterprise directory person object class and embedded within the enterprise directory. Separation allows changes to the commObject LDAP infrastructure without modifying the enterprise directory.
- 3) A call server may need to access commObject data very differently than other applications access the enterprise directory. A separate server can be tuned for performance and access policy to accommodate these implementation requirements, if so desired. For example, a call server may need to query the commObject server many times per second in order to handle real-time call processing, or it may read and cache many commObject attributes at once.

Any user or resource with multimedia conferencing capabilities should have an instance of commObject created and linked to an existing entry in the enterprise directory with a commURI. Call servers may operate in one of two ways. The simplest method is for the call server to periodically read instances of commObject into its internal endpoint table. The preferred and more scalable method is for the call server to query the commObject server each time it needs information, such as upon endpoint registration or call set-up.

#### I.1.1 People versus resources

Some multimedia conferencing implementations are heavily endpoint-oriented, whereas others are user oriented. For example, it is common to encounter a group video teleconferencing endpoint in a conference room the identity of which never changes. This endpoint may be referred to as 'Conference Room 201'. This endpoint is not associated with any particular person, but is associated with the resource Conference Room 201 and is shared by whoever needs to use the conference room.

On the other hand, some endpoints are user context-specific, deriving their identities from the current users. For example, when logging onto a computer as jdoe, a computer-based endpoint may configure itself with the address of jdoe as stored in that user's profile and register with a call server accordingly. Other users logging onto the same computer may have different identities associated with them, hence their registration messages will contain different identity information.

This dichotomy of users versus resources makes it difficult to associate endpoints with users or resources. Indeed, while person object classes are readily available, resource object classes are less so. Linking commObject to a person via a commURI generalizes this relationship. If a commOwner attribute is pointing to a person object class, then that commObject is associated with that person. If

a commOwner attribute is pointing to a resource object class, then that commObject is associated with that resource. Conversely, either people or resources can have commURI pointers that associate endpoints with them. Enterprise directories that only support people and not resources may choose to simply treat resources as people.

### I.1.2 Security and authentication

Most authentication realms are oriented toward people. Thus, shared resources such as conference room teleconferencing systems are often less secure, because they have no meaningful authentication identity associated with them. It is beyond the scope of this Recommendation to discuss authentication concerns, but implementors are strongly encouraged to thoroughly explore the security aspects of various architectural choices.

Access control lists and other security mechanisms associated with the directory are beyond the scope of this Recommendation. Implementors are encouraged to carefully consider privacy and security of the data in the directory.

### I.1.3 Authenticating across multiple directories for automatic endpoint configuration

In some implementations, ITU-T H.350 attributes are stored in the same enterprise directory as the people, or owner, information. However, other implementations have separate ITU-T H.350 and enterprise directories. One application of ITU-T H.350 is to enable a user to log onto the network using their single sign-on credentials, and have the endpoint download its configuration information from the ITU-T H.350 directory. If the ITU-T H.350 directory and enterprise directory are the same, then there is no problem. If the ITU-T H.350 directory and the enterprise directory are separate servers, then there is a security concern.

When an endpoint attempts to bind to an ITU-T H.350 directory, the ITU-T H.350 directory is not aware of the user's authentication credentials, because those are stored in a different directory, i.e., the enterprise directory. There must be some way for the ITU-T H.350 directory to determine the owner of a particular ITU-T H.350 entry and only allow access to that data to the owner. This can be accomplished by the use of the owner attribute as described in clause 2.21 of [b-IETF RFC 4519].

#### Example

An example of the use of **owner** is given below using specific information from the ViDeNet system. Be sure to not use these sample OIDs, domain or other ViDe specific values listed in the example.

Step one is to create an object class that would contain the attribute owner:

```
objectclass ( 1.3.6.1.4.1.10411.3.1.1.4
    NAME 'VIDEOwner'
    AUXILIARY
    SUP top
    MAY ( owner )
)
```

Step two is to have this new object class part of the endpoint or SIP UA entry.

Here is an example of a complete entry showing how to also populate the owner entry.

```
dn: commUniqueId=30,ou=commidentity,dc=vide,dc=net
objectClass: top
objectClass: commObject
objectClass: h323Identity
objectClass: h235Identity
objectClass: VIDEOwner
objectClass: SIPIdentity
objectClass: h320Identity
objectClass: genericIdentity
commUniqueId: 30
```

```
h323IdentityEndpointType: Terminal
commOwner: ldap://videnet.unc.edu/dc=vide,dc=net??sub?(uid=doe)
owner: uid=doe,ou=people,dc=vide,dc=net
h235IdentityEndpointID: JoeDoe
h323IdentitydialedDigits: 00112971208
h323IdentityGKDomain: 152.2.17.189
h235IdentityPassword: testing123
```

This is the section that should be entered in the commObject server.

```
database meta
suffix "ou=people,dc=vide,dc=net"
uri ldap://videnet.unc.edu/ou=people,dc=vide,dc=net"
lastmod off
```

On the Enterprise directory side, this is how you would protect the entry. Change the admin information account and the domain name from the example.

```
access to attr=h235IdentityPassword
  by dnattr="owner" write
  by self write
  by anonymous auth
  by dn="cn=Admin,dc=vide,dc=net" write
  by * none

access to attr=SIPIIdentityPassword
  by dnattr="owner" write
  by self write
  by anonymous auth
  by dn="cn=Admin,dc=vide,dc=net" write
  by * none
```

#### **I.1.4 Potential targets of commOwner**

Most ITU-T H.350 attributes are by design auxiliary classes. This enables a commObject to be 'owned' by many potential entries in a directory. For example, the most common scenario is that in which a person has an endpoint. In this scenario, a user's inetOrgPerson entry has a commURI value which points that the commObject endpoint that is associated with that person. Respectively, the endpoint represented by the target commObject has a value for commOwner that points back to the person's inetOrgPerson entry. This two-way 'ownership' relationship connects the person (represented by a 'person' object class) to its endpoint. Note that this linkage is usual, but not mandatory.

While the most common scenario is that a user is associated with an endpoint, ITU-T H.350 is by no means limited to this. For example, it is possible that a conference room is represented in a directory, and has a commURI/commOwner pair connecting that conference room with its endpoint. In this case, the linkage is not with a 'person' object class, but a 'conference room' object class. Further, it is possible to consider a zoological application in which video-enabled endpoints were placed in various cages. In this scenario, each cage would be represented in the directory with a 'cage' object class and have a commURI/commOwner pair connecting the cage to its endpoint.

This flexibility enables many possible implementation scenarios, but also requires attention during implementation, especially of white pages where lookups are generally performed on the owner of an endpoint, rather than on endpoint attributes directly. In particular, the problem is to determine the type of object class for which to search.

##### **I.1.4.1 Simple white page scenario**

In a simple single domain scenario, a white page application is configured to search an enterprise directory to find people, conference rooms, or other resources, and return their ITU-T H.350 information. The most common way that people are represented in a directory is with

inetOrgPerson. However, many institutions have derived institution-specific people object classes. In those cases, a white page application searching for inetOrgPerson attributes will not return anything useful. Therefore, the white page application should be configurable as to what attribute type to be searched.

#### **I.1.4.2 Directory of directories scenario**

A directory of directories is one in which a single search engine queries directories in many different domains. For example, a national government may maintain a white page application that searches many provincial directories. This is an extension of the simple white page scenario. However, each provincial directory may use a different object class to represent people. The white page application must maintain a table of each directory that it searches, authentication credentials if required for that directory, and the attributes to be searched for in that directory. This means that a registration process is required for target directories. Furthermore, the white page application should publish the attribute type that it expects when it receives queries.

# Appendix II

## Call flows

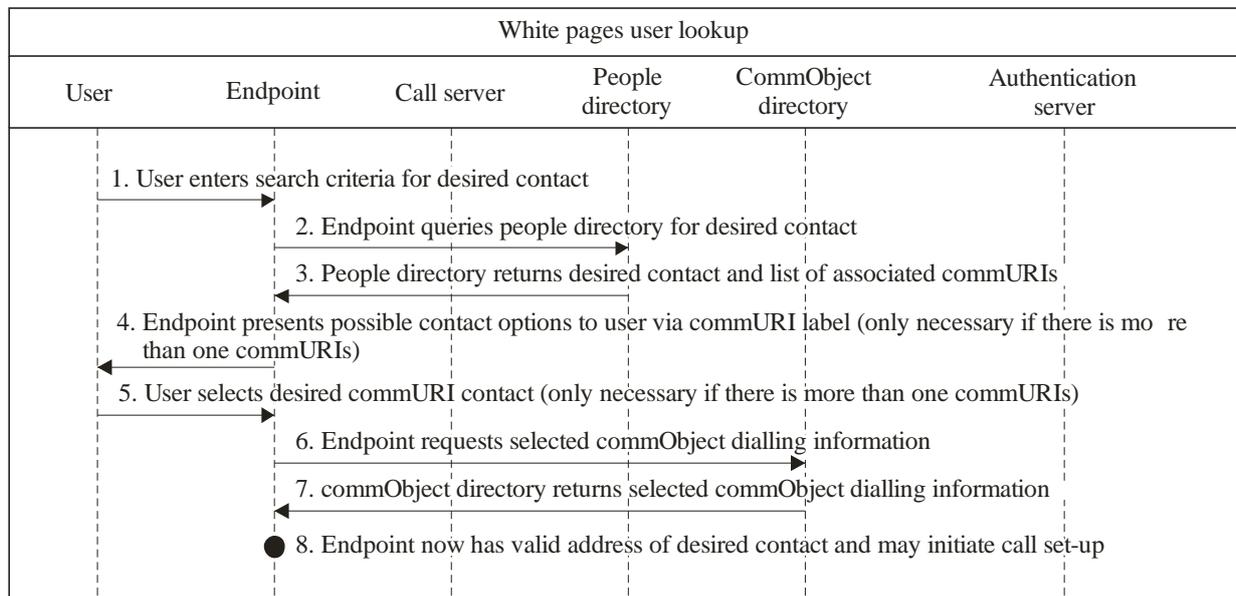
(This appendix does not form an integral part of this Recommendation.)

### II.1 Call flow scenarios

The following call flows represent possible application scenarios demonstrating how ITU-T H.350 can be used. These call flows are illustrative only and are non-normative. Specifically, ITU-T H.350 defines how data elements are represented in LDAP directories, and not how those data elements are used.

#### II.1.1 White pages user lookup

Figure II.1 shows how an LDAP-enabled endpoint can search a directory to find a user, get that user's commObject information, and dial the user's endpoint. In this scenario, the endpoint could be preconfigured to search a particular enterprise directory, or it could be preconfigured to search a directory portal that itself searches many directories. This scenario demonstrates how an appliance can handle the search, presenting choices and results through a user interface. However, this functionality can also be implemented through a web page, where the search criteria are entered into a web form and results are returned and displayed via a web page, thus enabling clickable dialling. Note that while this example illustrates an endpoint doing direct LDAP lookup, it is also possible that a call server could perform lookups on behalf of the endpoint and pass the information to the endpoint through an alternate communication path, thus centralizing some aspects of white page access.

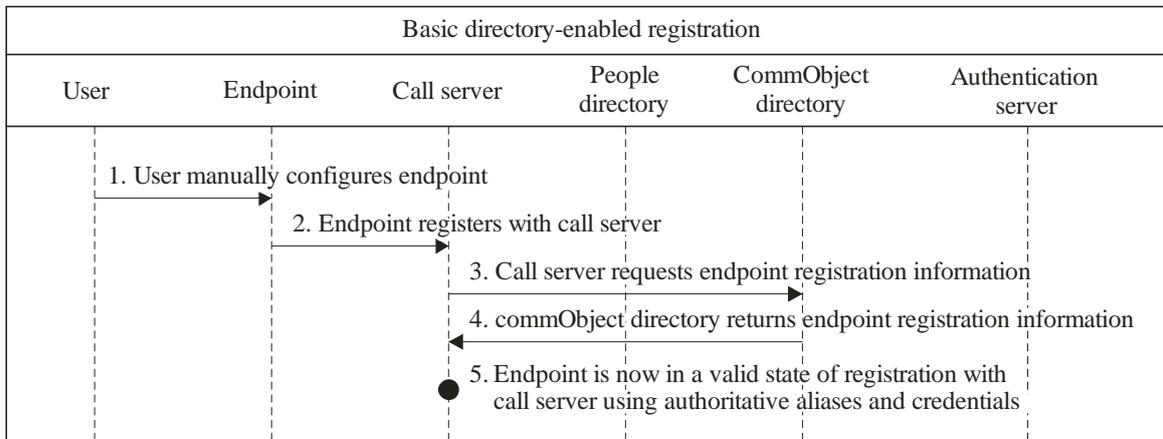


H.350(11)\_FII-1

Figure II.1 – White pages user lookup

## II.1.2 Basic directory-enabled registration

Figure II.2 shows how a call server can access a commObject directory to retrieve endpoint information, thus eliminating the need for the call server to have a proprietary internal endpoint database. The call server is always accessing authoritative and up to date information.

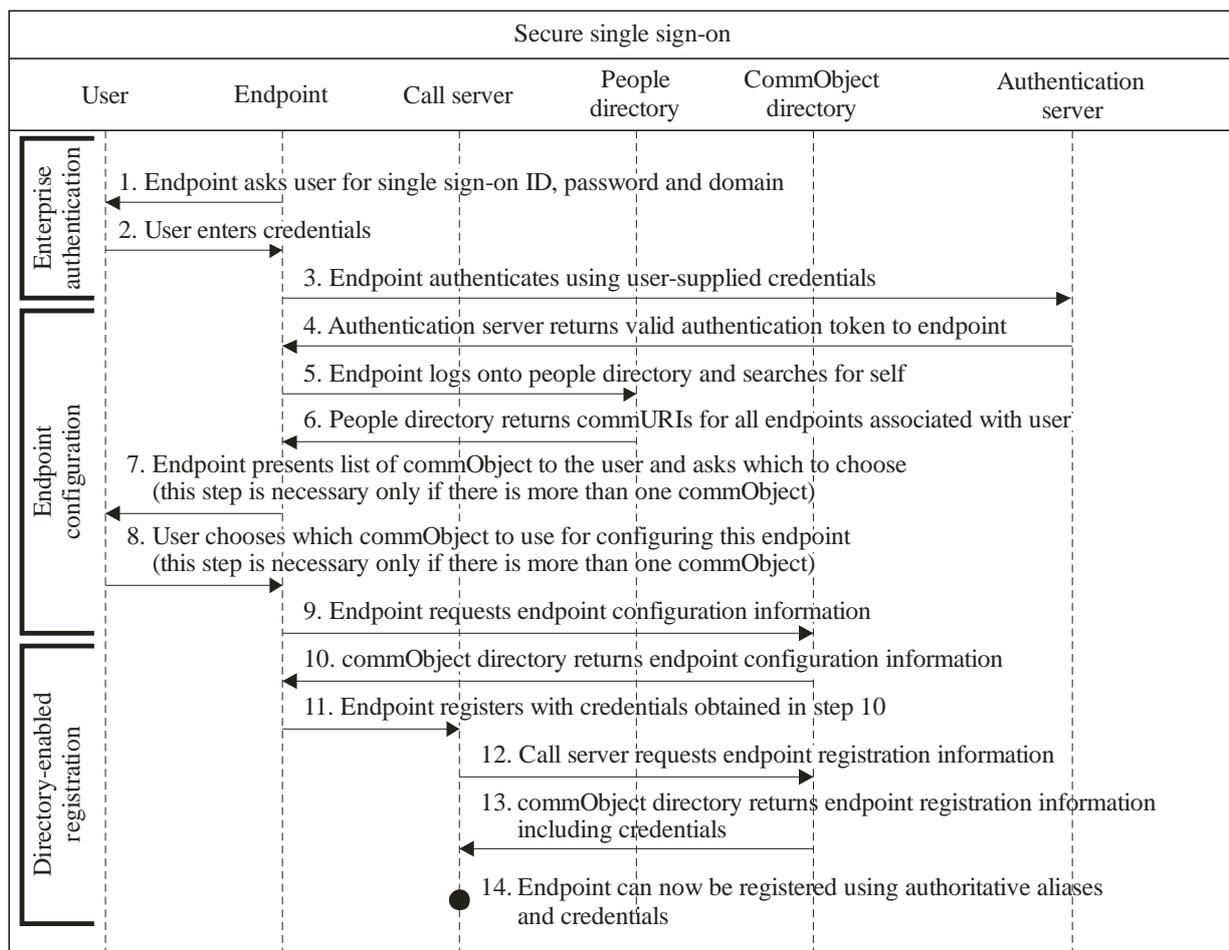


H.350(11)\_FII-2

**Figure II.2 – Basic directory enabled registration**

## II.1.3 Secure single sign-on

Figure II.3 illustrates how an environment can be created in which a user uses an existing enterprise authentication identity to access any of several endpoint identities. This scenario has several key features that are desirable for large-scale deployments. The endpoint is automatically configured using information from the directory. This eliminates user error in the configuration process and simplifies deployment. Second, users are only required to remember their single sign-on credentials, which is typically their enterprise user ID and password. Endpoints can (and often should) have different credentials, but the user does not need to know them, because they are loaded directly into the endpoint from the directory. Because the endpoint credentials are loaded automatically, it is possible that these credentials could be frequently refreshed. For example, a management tool could generate random credentials for each endpoint and store them in the commObject directory each night. This creates a highly secure environment in which credentials can be very strong, and even if compromised are aged out and recreated in a short period of time. This scenario supports the use of ID/password or certificate-based endpoint credentials. Certificates have traditionally been found to be difficult to deploy because they are difficult for users to manage. This scenario solves the certificate management problem, and opens the possibility that certificates can be managed by and on behalf of a central certificate management system, rather than on behalf of users, thus shielding users from the complexity of PKI while gaining its security advantages.



H.350(11)\_FI-3

**Figure II.3 – Secure single sign-on**

In this scenario, the enterprise authentication steps represent a user authenticating to an existing authentication server already deployed for general purposes (e.g., e-mail, web, file sharing) single sign-on authentication system. Once authenticated, the user can bind to the LDAP server directly and retrieve all configuration information for the selected endpoint, which includes configuration data and authentication credentials for the endpoint. Finally, using these credentials, the endpoint can authenticate to the call server using whichever authentication scheme is in place (for example, [b-ITU-T H.235.1] or [b-ITU-T H.235.2]). Transport layer security should be used to ensure privacy of these transactions.

## Appendix III

### Electronic attachments

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

The associated ZIP file for Recommendation ITU-T H.350 contains the following files:

- file `commURI.ldif.txt`, which contains a text-only version of the LDIF file described in clause 8.1.
- file `commObject.ldif.txt`, which contains a text-only version of the LDIF file described in clause 8.2.

The zip file is available for free download at: <http://www.itu.int/rec/T-REC-H.350> .

## Bibliography

- [b-ITU-T H.225.0] Recommendation ITU-T H.225.0 (2009), *Call signalling protocols and media stream packetization for packet-based multimedia communication systems*.
- [b-ITU-T H.235.x] Recommendations ITU-T H.235.x-series (in force), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*.
- [b-ITU-T H.235.1] Recommendation ITU-T H.235.1 (2005), *H.323 security: Baseline security profile*.
- [b-ITU-T H.235.2] Recommendation ITU-T H.235.2 (2005), *H.323 security: Signature security profile*.
- [b-ITU-T H.320] Recommendation ITU-T H.320 (2004), *Narrow-band visual telephone systems and terminal equipment*.
- [b-ITU-T H.323] Recommendation ITU-T H.323 (2009), *Packet-based multimedia communications systems*.
- [b-IETF RFC 4519] IETF RFC 4519 (2006), *Lightweight Directory Access Protocol (LDAP): Schema for User Applications*.
- [b-Howes-1] Howes, T., PhD, Smith, M., and Good, G. (1998), *Understanding and Deploying LDAP Directory Services*, New Riders Publishing, ISBN: 1578700701.
- [b-Howes-2] Howes, T., PhD, and Smith, M. (1997), *LDAP Programming Directory-Enabled Applications with Lightweight Directory Access Protocol*, New Riders Publishing, ISBN: 1578700000.





## **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
<b>Series H</b>	<b>Audiovisual and multimedia systems</b>
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