INTERNATIONAL STANDARD

# ISO/IEC 9594-5

Third edition
1998-12-15

# Information technology — Open Systems Interconnection — The Directory: Protocol specifications

*Technologies de l'information — Interconnexion de systèmes ouverts (OSI) — L'Annuaire: Spécifications du protocole*



Reference number
ISO/IEC 9594-5:1998(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Published by ISO in 2000

Printed in Switzerland

 

## Contents

# Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC 9594 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 9594-5 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*, in collaboration with ITU-T. The identical text is published as ITU-T Recommendation X.519.

This third edition cancels and replaces the second edition (ISO/IEC 9594-5:1995), of which it constitutes a minor revision.

ISO/IEC 9594 consists of the following parts, under the general title *Information technology — Open Systems Interconnection — The Directory*:

— *Part 1: Overview of concepts, models and services*

— *Part 2: Models*

— *Part 3: Abstract service definition*

— *Part 4: Procedures for distributed operation*

— *Part 5: Protocol specifications*

— *Part 6: Selected attribute types*

— *Part 7: Selected object classes*

— *Part 8: Authentication framework*

— *Part 9: Replication*

— *Part 10: Use of systems management for administration of the Directory*

Annexes A to G form a normative part of this part of ISO/IEC 9594. Annex H is for information only.

## Introduction

This Recommendation | International Standard, together with the other Recommendations | International Standards, has been produced to facilitate the interconnection of information processing systems to provide directory services. A set of such systems, together with the directory information which they hold, can be viewed as an integrated whole, called the *Directory*. The information held by the Directory, collectively known as the Directory Information Base (DIB), is typically used to facilitate communication between, with or about objects such as application entities, people, terminals and distribution lists.

The Directory plays a significant role in Open Systems Interconnection, whose aim is to allow, with a minimum of technical agreement outside of the interconnection standards themselves, the interconnection of information processing systems:

- from different manufacturers;
- under different managements;
- of different levels of complexity; and
- of different ages.

This Recommendation | International Standard specifies the application service elements and application contexts for two protocols - the Directory Access Protocol (DAP) and the Directory System Protocol (DSP). The DAP provides for access to the Directory to retrieve or modify Directory information. The DSP provides for the chaining of requests to retrieve or modify Directory information to other parts of the distributed Directory System where the information may be held.

In addition this Recommendation | International Standard specifies the application service elements and application contexts for the Directory Information Shadowing Protocol (DISP) and the Directory Operational Binding Management Protocol (DOP). The DISP provides for the shadowing of information held in one DSA to another DSA. The DOP provides for the establishment, modification and termination of bindings between pairs of DSAs for the administration of relationships between the DSAs (such as for shadowing or hierarchical relationships).

This third edition technically revises and enhances, but does not replace, the second edition of this Recommendation | International Standard. Implementations may still claim conformance to the second edition. However, at some point, the second edition will not be supported (i.e. reported defects will no longer be resolved). It is recommended that implementations conform to this third edition as soon as possible.

This third edition specifies version 1 and version 2 of the Directory protocols.

The first and second editions also specified version 1. Most of the services and protocols specified in this edition are designed to function under version 1. When version 1 has been negotiated differences between the services and between the protocols defined in the three editions are accommodated using the rules of extensibility defined in this edition of ITU-T Rec. X.519 | ISO/IEC 9594-5. However, some enhanced services and protocols, e.g. signed errors, will not function unless all Directory entities involved in the operation have negotiated version 2.

Implementors should note that a defect resolution process exists and that corrections may be applied to this part of this International Standard in the form of technical corrigenda. The identical corrections will be applied to this Recommendation in the form of Corrigenda and/or an Implementor's Guide. A list of approved technical corrigenda for this part of this International Standard can be obtained from the subcommittee secretariat. Published technical corrigenda are available from your national standards organization. The ITU-T Corrigenda and Implementor's Guides may be obtained from the ITU Web site.

Annex A, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module for the directory access protocol.

Annex B, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module for the directory system protocol.

Annex C, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module for the directory information shadowing protocol.

Annex D, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module for the directory operational binding management protocol.

Annex E, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module which contains all the ASN.1 object identifiers assigned in this Recommendation | International Standard.

Annex F, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module which contains all the ASN.1 object identifiers assigned to identify operational binding types in this series of Recommendations | International Standards.

Annex G, which is an integral part of this Recommendation | International Standard, provides the ASN.1 module for the security exchanges.

Annex H, which is not an integral part of this Recommendation | International Standard, lists the amendments and defect reports that have been incorporated to form this edition of this Recommendation | International Standard.

**INTERNATIONAL STANDARD**

**ITU-T RECOMMENDATION**

# INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – THE DIRECTORY: PROTOCOL SPECIFICATIONS

## 1 Scope

This Recommendation | International Standard specifies the Directory Access Protocol, the Directory System Protocol, the Directory Information Shadowing Protocol, and the Directory Operational Binding Management Protocol fulfilling the abstract services specified in ITU-T Rec. X.511 | ISO/IEC 9594-3, ITU-T Rec. X.518 | ISO/IEC 9594-4, and ITU-T Rec. X.525 | ISO/IEC 9594-9.

## 2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

### 2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model.*
- ITU-T Recommendation X.213 (1995) | ISO/IEC 8348:1996, *Information technology – Open Systems Interconnection – Network service definition.*
- ITU-T Recommendation X.214 (1995) | ISO/IEC 8072:1996, *Information technology – Open Systems Interconnection – Transport service definition.*
- ITU-T Recommendation X.215 (1995) | ISO/IEC 8326:1996, *Information technology – Open Systems Interconnection – Session service definition.*
- ITU-T Recommendation X.216 (1994) | ISO/IEC 8822:1994, *Information technology – Open Systems Interconnection – Presentation service definition.*
- ITU-T Recommendation X.217 (1995) | ISO/IEC 8649:1996, *Information technology – Open Systems Interconnection – Service definition for the Association Control Service Element.*
- ITU-T Recommendation X.227 (1995) | ISO/IEC 8650-1:1996, *Information technology – Open Systems Interconnection – Connection-oriented protocol for the Association Control Service Element: Protocol specification.*
- ITU-T Recommendation X.500 (1997) | ISO/IEC 9594-1:1998, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services.*
- ITU-T Recommendation X.501 (1997) | ISO/IEC 9594-2:1998, *Information technology – Open Systems Interconnection – The Directory: Models.*
- ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8:1998, *Information technology – Open Systems Interconnection – The Directory: Authentication framework.*
- ITU-T Recommendation X.511 (1997) | ISO/IEC 9594-3:1998, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition.*

- ITU-T Recommendation X.518 (1997) | ISO/IEC 9594-4:1998, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation.*
- ITU-T Recommendation X.520 (1997) | ISO/IEC 9594-6:1998, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types.*
- ITU-T Recommendation X.521 (1997) | ISO/IEC 9594-7:1998, *Information technology – Open Systems Interconnection – The Directory: Selected object classes.*
- ITU-T Recommendation X.525 (1997) | ISO/IEC 9594-9:1998, *Information technology – Open Systems Interconnection – The Directory: Replication.*
- ITU-T Recommendation X.530 (1997) | ISO/IEC 9594-10:1998, *Information technology – Open Systems Interconnection – The Directory: Use of systems management for administration of the Directory.*
- ITU-T Recommendation X.680 (1997) | ISO/IEC 8824-1:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*
- ITU-T Recommendation X.681 (1997) | ISO/IEC 8824-2:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.*
- ITU-T Recommendation X.682 (1997) | ISO/IEC 8824-3:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.*
- ITU-T Recommendation X.683 (1997) | ISO/IEC 8824-4:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Parametrization of ASN.1 specifications.*
- ITU-T Recommendation X.690 (1997) | ISO/IEC 8825-1:1998, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*
- ITU Recommendation X.803 (1994) | ISO/IEC 10745:1995, *Information technology – Open Systems Interconnection – Upper layers security model.*
- ITU Recommendation X.830 (1995) | ISO/IEC 11586-1:1996, *Information technology – Open Systems Interconnection – Generic upper layers security: Overview, models and notation.*
- ITU Recommendation X.831 (1995) | ISO/IEC 11586-2:1996, *Information technology – Open Systems Interconnection – Generic upper layers security: Security Exchange Service Element (SESE) service definition.*
- ITU Recommendation X.832 (1995) | ISO/IEC 11586-3:1996, *Information technology – Open Systems Interconnection – Generic upper layers security: Security Exchange Service Element (SESE) protocol specification.*
- ITU Recommendation X.833 (1995) | ISO/IEC 11586-4:1996, *Information technology – Open Systems Interconnection – Generic upper layers security: Protecting transfer syntax specification.*
- ITU-T Recommendation X.880 (1994) | ISO/IEC 13712-1:1995, *Information technology – Remote Operations: Concepts, model and notation plus Technical Corrigendum 1 (1996).*
- ITU-T Recommendation X.880 (1994)/Amd.1 (1995) | ISO/IEC 13712-1:1995/Amd.1:1996, *Information technology – Remote Operations: Concepts, model and notation – Amendment 1: Built-in operations.*
- ITU-T Recommendation X.881 (1994) | ISO/IEC 13712-2:1995, *Information technology – Remote Operations: OSI realizations – Remote Operations Service Element (ROSE) service definition.*
- ITU-T Recommendation X.881 (1994)/Amd.1 (1995) | ISO/IEC 13712-2:1995/Amd.1:1996, *Information technology – Remote Operations: OSI realizations – Remote Operations Service Element (ROSE) service definition – Amendment 1: Mapping to A-UNIT-DATA and built-in operations.*
- ITU-T Recommendation X.882 (1994) | ISO/IEC 13712-3:1995, *Information technology – Remote Operations: OSI realizations – Remote Operations Service Element (ROSE) protocol specification plus Technical Corrigendum 1 (1996).*
- ITU-T Recommendation X.882 (1994)/Amd.1 (1995) | ISO/IEC 13712-3:1995/Amd.1:1996, *Information technology – Remote Operations: OSI realizations – Remote Operations Service Element (ROSE) protocol specification – Amendment 1: Mapping to A-UNIT-DATA and built-in operations.*

## 2.2 Paired Recommendations | International Standards equivalent in technical content

- ITU-T Recommendation X.218 (1993) *Reliable Transfer: Model and Service Definition.*

  ISO/IEC 9066-1:1989, *Information processing systems – Text communication – Reliable Transfer – Part 1: Model and service definition.*

### 2.3 Other references

- ITU-T Recommendation E.164 (1997), *The international public telecommunication numbering plan*.
- ITU-T Recommendation X.121 (1996), *International numbering plan for public data networks*.
- RFC 2025 (1996), *The Simple Public-Key GSS-API Mechanism (SPKM)*.

## 3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply:

### 3.1 OSI Reference Model Definitions

The following terms are defined in ITU-T Rec. X.200 | ISO/IEC 7498-1:

a) *abstract-syntax;*
b) *application-context*;
c) *application-entity*;
d) *application process*;
e) *application-protocol-control-information*;
f) *application-protocol-data-unit*;
g) *application-service-element*.

### 3.2 Remote Operations Definitions

The following terms are defined in ITU-T Rec. X.880 | ISO/IEC 13712-1:

a) *connection package*;
b) *contract, association contract*;
c) *error*;
d) *operation*;
e) *operation package*;
f) *ROS-object*.

### 3.3 Basic Directory Definitions

The following terms are defined in ITU-T Rec. X.501 | ISO/IEC 9594-2:

a) *the Directory;*
b) *(Directory) user*;
c) *Directory System Agent (DSA)*;
d) *Directory User Agent (DUA).*

### 3.4 Distributed Operation Definitions

The following terms are defined in ITU-T Rec. X.518 | ISO/IEC 9594-4:

a) *chaining*;
b) *referral*.

### 3.5 Upper layer security definitions

The following terms are used as defined in ITU-T Rec. X.803 | ISO/IEC 10745:

a) *security association;*
b) *security transformation;*
c) *security exchange;*
d) *security exchange item.*

The following terms are used as defined in ITU-T Rec. X.830 | ISO/IEC 11586-1:

a) *protecting transfer syntax;*

b) *protection mapping.*

## 4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

| | |
|---|---|
| AC | Application Context[COMP/RM1] |
| ACSE | Association Control Service Element[COMP/RM2] |
| AE | Application Entity |
| APCI | Application Protocol Control Information |
| APDU | Application Protocol Data Unit |
| ASE | Application Service Element |
| DAP | Directory Access Protocol |
| DISP | Directory Information Shadowing Protocol |
| DOP | Directory Operational Binding Management Protocol |
| DSA | Directory System Agent |
| DSP | Directory System Protocol |
| DUA | Directory User Agent |
| GULS | Generic Upper Layers Security |
| ROS | Remote Operations Service |
| ROSE | Remote Operations Service Element |
| RTSE | Reliable Transfer Service Element |
| SESE | Security Exchange Service Element |

## 5 Conventions

With minor exceptions this Directory Specification has been prepared according to the "Presentation of ITU-T | ISO/IEC common text" guidelines in the Guide for ITU-T and ISO/IEC JTC 1 Cooperation.

The term "Directory Specification" (as in "this Directory Specification") shall be taken to mean ITU-T Rec. X.519 | ISO/IEC 9594-5. The term "Directory Specifications" shall be taken to mean the X.500-series Recommendations and all parts of ISO/IEC 9594.

This Directory Specification uses the term "1988 edition systems" to refer to systems conforming to the first (1988) edition of the Directory Specifications, i.e. the 1988 edition of the series of CCITT X.500 Recommendations and the ISO/IEC 9594:1990 edition. This Directory Specification uses the term "1993 edition systems" to refer to systems conforming to the second (1993) edition of the Directory Specifications, i.e. the 1993 edition of the series of ITU-T X.500 Recommendations and the ISO/IEC 9594:1995 edition. Systems conforming to this third edition of the Directory Specifications are referred to as "1997 edition systems".

This Directory Specification presents ASN.1 notation in the bold Helvetica typeface. When ASN.1 types and values are referenced in normal text, they are differentiated from normal text by presenting them in the bold Helvetica typeface. The names of procedures, typically referenced when specifying the semantics of processing, are differentiated from normal text by displaying them in bold Times. Access control permissions are presented in italicized Times.

If the items in a list are numbered (as opposed to using "–" or letters), then the items shall be considered steps in a procedure.

This Directory Specification defines directory operations using the Remote Operation notation defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

# 6 Protocol overview

## 6.1 Remote Operations – Specification and OSI Realization

ITU-T Rec. X.880 | ISO/IEC 13712-1 defines several information object classes that are useful in the specification of ROS-based application protocols such as the various Directory protocols defined in this Directory Specification. A number of these classes are used in this and subsequent clauses. The specification techniques provided in ITU-T Rec. X.880 | ISO/IEC 13712-1 are used to define a generic protocol between objects. When realized as an OSI application layer protocol, the concepts of ITU-T Rec. X.880 | ISO/IEC 13712-1 are mapped to OSI concepts in ITU-T Rec. X.881 | ISO/IEC 13712-2 and ITU-T Rec. X.882 | ISO/IEC 13712-3.

The **ROS-OBJECT-CLASS** class is used to define a set of common capabilities of a set of ROS-objects in terms of the (association) contracts they support as initiators and/or responders. When realized using the communication services of OSI, a ROS-object maps to an application process and a contract to an application context. In these Directory Specifications the term abstract service is used to refer to a ROS association contract and OSI application layer protocol to refer to the realization of a contract between two open systems using OSI communication services.

The **OPERATION-PACKAGE** class is used to define a set of operations which may be invoked by a ROS-object assuming the role of "consumer", the operations which may be invoked by a ROS-object assuming the role of "supplier", and the operations which may be invoked by both ROS-objects. When using the communication services of OSI, an operation package is realized as an application service element (ASE).

The **CONNECTION-PACKAGE** class is used to define the bind and unbind operations used to establish and release an association. When realized using the communication services of OSI, a connection package is realized as the procedures that use the services of the Association Control Service Element.

The **CONTRACT** class is used to define an association contract in terms of a connection package and one or more operation packages. When specifying the contract, the packages in which the association initiator assumes the role of consumer, the association responder assumes the role of consumer, and either may assume the role of consumer are identified. When using the communication services of OSI, a contract is realized as an application context.

The **APPLICATION-CONTEXT** class is used to define the static aspects of an application context. These include the contract that is realized via the application context, the OSI service that establishes and releases the association, the OSI service that provides information transfer for the interactions of the contract, and the abstract syntaxes used.

The **ABSTRACT-SYNTAX** class, which is built in to ASN.1, is used to define and assign an object identifier to an ASN.1 type whose values comprise an abstract syntax.

The OSI application layer protocols defined in the Directory Specifications, the DAP, DSP, DISP and DOP, are protocols to provide communication between a pair of application processes. In the OSI environment this is represented as communication between a pair of Application-Entities (AEs) using the presentation service. The function of an AE is provided by a set of Application-Service-Elements (ASEs). The interaction between AEs is described in terms of their use of the services provided by the ASEs. All the services provided by the Directory ASEs are contained in a single AE.

The Remote Operations Service Element (ROSE) supports the request/reply paradigm of the operation. The Directory ASEs provide the mapping function of the abstract-syntax notation of the directory operation packages onto the services provided by the ROSE.

The Association Control Service Element (ACSE) supports the establishment and release of an application-association between a pair of AEs. Associations between a DUA and a DSA may be established only by the DUA. Only the initiator of an established association can release it.

Optionally, the Reliable Transfer Service Element (RTSE) may be used to reliably transfer the Application Protocol Data Units (APDUs) of the DISP.

Optionally, the Security Exchange Service Element (SESE) may be used to convey credentials for authentication and to establish a security association.

## 6.2 Directory ROS-Objects and Contracts

ITU-T Rec. X.511 | ISO/IEC 9594-3 defines the abstract service between a DUA and the Directory which provides an access point to support a user accessing Directory services.

The **dua** class of ROS-object describes a DUA, being an instance of this class, as the initiator of the contract **dapContract**. This contract is referred to in these Directory Specifications as the Directory Abstract Service. It is specified as a ROS-based information object in 6.3.

```
dua ROS-OBJECT-CLASS ::= {
        INITIATES       { dapContract }
        ID              id-rosObject-dua }
```

The **directory** class of ROS-object describes the provider of the Directory Abstract Service. This provider is the responder of the **dapContract**.

```
directory ROS-OBJECT-CLASS ::= {
        RESPONDS        { dapContract }
        ID              id-rosObject-directory }
```

The Directory is further modelled, as depicted in Figure 1, as being represented to a DUA by a DSA which supports the particular access point concerned. ITU-T Rec. X.518 | ISO/IEC 9594-4 defines the interactions between a pair of DSAs within the Directory to support user requests which are chained.

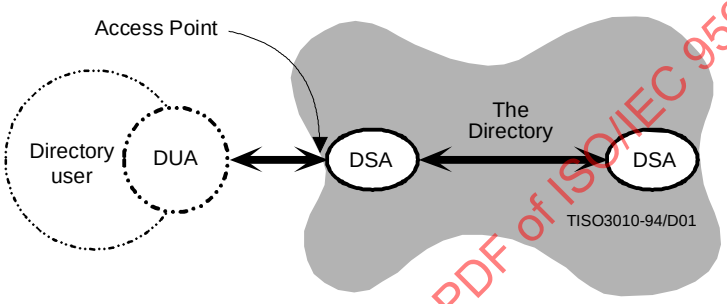


**Figure 1 – Directory Interactions**



The **directory** object is therefore manifested as a set interacting DSAs. Each DSA comprising the **directory** is an instance of the **dap-dsa** class. A **dap-dsa** object assumes the role of responder in the **dapContract**.

```
dap-dsa ROS-OBJECT-CLASS ::= {
        RESPONDS        { dapContract }
        ID              id-rosObject-dapDSA }
```

In addition to interacting with DUAs, DSAs interact with one another to achieve various objectives. In what follows, a number of contracts and ROS-objects expressing how DSAs participate in these contracts are defined. Any real DSA may instantiate one or more of these DSA ROS-objects.

The interactions between DSAs generally required to provide the Directory Abstract Service in the presence of a distributed DIB are defined as a **dspContract**. A DSA that participates in this contract is defined as a ROS-object of class **dsp-dsa**. The contract is referred to in these Directory Specifications as the DSA Abstract Service. It is specified as a ROS-based information object in 6.4.

```
dsp-dsa ROS-OBJECT-CLASS ::= {
        BOTH            { dspContract }
        ID              id-rosObject-dspDSA }
```

The Shadow Abstract Service specifies the shadowing of information between a shadow supplier and a shadow consumer DSA. This service is manifested in two forms and therefore is defined as two distinct contracts. They are specified as a ROS-based information objects in 6.5.

The **shadowConsumerContract** expresses the form of the service in which the shadow consumer, a ROS-object of class **initiating-consumer-dsa**, initiates the contract. A ROS-object of class **responding-supplier-dsa**, responds in this contract.

```
initiating-consumer-dsa ROS-OBJECT-CLASS ::= {
        INITIATES       { shadowConsumerContract }
        ID              id-rosObject-initiatingConsumerDSA }
```

```
responding-supplier-dsa  ROS-OBJECT-CLASS  ::= {
        RESPONDS        { shadowConsumerContract }
        ID              id-rosObject-respondingSupplierDSA }
```

The **shadowSupplierContract** expresses the form of the service in which the shadow supplier, a ROS-object of class **initiating-supplier-dsa**, initiates the contract. A ROS-object of class **responding-consumer-dsa**, responds in this contract.

```
initiating-supplier-dsa  ROS-OBJECT-CLASS  ::= {
        INITIATES       { shadowSupplierContract }
        ID              id-rosObject-initiatingSupplierDSA }

responding-consumer-dsa  ROS-OBJECT-CLASS  ::= {
        RESPONDS        { shadowSupplierContract }
        ID              id-rosObject-respondingConsumerDSA }
```

The interactions between two DSAs to manage a set of operational bindings are defined as a **dopContract**.

```
dop-dsa  ROS-OBJECT-CLASS  ::= {
        BOTH            { dopContract }
        ID              id-rosObject-dopDSA }
```

A DSA that participates in this contract is defined as a ROS-object of class **dop-dsa**. This contract is specified as a ROS-based information object in 6.6.

## 6.3 DAP Contract and Packages

The **dapContract** is defined as an information object of class **CONTRACT**.

```
dapContract  CONTRACT  ::= {
        CONNECTION                  dapConnectionPackage
        INITIATOR CONSUMER OF       { readPackage | searchPackage | modifyPackage }
        ID                          id-contract-dap }
```

When a DUA and DSA from different open systems interact, this association contract may be realized as an OSI application layer protocol, referred to in these Directory Specifications as the Directory Access Protocol (DAP). The definition of this protocol in terms of an OSI application context is provided in 7.2 of this Directory Specification.

The **dapContract** is composed of a connection package, **dapConnectionPackage**, and three operation packages, **readPackage**, **searchPackage** and **modifyPackage**.

The connection package, **dapConnectionPackage**, is defined as an information object of class **CONNECTION-PACKAGE**. The bind and unbind operations of this connection package, **directoryBind** and **directoryUnbind**, are defined in ITU-T Rec. X.511 | ISO/IEC 9594-3.

```
dapConnectionPackage  CONNECTION-PACKAGE  ::= {
        BIND        directoryBind
        UNBIND      directoryUnbind
        ID          id-package-dapConnection }
```

The operation packages, **readPackage**, **searchPackage** and **modifyPackage**, are defined as information objects of class **OPERATION-PACKAGE**. The operations of these operation packages are defined in ITU-T Rec. X.511 | ISO/IEC 9594-3.

```
readPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES        { read | compare | abandon }
        ID                      id-package-read }

searchPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES        { list | search }
        ID                      id-package-search }

modifyPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES        { addEntry | removeEntry | modifyEntry | modifyDN }
        ID                      id-package-modify }
```

NOTE – These packages, when realized as ASEs, are used for the construction of application contexts defined in this Specification They are not intended to allow for claims of conformance to individual, or other combinations of, ASEs.

Since the DUA is the initiator of the **dapContract**, it assumes the role of consumer of the operation packages of the contract. This means that only the DUA can invoke operations in this contract and its OSI realization.

## 6.4 DSP Contract and Packages

The **dspContract** is defined as an information object of class **CONTRACT**.

```
dspContract  CONTRACT  ::= {
        CONNECTION          dspConnectionPackage
        OPERATIONS OF       { chainedReadPackage | chainedSearchPackage | chainedModifyPackage }
        ID                  id-contract-dsp }
```

When a pair of DSAs from different open systems interact, this association contract is realized as an OSI application layer protocol, referred to in these Directory Specifications as the Directory System Protocol (DSP). The definition of this protocol in terms of an OSI application context is provided in 7.2.

The **dspContract** is composed of a connection package, **dspConnectionPackage**, and three operation packages, **chainedReadPackage**, **chainedSearchPackage** and **chainedModifyPackage**.

The connection package, **dspConnectionPackage**, is defined as an information object of class **CONNECTION-PACKAGE**. It is identical to the connection package, **dapConnectionPackage**.

```
dspConnectionPackage  CONNECTION-PACKAGE  ::= {
        BIND        dSABind
        UNBIND      dSAUnbind
        ID          id-package-dspConnection }
```

The operation packages, **chainedReadPackage**, **chainedSearchPackage** and **chainedModifyPackage**, are defined as information objects of class **OPERATION-PACKAGE**. The operations of these operation packages are defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

```
chainedReadPackage  OPERATION-PACKAGE  ::= {
        OPERATIONS      { chainedRead | chainedCompare | chainedAbandon }
        ID              id-package-chainedRead }

chainedSearchPackage  OPERATION-PACKAGE  ::= {
        OPERATIONS      { chainedList | chainedSearch }
        ID              id-package-chainedSearch }

chainedModifyPackage  OPERATION-PACKAGE  ::= {
        OPERATIONS      { chainedAddEntry | chainedRemoveEntry |
                        chainedModifyEntry | chainedModifyDN }
        ID              id-package-chainedModify }
```

NOTE – These packages, when realized as ASEs, are used for the construction of application contexts defined in this Specification They are not intended to allow for claims of conformance to individual, or other combinations of, ASEs.

In the **dspContract** either DSA may assume the role of initiator and either the initiating or responding DSA may invoke the operations of the contract.

## 6.5 DISP Contracts and Packages

The **shadowConsumerContract** and **shadowSupplierContract** are defined as information objects of class **CONTRACT**.

```
shadowConsumerContract  CONTRACT  ::= {
        CONNECTION                  dispConnectionPackage
        INITIATOR CONSUMER OF       { shadowConsumerPackage }
        ID                          id-contract-shadowConsumer }

shadowSupplierContract  CONTRACT  ::= {
        CONNECTION                  dispConnectionPackage
        RESPONDER CONSUMER OF       { shadowSupplierPackage }
        ID                          id-contract-shadowSupplier }
```

NOTE – The term consumer and supplier are employed in the notation for the **CONTRACT** and **OPERATION-PACKAGE** classes are used to designate two roles. These roles correspond to the two terms shadow consumer and shadow supplier, respectively, used in ITU-T Rec. X.525 | ISO/IEC 9594-9.

The OSI realizations of the two forms of the Shadow Abstract Service, referred to collectively as the Directory Information Shadowing Protocol (DISP), are defined in terms of several OSI application contexts, provided in 7.2.

The **shadowConsumerContract** and **shadowSupplierContract** are composed of a common connection package, **dispConnectionPackage**, and one operation package, either **shadowConsumerPackage** in the first case or **shadowSupplierPackage** in the second.

The connection package, **dispConnectionPackage**, is defined as an information object of class **CONNECTION-PACKAGE**. It is identical to the connection package, **dapConnectionPackage**.

```
dispConnectionPackage  CONNECTION-PACKAGE  ::= {
        BIND                    dSAShadowBind
        UNBIND                  dSAShadowUnbind
        ID                      id-package-dispConnection }
```

The operation packages, **shadowConsumerPackage** and **shadowSupplierPackage**, are defined as information objects of class **OPERATION-PACKAGE**. The operations of these operation packages are defined in ITU-T Rec. X.525 | ISO/IEC 9594-9.

```
shadowConsumerPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES  { requestShadowUpdate }
        SUPPLIER INVOKES  { updateShadow }
        ID                id-package-shadowConsumer }

shadowSupplierPackage  OPERATION-PACKAGE  ::= {
        SUPPLIER INVOKES  { coordinateShadowUpdate |
                            updateShadow }
        ID                id-package-shadowSupplier }
```

Since the shadow consumer is the initiator of the **shadowConsumerContract**, it assumes the role of consumer of the **shadowConsumerPackage**. This means that the shadow consumer invokes the **requestShadowUpdate** operation and that the shadow supplier invokes the **updateShadow** operation.

Since the shadow supplier is the initiator of the **shadowSupplierContract**, it assumes the role of supplier of the **shadowSupplierPackage**. This means that the shadow supplier invokes the operations of the contract.

## 6.6 DOP Contract and Packages

The **dopContract** is defined as an information object of class **CONTRACT**.

```
dopContract  CONTRACT  ::= {
        CONNECTION              dopConnectionPackage
        OPERATIONS OF           { dopPackage }
        ID                      id-contract-dop }
```

When a pair of DSAs from different open systems interact, this association contract is realized as an OSI application layer protocol, referred to in these Directory Specifications as the Directory Operational Binding Management Protocol (DOP). The definition of this protocol in terms of an OSI application context is provided in 7.2.

The connection package, **dopConnectionPackage**, is defined as an information object of class **CONNECTION-PACKAGE**. It is identical to the connection package, **dapConnectionPackage**.

```
dopConnectionPackage  CONNECTION-PACKAGE  ::= {
        BIND                    dSAOperationalBindingManagementBind
        UNBIND                  dSAOperationalBindingManagementUnbind
        ID                      id-package-dopConnection }
```

The operation package, **dopPackage**, is defined as information objects of class **OPERATION-PACKAGE**. The operations of these operation packages are defined in ITU-T Rec. X.501 | ISO/IEC 9594-2.

```
dopPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES  { establishOperationalBinding |
                            modifyOperationalBinding |
                            terminateOperationalBinding }
        ID                id-package-operationalBindingManagement }
```

The DSA that may assume the role of initiator of the **dopContract**, depends on the DSA roles assigned for the operational binding(s) to be managed using the operations of this contract. Only the initiator may invoke the operations of the **dopContract**. More than one operational binding type may be managed with this contract only if the DSA roles for the distinct types are compatible (e.g. a DSA assumes Role A for each binding type).

## 6.7 Use of underlying services

The DAP, DSP, DOP and DISP protocols make use of underlying services as described below.

### 6.7.1 Use of ROSE services

The Remote Operations Service Element (ROSE) is defined ITU-T Rec. X.881 | ISO/IEC 13712-2.

The ROSE supports the request/reply paradigm of remote operations.

The Directory ASEs are users of the RO-INVOKE, RO-RESULT, RO-ERROR, RO-REJECT-U and RO-REJECT-P services of the ROSE.

The remote operations of the DAP and the DSP are asynchronous. Note that, as the DUA is a consumer of the DAP, it may choose to operate in a synchronous manner.

The remote operations of the DISP shall be supported as synchronous operations and may optionally be supported as asynchronous operations.

The remote operations of the DOP are asynchronous.

### 6.7.2 Use of RTSE services

The Reliable Transfer Service Element (RTSE) is defined in ITU-T Rec. X.218 | ISO/IEC 9066-1.

The RTSE provides for the reliable transfer of Application Protocol Data Units (APDUs). The RTSE ensures that each APDU is completely transferred exactly once, or that the sender is warned of an exception. The RTSE recovers from communication and end-system failure and minimizes the amount of retransmission needed for recovery.

Alternative application contexts with and without RTSE are defined to support the DISP.

The RTSE is used in normal mode. The use of the normal mode of the RTSE implies the use of the normal mode of the ACSE and the normal mode of the Presentation Service.

If the RTSE is included in an application context, the RO-BIND service maps onto the RT-OPEN service of the RTSE and the RO-UNBIND service maps onto the RT-CLOSE service of the RTSE. The basic ROSE services are the sole user of the RT-TRANSFER, RT-TURN-PLEASE, RT-TURN-GIVE, RT-P-ABORT and RT-U-ABORT services of the RTSE.

### 6.7.3 Use of ACSE services

The Association Control Service Element (ACSE) is defined in ITU-T Rec. X.217 | ISO/IEC 8649.

The ACSE provides for the control (establishment, release, abort) of application-associations between AEs.

If the RTSE but not SESE is included in an application context, the RTSE is the sole user of the A-ASSOCIATE, A-RELEASE, A-ABORT and A-P-ABORT services of the ACSE.

If the RTSE and SESE are both not included in an application context, the RO-BIND and RO-UNBIND services are the sole users of the A-ASSOCIATE and A-RELEASE services of the ACSE. The application-process is the user of the A-ABORT and A-P-ABORT services of the ACSE.

If the SESE but not RTSE is included in an application context:

a) the RO-BIND services use the ACSE A-ASSOCIATE kernel functional unit;

b) the SE-TRANSFER services carrying the first two security exchange items of the security exchange use the ACSE A-ASSOCIATE authentication functional unit;

c) the RO-UNBIND services are the sole users of A-RELEASE services of the ACSE;

d) the application-process is the user of the A-ABORT and A-P-ABORT services of the ACSE.

If the SESE and RTSE are both included in an application context:

a) the RTSE services use the ACSE A-ASSOCIATE kernel functional unit;

b) the SE-TRANSFER services carrying the first two security exchange items of the security exchange use the ACSE A-ASSOCIATE authentication functional unit.

c) RTSE is the sole users of A-RELEASE, A-ABORT and A-P-ABORT services of the ACSE.

The receipt of an A-ABORT or A-P-ABORT on an association supporting the dap terminates all request processing. Except for certain conditions described in ITU-T Rec. X.518 | ISO/IEC 9594-4, this is also true for the DSP. It is a Directory user responsibility to confirm if requested modifications to the dib occurred.

The receipt of an A-ABORT or A-P-ABORT on an association supporting the DISP is described in ITU-T Rec. X.525 | ISO/IEC 9594-9.

The receipt of an A-ABORT or A-P-ABORT on an association supporting the DOP is described in ITU-T Rec. X.518 | ISO/IEC 9594-4.

### 6.7.4 Use of the presentation service

The presentation service is defined in ITU-T Rec. X.216 | ISO/IEC 8822.

The Presentation Layer coordinates the representation (syntax) of the Application Layer semantics that are to be exchanged.

In normal mode, a different presentation-context is used for each abstract-syntax included in the application-context.

The ACSE is the sole user of the P-CONNECT, P-RELEASE, P-U-ABORT and P-P-ABORT services of the presentation service.

If the RTSE but not SESE is included in an application context, the RTSE is the sole user of the P-ACTIVITY-START, P-ACTIVITY-END, P-ACTIVITY-INTERRUPT, P-ACTIVITY-DISCARD, P-ACTIVITY-RESUME, P-DATA, P-MINOR-SYNCHRONIZE, P-U-EXCEPTION-REPORT, P-P-EXCEPTION-REPORT, P-TOKEN-PLEASE and P-CONTROL-GIVE services of the presentation service.

If the RTSE and SESE are both not included in an application context, the ROSE is the sole user of the P-DATA service of the presentation service.

If the SESE but not RTSE is included in an application context, and the security exchange involves more than a two-way exchange, then SESE is the sole user of the P-DATA service of the presentation service until after an SE-TRANSFER request or indication with the End Flag set or until an SE-U-ABORT or an SE-P-ABORT. Following this the ROSE is the sole user of the P-DATA service.

If the SESE and RTSE are both included in an application context, and the security exchange involves more than a two-way exchange, then SESE is the sole user of the P-TYPE-DATA service of the presentation service until after an SE-TRANSFER request or indication with the End Flag set or until an SE-U-ABORT or an SE-P-ABORT. Following this the RTSE is the sole user of the P-ACTIVITY-START, P-ACTIVITY-END, P-ACTIVITY-INTERRUPT, P-ACTIVITY-DISCARD, P-ACTIVITY-RESUME, P-DATA, P-MINOR-SYNCHRONIZE, P-U-EXCEPTION-REPORT, P-P-EXCEPTION-REPORT, P-TOKEN-PLEASE and P-CONTROL-GIVE services of the presentation service.

Presentation default context, context restoration, and context management are not used.

### 6.7.5 Use of Lower Layer Services

*(This subclause applies to ITU-T Rec. X.519 only and not to ISO/IEC 9594-5)*

The session-service is defined in ITU-T Rec. X.215 | ISO/IEC 8326. The Session Layer structures the dialogue of the flow of information between the end-systems.

If the RTSE is included in an application context, the Kernel, Half-duplex, Exceptions, Minor-synchronize and Activity Management functional units of the Session Service are used by the Presentation Layer.

If the RTSE is not included in the application context, the Kernel and Duplex functional units of the Session Service are used by the Presentation Layer.

The transport-service is defined in ITU-T Rec. X.214 | ISO/IEC 8072. The Transport Layer provides for the end-to-end transparent transfer of data over the underlying network connection.

The choice of the class of transport-service used by the Session Layer depends on the requirements for multiplexing and error recovery. Support for Transport Class 0 (non-multiplexing) is mandatory. Transport Expedited Service is not used.

Support for other classes is optional. A multiplexing class may be used to multiplex the DAP or DSP and other protocols over the same network connection. An error recovery class may be chosen over a network connection with an unacceptable residual error rate.

An underlying network supporting the OSI network-service defined in ITU-T Rec. X.213 | ISO/IEC 8348 is assumed.

A network-address is as defined in ITU-T Recs. X.121, E.164, or X.213 | ISO/IEC 8348 (OSI NSAP-address).

### 6.7.6 Use of SESE Services

The Generic Upper Layers Security – Security Exchange Service Element is defined in ITU-T Rec. X.832 | ISO/IEC 11586-3.

The SESE supports the exchange of security exchange items to carry information required for authentication and association establishment (e.g. strong credentials). It can be used to support any number of exchanges.

Three security exchanges are defined in Annex G of this Directory Specification. One is for the two-way exchange of strong credentials including a time stamp (as defined in ITU-T Rec. X.511 | ISO/IEC 9594-3). The other is for a three-way exchange of strong credentials without a time stamp. The third is for a simple public key mechanism defined in RFC 2025.

Application contexts with and without SESE, using the security exchanges defined in Annex G, are defined to support DAP, DSP, DOP and DISP. DISP may be used with or without SESE and with or without RTSE.

If selected the security exchange shall be invoked on every association establishment (including RTSE association recovery).

# 7 Directory protocol abstract syntax

## 7.1 Abstract syntaxes

Two abstract syntaxes used in the Directory protocols are specified elsewhere. The abstract-syntax of ACSE, **acse-abstract-syntax**, is needed to establish the associations. The abstract-syntax of RTSE, **rtse-abstract-syntax**, is optionally needed for the DISP.

The ASN.1 type from which the values of the abstract syntaxes are derived is specified using the parameterized types **ROS {}**, **Bind {}**, and **Unbind {}** which are defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

These abstract syntaxes and those specified below shall (as a minimum) be encoded according to the Basic ASN.1 encoding rules.

NOTE – The abstract syntaxes defined in this clause that import from module **DirectoryShadowAbstractService** will use a mixture of implicit and explicit tags.

### 7.1.1 DAP Abstract Syntax

The Directory ASEs that realize the operation packages specified in 6.3 share a single abstract syntax, **directoryAccessAbstractSyntax**. This is specified as an information object of the class **ABSTRACT-SYNTAX**.

```
directoryAccessAbstractSyntax  ABSTRACT-SYNTAX  ::=  {
        DAP-PDUs
        IDENTIFIED BY     id-as-directoryAccessAS }

DAP-PDUs  ::=  CHOICE  {
        basicRos     ROS { { DAP-InvokeIDSet }, { DAP-Invokable }, { DAP-Returnable } },
        bind         Bind { directoryBind },
        unbind       Unbind { directoryUnbind } }

DAP-InvokeIDSet  ::=  InvokeId (ALL EXCEPT absent:NULL)

DAP-Invokable  OPERATION  ::=     { read | compare | abandon |
                                    list | search |
                                    addEtry | removeEntry | modifyEntry | modifyDN }

DAP-Returnable  OPERATION  ::=    { read | compare | abandon |
                                    list | search |
                                    addEntry | removeEntry | modifyEntry | modifyDN }
```

### 7.1.2 DSP Abstract Syntax

The Directory ASEs that realize the operation packages specified in 6.4 share a single abstract syntax, **directorySystemAbstractSyntax**. This is specified as an information object of the class **ABSTRACT-SYNTAX**.

```
directorySystemAbstractSyntax  ABSTRACT-SYNTAX  ::=  {
        DSP-PDUs
        IDENTIFIED BY     id-as-directorySystemAS }
```

```
DSP-PDUs ::= CHOICE {
        basicRos    ROS { {DSP-InvokeIDSet }, { DSP-Invokable }, { DSP-Returnable } },
        bind        Bind { dSABind },
        unbind      Unbind { dSAUnbind } }

DSP-InvokeIDSet ::= InvokeId (ALL EXCEPT absent:NULL)

DSP-Invokable OPERATION ::=     { chainedRead | chainedCompare | chainedAbandon |
                                  chainedList | chainedSearch |
                                  chainedAddEntry | chainedRemoveEntry | chainedModifyEntry |
                                  chainedModifyDN }

DSP-Returnable OPERATION ::=    { chainedRead | chainedCompare | chainedAbandon |
                                  chainedList | chainedSearch |
                                  chainedAddEntry | chainedRemoveEntry | chainedModifyEntry |
                                  chainedModifyDN }
```

### 7.1.3 DISP Abstract Syntax

The Directory ASEs that realize the operation packages specified in 6.5 either the abstract syntax **directoryShadowAbstractSyntax** or **directoryReliableShadowAbstractSyntax**, depending on whether RTSE is not or is used in the application context. These two abstract syntaxes are specified as information objects of the class **ABSTRACT-SYNTAX**.

```
directoryShadowAbstractSyntax ABSTRACT-SYNTAX ::= {
        DISP-PDUs
        IDENTIFIED BY    id-as-directoryShadowAS }

directoryReliableShadowAbstractSyntax ABSTRACT-SYNTAX ::= {
        Reliable-DISP-PDUs
        IDENTIFIED BY    id-as-directoryReliableShadowAS }
```

In addition, the following abstract syntax is used in the contexts employing RTSE. It is comprised of the abstract syntax of RTSE itself and the abstract syntax of **Bind { dSAShadowBind }**, and **Unbind { dSAShadowUnbind }**.

```
reliableShadowBindingAbstractSyntax    ABSTRACT-SYNTAX ::= {
        ReliableShadowBinding-PDUs
        IDENTIFIED BY    id-as-reliableShadowBindingAS }
```

The ASN.1 types from which the values of the abstract syntaxes are derived are specified using the **ROS {}**, **Bind {}**, and **Unbind {}** parameterized types.

```
DISP-PDUs ::= CHOICE {
        basicROS    ROS { { DISP-InvokeIDSet }, { DISP-Invokable }, {DISP-Returnable } },
        bind        Bind { dSAShadowBind },
        unbind      Unbind { dSAShadowUnbind } }

Reliable-DISP-PDUs ::= ROS { { DISP-InvokeIDSet }, { DISP-Invokable },
                    {DISP-Returnable } }

ReliableShadowBinding-PDUs ::= CHOICE {
        rTS         [0] RTSE-apdus,
        bind        Bind { dSAShadowBind },
        unbind      Unbind { dSAShadowUnbind } }

DISP-InvokeIDSet ::= InvokeID (ALL EXCEPT absent:NULL)

DISP-Invokable OPERATION ::=    { requestShadowUpdate | updateShadow |
                                  coordinateShadowUpdate }

DISP-Returnable OPERATION ::=   { requestShadowUpdate | updateShadow |
                                  coordinateShadowUpdate }
```

### 7.1.4 DOP Abstract Syntax

The Directory ASE that realizes the operation package specified in 6.6 employs the abstract syntax, **directory-OperationalBindingManagementAbstractSyntax**. This is specified as an information object of the class **ABSTRACT-SYNTAX**.

```
directoryOperationalBindingManagementAbstractSyntax ABSTRACT-SYNTAX ::= {
        DOP-PDUs
        IDENTIFIED BY    id-as-directoryOperationalBindingManagementAS }
```

```
DOP-PDUs ::= CHOICE {
        basicRos    ROS { { DOP-InvokeIDSet }, { DOP-Invokable }, { DOP-Returnable } },
        bind        Bind { directoryBind },
        unbind      Unbind { directoryUnbind } }

DOP-InvokeIDSet ::= InvokeId (ALL EXCEPT absent:NULL)

DOP-Invokable OPERATION ::=     { establishOperationalBinding |
                                  modifyOperationalBinding |
                                  terminateOperationalBinding }

DOP-Returnable OPERATION ::=    { establishOperationalBinding |
                                  modifyOperationalBinding |
                                  terminateOperationalBinding }
```

## 7.2 Directory application contexts

### 7.2.1 Directory Access Application Context

The **dapContract** is realized as the **directoryAccessAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
directoryAccessAC APPLICATION-CONTEXT ::= {
        CONTRACT                    dapContract
        ESTABLISHED BY              acse
        INFORMATION TRANSFER BY     pData
        ABSTRACT SYNTAXES           { acse-abstract-syntax | directoryAccessAbstractSyntax }
        APPLICATION CONTEXT NAME    id-ac-directoryAccessAC }
```

### 7.2.2 Directory System Application Context

The **dspContract** is realized as the **directorySystemAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
directorySystemAC APPLICATION-CONTEXT ::= {
        CONTRACT                    dspContract
        ESTABLISHED BY              acse
        INFORMATION TRANSFER BY     pData
        ABSTRACT SYNTAXES           { acse-abstract-syntax | directorySystemAbstractSyntax }
        APPLICATION CONTEXT NAME    id-ac-directorySystemAC }
```

### 7.2.3 Directory Shadow Application Contexts

If a DSA supports the DISP, that DSA shall support at least one of the shadow supplier roles or shadow consumer roles and at least one of the **shadowSupplierInitiatedAC** or the **shadowConsumerInitiatedAC**. If a DSA supports the **shadowSupplierInitiatedAC** for a particular role, it may also optionally support the **reliableShadowSupplierInitiated-AC** for the same role. If a DSA supports the **shadowConsumerInitiatedAC** for a particular role, it may also optionally support the **reliableShadowConsumerInitiatedAC** for the same role.

#### 7.2.3.1 Shadow Supplier Initiated Contexts

The **shadowSupplierContract** may be realized as the **shadowSupplierInitiatedAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
shadowSupplierInitiatedAC APPLICATION-CONTEXT ::= {
        CONTRACT                    shadowSupplierContract
        ESTABLISHED BY              acse
        INFORMATION TRANSFER BY     pData
        ABSTRACT SYNTAXES           { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME    id-ac-shadowSupplierInitiatedAC }
```

This application context requires that only synchronous operations be employed.

The **shadowSupplierInitiatedAsynchronousAC** variant of this application context permits the use of asynchronous operations.

```
shadowSupplierInitiatedAsynchronousAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowSupplierInitiatedAsynchronousAC }
```

The **shadowSupplierContract** may optionally be realized as the **reliableShadowSupplierInitiatedAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
reliableShadowSupplierInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowSupplierInitiatedAC }
```

#### 7.2.3.2 Shadow Consumer Initiated Contexts

The **shadowConsumerContract** may be realized as the **shadowConsumerInitiatedAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
shadowConsumerInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowConsumerInitiatedAC }
```

This application context requires that only synchronous operations be employed.

The **shadowConsumerInitiatedAsynchronousAC** variant of this application context permits the use of asynchronous operations.

```
shadowConsumerInitiatedAsynchronousAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowConsumerInitiatedAsynchronousAC }
```

The **shadowConsumerContract** may optionally be realized as the **reliableShadowConsumerInitiatedAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
reliableShadowConsumerInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowConsumerInitiatedAC }
```

### 7.2.4 Directory Operational Binding Management Application Context

The **dopContract** is realized as the **directoryOperationalBindingManagementAC**. This application context is specified as an information object of the class **APPLICATION-CONTEXT**.

```
directoryOperationalBindingManagementAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        dopContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        directoryOperationalBindingManagementAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-directoryOperationalBindingManagementAC }
```

## 7.3 Operation Codes

### 7.3.1 Operation Codes for DAP and DSP Packages

The following operation codes are used by the operation packages of the DAP and the DSP:

```
id-opcode-read              Code    ::=    local : 1
id-opcode-compare           Code    ::=    local : 2
id-opcode-abandon           Code    ::=    local : 3
id-opcode-list              Code    ::=    local : 4
id-opcode-search            Code    ::=    local : 5
id-opcode-addEntry          Code    ::=    local : 6
id-opcode-removeEntry       Code    ::=    local : 7
id-opcode-modifyEntry       Code    ::=    local : 8
id-opcode-modifyDN          Code    ::=    local : 9
```

### 7.3.2 Operation Codes for DISP Packages

The following operation codes are used by the operation packages of the DISP.

```
id-opcode-requestShadowUpdate       Code    ::=    local : 1
id-opcode-updateShadow              Code    ::=    local : 2
id-opcode-coordinateShadowUpdate    Code    ::=    local : 3
```

### 7.3.3 Operation Codes for DOP Packages

The following operation codes are used by the operation package of the DOP.

```
id-op-establishOperationalBinding   Code    ::=    local : 100
id-op-modifyOperationalBinding      Code    ::=    local : 102
id-op-terminateOperationalBinding   Code    ::=    local : 101
```

## 7.4 Error Codes

### 7.4.1 Error Codes for DAP and DSP Packages

The following error codes are used by the operation packages of the DAP and the DSP. The code **id-errcode-referral** is only used in the DAP. The code **id-opcode-dsaReferral** is only used in the DSP.

```
id-errcode-attributeError   Code    ::=    local : 1
id-errcode-nameError        Code    ::=    local : 2
id-errcode-serviceErro      Code    ::=    local : 3
id-errcode-referral         Code    ::=    local : 4
id-errcode-abandoned        Code    ::=    local : 5
id-errcode-securityError    Code    ::=    local : 6
id-errcode-abandonFailed    Code    ::=    local : 7
id-errcode-updateError      Code    ::=    local : 8
id-errcode-dsaReferral      Code    ::=    local : 9
```

### 7.4.2 Error Codes for DISP Packages

The following error code is used by the operation packages of the DISP.

```
id-errcode-shadowError      Code    ::=    local : 1
```

### 7.4.3 Error Codes for DOP Packages

The following error codes are used by the operation package of the DOP.

```
id-err-operationalBindingError      Code    ::=    local : 100
```

## 7.5 Versions and the rules for extensibility

The Directory may be distributed and more than two Directory Application Entities may interoperate to service a request. The Directory AEs may be implemented conforming to different editions of the Directory specification of the Directory service which may or may not be represented by different protocol version numbers. The version number is negotiated to the highest common version number between two directly binding Directory AEs.

NOTE 1 – There are currently two versions of each Directory protocol. The 1988 edition and the 1993 edition are of version 1. Most features added in post-1993 editions are also available in version 1. However, some enhanced services and protocols, e.g. signed errors, require that version 2 has been negotiated among all involved parties.

A DUA may issue a request as specified in the latest edition of the Directory specification to which the DUA was implemented. Using the rules of extensibility defined below, that request shall be forwarded to the appropriate DSA that will respond to that request, regardless of the edition of the intervening DSAs. The responding DSA shall function as defined below.

NOTE 2 – An intermediate DSA only chaining the request may choose to examine selected elements of the Directory APDU that is needed to perform its function, e.g. name resolution.

### 7.5.1 DUA to DSA

#### 7.5.1.1 Version negotiation

When accepting an association, i.e. binding, utilizing the DAP, the version negotiated shall only affect the point-to-point aspects of the protocol exchanged between the DUA and the DSA to which it is connected. Subsequent requests or responses on the association shall not be constrained by the version negotiated.

NOTE – There are no point-to-point aspects of the DAP that are currently indicated by different protocol versions.

#### 7.5.1.2 Request and response processing

The DUA may initiate requests using the highest edition of the specification of that request it supports. If one or more elements of the request are critical, it shall indicate the extension number(s) in the **criticalExtensions** parameter.

NOTE 1 – If a value defined by an extension is encoded in a **CHOICE**, **ENUMERATED**, or **INTEGER** (used as **ENUMERATED**) type and if that type is essential for proper operation in a DSA implemented according to an earlier edition of the Specification, it is recommended that the extension be marked critical.

When processing a request from a DUA, a DSA shall follow the rules defined in 7.5.2.2.

When processing a response, a DUA shall:

a) ignore all unknown bit name assignments within a bit string; and

b) ignore all unknown named numbers in an **ENUMERATED** type or **INTEGER** type that is being used in the enumerated style, provided the number occurs as an optional element of a **SET** or **SEQUENCE**; and

c) ignore all unknown elements in **SET**s, at the end of **SEQUENCE**s, or in **CHOICE**s where the **CHOICE** is itself an optional element of a **SET** or **SEQUENCE**.

NOTE 2 – Implementations may as a local option ignore certain additional elements in a Directory PDU. In particular, some unknown named numbers and unknown **CHOICE**s in mandatory elements of **SET**s and **SEQUENCE**s can be ignored without invalidating the operation. The identification of such elements is for further study.

d) not consider the receipt of unknown attribute types and attribute values as a protocol violation; and

e) optionally report the unknown attribute types and attribute values to the user.

#### 7.5.1.3 Extensibility rules for error handling

When processing a known error type with unknown indicated problems and parameters, a DUA shall:

a) not consider the receipt of unknown indicated problems and parameters as a protocol violation (i.e. it shall not issue a **RO-U-REJECT** or abort the application association); and

b) optionally report the additional error information to the user.

When processing an unknown error type, a DUA shall:

a) not consider the receipt of unknown error type as a protocol violation (i.e. it shall not issue a **RO-U-REJECT** or abort the application association); and

b) optionally report the error to the user.

### 7.5.2 DSA to DSA

#### 7.5.2.1 Version negotiation

When establishing or accepting an association, i.e. binding, utilizing the DSP, the version negotiated shall only effect the point-to-point aspects of the protocol exchanged between the DSAs. Subsequent requests or responses on the association shall not be constrained by the version negotiated.

NOTE 1 – There are no point-to-point aspects of the DSP that are currently indicated by different protocol versions.

When establishing or accepting an association, i.e. binding, utilizing the DISP, the version negotiated shall define all aspects of the protocol exchanged between the DSAs. Subsequent requests or responses on the association shall be constrained by the version negotiated.

NOTE 2 – There is currently only one version of the DISP protocol.

When establishing or accepting an association, i.e. binding, utilizing the DOP, the version negotiated shall define all aspects of the protocol exchanged between the DSAs. Subsequent requests or responses on the association shall be constrained by the version negotiated.

NOTE 3 – There is currently only one version of the DOP protocol.

#### 7.5.2.2 Rules of extensibility for operation processing

If any DSA performing an operation (after name resolution is completed) detects an element of **criticalExtensions** whose semantic is unknown, it shall return an **unavailableCriticalExtension** indication as a **serviceError** or in a **PartialOutcomeQualifier**.

NOTE 1 – If a **criticalExtensions** string with one or more zero values is received, this indicates either that the extensions corresponding to the values are not present in the operation or are not critical. The presence of a zero value in a **criticalExtensions** string shall not be inferred as either the presence or absence of the corresponding extension in the APDU.

Otherwise, when processing a Directory PDU a DSA shall:

a) ignore all unknown bit name assignments within a bit string; and

b) ignore all unknown named numbers in an **ENUMERATED** type or **INTEGER** type that is being used in the enumerated style, provided the number occurs as an optional element of a **SET** or **SEQUENCE**; and

c) ignore all unknown elements in **SET**s, at the end of **SEQUENCE**s, or in **CHOICE**s where the **CHOICE** is itself an optional element of a **SET** or **SEQUENCE**.

NOTE 2 – Implementations may, as a local option, ignore certain additional elements in a Directory PDU. In particular, some unknown named numbers and unknown **CHOICE**s in mandatory elements of **SET**s and **SEQUENCE**s can be ignored without invalidating the operation. The identification of such elements is for further study.

#### 7.5.2.3 Rules of extensibility for chaining

If the PDU is a request, the DSA shall forward the request containing the unknown types and values to any additional DSAs determined by the name resolution process.

If the PDU is a response, the DSA shall process the unknown types and values as it would process known types and values (see clause on results merging in the Directory Specification on Distributed Operations) and forward to the initiating DSA or DUA.

#### 7.5.2.4 Rules of extensibility for error handling

When processing an known error type with unknown indicated problems and parameters, a DSA:

a) shall not consider the receipt of unknown indicated problems and parameters as a protocol violation (i.e. it shall not issue a **RO-U-REJECT** or abort the application association); and

b) may attempt to recover as appropriate to its understanding of just the error type, or may just return the error (and its unknown indicated problems and parameters) to the next appropriate DSA or DUA.

When processing an unknown error type, a DSA which is only involved in chaining the request shall:

a) not consider the unknown error type as a protocol violation (i.e. it shall not issue a **RO-U-REJECT** or abort the application association); and

b) not attempt to correct or recover from the error and its indicated problems and parameters; and

c) return the unknown error type to the next appropriate DSA or DUA.

When processing an unknown error, a DSA which is correlating multiple responses shall:

a) not consider the unknown error type as a protocol violation (i.e. it shall not issue a **RO-U-REJECT** or abort the application association); and

b) not attempt to correct or recover from the error and its indicated problems and parameters; and

c) put the unknown error in **PartialOutcomeQualifier**; and

d) continue correlating results as usual.

# 8 Mapping onto used services

This clause defines the mapping of the DAP, DSP, DOP and DISP onto the used services.

The mapping onto used services of the DAP, DSP and DOP, as well as for the DISP application contexts that omit the RTSE is defined in 8.1. The mapping onto used services for the DISP application contexts that use the RTSE is defined in 8.2.

## 8.1 Application contexts omitting RTSE

This subclause defines the mapping onto used services of the DAP, DSP and DOP application contexts, as well as the DISP application contexts that do not include the RTSE.

### 8.1.1 Mapping onto ACSE

This subclause defines the mapping of the (**DirectoryBind**, **DSABind**, **DSAShadowBind** or **DSADOPBind**) and (**DirectoryUnbind**, **DSAUnbind**, **DSAShadowUnbind** or **DSADOPUnbind**) services onto the services of the ACSE. The ACSE is defined in ITU-T Rec. X.217 | ISO/IEC 8649.

If the SESE is included in the application context the first two security exchange items of the security exchange are mapped on to the ACSE A-ASSOCIATE parameters for the authentication functional unit as defined in ITU-T Rec. X.832 | ISO/IEC 11586-3.

#### 8.1.1.1 Bind onto A-ASSOCIATE

The **DirectoryBind**, **DSABind**, **DSAShadowBind** or **DSADOPBind** service is mapped onto the A-ASSOCIATE service of the ACSE. The use of the parameters of the A-ASSOCIATE service is qualified in the following subclauses.

##### 8.1.1.1.1 Mode

This parameter shall be supplied by the initiator of the association in the A-ASSOCIATE request primitive, and shall have the value 'normal mode'.

##### 8.1.1.1.2 Application context name

The initiator of the association shall propose one of the following application contexts if SESE is not used:

a) For the DAP, the **directoryAccessAC**;

b) For the DSP, the **directorySystemAC**;

c) For the DOP, the **directoryOperationalBindingManagementAC**;

d) For the DISP, one of **shadowSupplierInitiatedAC**, **shadowConsumerInitiatedAC**, **shadowSupplierInitiatedAsynchronousAC**, or **shadowConsumerInitiatedAsynchronousAC**.

The initiator of the association shall propose one of the following application contexts if using SESE with two-way or three-way authentication exchanges as defined in Annex G:

e) For the DAP with the security exchanges, the **directoryAccessWith2or3seAC**;

f) For the DSP with SESE, the **directorySystemWith2or3seAC**;

g) For the DOP with SESE, the **directoryOperationalBindingManagementWith2or3seAC**;

h) For the DISP with SESE, either the **shadowSupplierInitiatedWith2or3seAC** or the **shadowConsumerInitiatedWith2or3seAC**.

##### 8.1.1.1.3 User Information

The mapping of the **DirectoryBind** or **DSABind** onto the User Information parameters of the A-ASSOCIATE request primitive is defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

##### 8.1.1.1.4 Presentation Context Definition List

The initiator of the association shall supply the Presentation Context Definition List in the A-ASSOCIATE request primitive which shall contain the ACSE abstract-syntax (**id-as-acse**) and either the DAP abstract syntax (**id-as-directoryAccessAS**), the DSP abstract syntax (**id-as-directorySystemAS**), the DOP abstract syntax (**id-as-directoryOperationalBinding ManagementAS**), or the DISP abstract syntax (**id-as-directoryShadowAS**).

If the SESE, with two-way or three-way authentication exchanges as defined in Annex G, is included in the application context then the Presentation Context Definition List shall also contain the abstract syntax **id-as-2or3se**.

#### 8.1.1.1.5 Quality of Service

This parameter shall be supplied by the initiator of the association in the A-ASSOCIATE request primitive, and by the responder of the association in the A-ASSOCIATE response primitive. The parameters 'Extended Control' and 'Optimized Dialogue Transfer' shall be set to "feature not desired". The remaining parameters shall be such that default values are used.

#### 8.1.1.1.6 Session Requirements

This parameter shall be set by the initiator of the association in the A-ASSOCIATE request primitive, and by the responder of the association in the A-ASSOCIATE response primitive. The parameter shall be set to specify the following functional units:

a) Kernel;

b) Duplex.

#### 8.1.1.1.7 Application Entity Title and Presentation Address

These parameters shall be supplied by the initiator and the responder of the association (Application Entity Title is optionally supplied).

For a DUA establishing an association for an initial request, these parameters are obtained from locally held information.

For a DUA (or DSA) establishing an association with a DSA to which it has been referred, these parameters are obtained from the **AccessPoint** value of a **Continuation Reference**.

For a DSA establishing an association, this parameter is obtained from its knowledge information, i.e. an external reference.

### 8.1.1.2 Unbind onto A-RELEASE

The **DirectoryUnbind**, **DSAUnbind**, **DSAShadowUnbind** or **DSADOPUnbind** is mapped onto the A-RELEASE service of the ACSE. The use of the parameters of the A-RELEASE service is qualified in the following subclause.

#### 8.1.1.2.1 Result

This parameter shall have the value 'affirmative'.

### 8.1.1.3 Use of A-ABORT and A-P-ABORT Services

The application-process is the user of the A-ABORT and A-P-ABORT services of the ACSE.

### 8.1.2 Mapping onto ROSE

The Directory ASE services are mapped onto the RO-INVOKE, RO-RESULT, RO-ERROR, RO-REJECT-U and RO-REJECT-P services of the ROSE. The mapping of the abstract-syntax notation of the Directory ASEs onto the ROSE services is as defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

### 8.1.3 Mapping of SESE to Presentation Layer

For application contexts including SESE, if the security exchange involves more than a two-way exchange, the third and subsequent SE-TRANSFER are mapped on to P-DATA.

## 8.2 Application contexts including RTSE

This subclause defines the mapping onto used services for the DISP application contexts that include the RTSE. Support for this mapping is conditional on a claim of conformance to these application contexts. The RTSE is defined in ITU-T Rec. X.218 | ISO/IEC 9066-1.

### 8.2.1 Mapping onto RT-OPEN and RT-CLOSE

This subclause defines the mapping of the **DSAShadowBind** and **DSAShadowUnbind** services onto the RT-OPEN and RT-CLOSE services of the RTSE.

#### 8.2.1.1 DSAShadowBind onto RT-OPEN

The **DSAShadowBind** is mapped onto the RT-OPEN service of the RTSE. The use of the parameters of the RT-OPEN service is qualified in the following subclauses.

#### 8.2.1.1.1 Mode

This parameter shall be supplied by the initiator of the association in the RT-OPEN request primitive, and shall have the value "normal mode".

#### 8.2.1.1.2 Application context name

If SESE is not used, the initiator of the association shall propose either the **reliableShadowSupplierInitiatedAC** application context or the **reliableShadowConsumerInitiatedAC** application context in the RT-OPEN request primitive.

If SESE is used with two-way or three-way authentication exchanges as defined in Annex G, then the initiator of the association shall propose either the **reliableShadowSupplierInitiatedWith2or3seAC** application context or the **reliableShadowConsumerInitiatedWith2or3seAC** application context in the RT-OPEN request primitive.

#### 8.2.1.1.3 User-data

The mapping of the bind-operation onto the user-data parameter of the RT-OPEN request primitive is defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

#### 8.2.1.1.4 Presentation Context Definition List

If the SESE, with two-way or three-way authentication exchanges as defined in Annex G, is included in the application context then the Presentation Context Definition List shall also contain the abstract syntax **id-as-2or3se**.

#### 8.2.1.1.5 Initial turn

This parameter shall be supplied by the initiator of the association in the RT-OPEN request primitive, and shall have the value "association-initiator".

#### 8.2.1.1.6 Application Entity Title and Presentation Address

These parameters shall be supplied by the initiator and the responder of the association in the RT-OPEN request primitive (Application Entity Title is optionally supplied).

### 8.2.1.2 DSAShadowUnbind onto RT-CLOSE

The **DSAShadowUnbind** is mapped onto the RT-CLOSE service of the RTSE.

### 8.2.2 Mapping onto ROSE

The **shadowSupplierASE** and the **shadowConsumerASE** services are mapped onto the RO-INVOKE, RO-RESULT, RO-ERROR, RO-REJECT-U and RO-REJECT-P services of the ROSE. The mapping of the abstract-syntax notation of these DISP ASEs onto the ROSE services is as defined in ITU-T Rec. X.880 | ISO/IEC 13712-1.

ROSE is the user of the RT-TRANSFER, RT-TURN-PLEASE, RT-TURN-GIVE, RT-P-ABORT and RT-U-ABORT services of the RTSE. The use of the RTSE services by the ROSE is defined in ITU-T Rec. X.882 | ISO/IEC 13712-3.

### 8.2.2.1 Managing the turn

ITU-T Rec. X.881 | ISO/IEC 13712-2 defines the use by the ROSE of the RT-TURN-PLEASE and RT-TURN-GIVE services of the RTSE to manage the turn.

The values of the priority parameter of the RT-TURN-PLEASE service used by the ROSE to request the turn are as follows:

- *Priority zero* is the highest priority, and is reserved for the action of releasing the association by the initiator.
- *Priority one* is used by the ROSE to provide the RO-REJECT-U and RO-ERROR services of the ROSE.
- *Priority two* is used by the ROSE to provide the RO-RESULT service of the ROSE.
- *Priority three* is used by the ROSE to provide the RO-INVOKE service of the ROSE.

#### 8.2.3 Mapping of SESE to ACSE and Presentation Layer

If the SESE is included in the application context, the first two SESE APDUs for the security exchange are mapped on to the ACSE A-ASSOCIATE parameters for the authentication functional unit as defined in ITU-T Rec. X.832 | ISO/IEC 11586-3.

If the security exchange involves more than a two-way exchange, subsequent SESE APDUs are mapped on to P-TYPE-DATA.

## 9 Conformance

This clause defines the requirements for conformance to this Directory Specification.

### 9.1 Conformance by DUAs

A DUA implementation claiming conformance to this Directory Specification shall satisfy the requirements specified in 9.1.1 through 9.1.3.

#### 9.1.1 Statement requirements

The following shall be stated:

a) the operations of the **directoryAccessAC** or **directoryAccessWith2or3seAC** application-context that the DUA is capable of invoking for which conformance is claimed.

b) The bind security level(s) for which conformance is claimed (none, simple, strong – and if simple, then whether without password, with password or with protected-password); and whether the DUA can generate signed arguments or validate signed results.

c) Whether conformance is claimed to Rule-based Access Control.

d) The application-contexts for which conformance is claimed: **directoryAccessAC** or **directoryAccessWith2or3seAC**.

#### 9.1.2 Static requirements

A DUA shall:

a) have the capability of supporting the **directoryAccessAC** or **directoryAccessWith2or3seAC** application-context as defined by its abstract syntax in clause 7;

b) conform to the extensions for which conformance was claimed in 9.1.1, c);

c) if conformance is claimed to Rule-based Access Control, have the capability of supporting security labels as identified in ITU-T Rec. X.501 | ISO/IEC 9594-2 subclause 17.4.

#### 9.1.3 Dynamic Requirements

A DUA shall:

a) conform to the mapping onto used services defined in clause 8;

b) shall conform to the rules of extensibility procedures defined in 7.5.1.

### 9.2 Conformance by DSAs

A DSA implementation claiming conformance to this Directory Specification shall satisfy the requirements specified in 9.2.1 through 9.2.3.

#### 9.2.1 Statement requirements

The following shall be stated:

a) The application-contexts for which conformance is claimed: **directoryAccessAC**, **directorySystemAC**, **directoryOperationalBindingManagementAC**, **directoryAccessWith2or3seAC**, **directorySystemWith2or3seAC**, or a combination of these. A DSA that claims conformance to the **directoryOperationalBinding ManagementAC** in support of hierarchical operational bindings shall also support the **directorySystemAC**. If a DSA is such that knowledge of it has been disseminated, causing knowledge references to the DSA to be held in other DSAs outside of its own DMD, then it shall claim conformance to the **directorySystemAC**.

NOTE 1 – An application context shall not be divided except as stated herein; in particular, conformance shall not be claimed to particular operations.

b) The operational binding types for which conformance is claimed: **shadowOperationalBindingID**, **specificHierarchicalBindingID**, **non-specificHierarchicalBindingID**, or a combination of these. A DSA that claims conformance to the **shadowOperationalBindingID** shall support one or more of the application contexts for shadow suppliers and/or shadow consumers indicated in 9.3 and 9.4.

c) Whether or not the DSA is capable of acting as a first-level DSA, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

d) If conformance is claimed to the **directorySystemAC** or **directorySystemWith2or3seAC** application-context, whether or not the chained mode of operation is supported, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

e) If conformance is claimed to the **directoryAccessAC** application context, the bind security level(s) for which conformance is claimed (none, simple, strong – and if simple, then whether without password, with password, or with protected password); whether the DSA can perform originator authentication as defined in 22.1 of ITU-T Rec. X.518 | ISO/IEC 9594-4 and if so, whether identity-based or signature-based; and whether the DSA can perform result authentication as defined in 22.2 of ITU-T Rec. X.518 | ISO/IEC 9594-4.

f) If conformance is claimed to the **directorySystemAC** application context, the bind security level(s) for which conformance is claimed (none, simple, strong – and if simple, then whether without password, with password, or with protected password); whether the DSA can perform originator authentication as defined in 22.1 of ITU-T Rec. X.518 | ISO/IEC 9594-4 and if so, whether identity-based or signature-based; and whether the DSA can perform result authentication as defined in 22.2 of ITU-T Rec. X.518 | ISO/IEC 9594-4.

g) The selected attribute types defined in ITU-T Rec. X.520 | ISO/IEC 9594-6, and any other attribute types, for which conformance is claimed and whether for attributes based on the syntax **DirectoryString** conformance is claimed for the choice **BMPString**, **UniversalString**, or both.

h) The selected object classes defined in ITU-T Rec. X.521 | ISO/IEC 9594-7, and any other object classes, for which conformance is claimed.

i) The extensions listed in the table of 7.3.1 of ITU-T Rec. X.511 | ISO/IEC 9594-3, that the DSA is capable of responding to for which conformance is claimed.

j) Whether conformance is claimed for collective attributes as defined in 8.8 of ITU-T Rec. X.501 | ISO/IEC 9594-2 and 7.6, 7.8.2 and 9.2.2 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

k) Whether conformance is claimed for hierarchical attributes as defined in 7.6, 7.8.2 and 9.2.2 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

l) The operational attribute types defined in ITU-T Rec. X.501 | ISO/IEC 9594-2 and any other operational attribute types for which conformance is claimed.

m) Whether conformance is claimed for return of alias names as described in 7.7.1 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

n) Whether conformance is claimed for indicating that returned entry information is complete, as described in 7.7.6 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

o) Whether conformance is claimed for modifying the object class attribute to add and/or remove values identifying auxiliary object classes, as described in 11.3.2 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

p) Whether conformance is claimed to Basic Access Control.

q) Whether conformance is claimed to Simplified Access Control.

r) Whether the DSA is capable of administering the subschema for its portion of the DIT, as defined in ITU-T Rec. X.501 | ISO/IEC 9594-2.

NOTE 2 – The capability to administer a subschema shall not be divided; specifically, the capability to administer particular subschema definitions shall not be claimed.

s) The selected name bindings defined in ITU-T Rec. X.521 | ISO/IEC 9594-7 and any other name bindings, for which conformance is claimed.

t) Whether the DSA is capable of administering collective attributes, as defined in ITU-T Rec. X.501 | ISO/IEC 9594-2.

u) The selected context types defined in ITU-T Rec. X.520 | ISO/IEC 9594-6, and any other context types, for which conformance is claimed.

v) Whether conformance is claimed for contexts as defined in 8.7, 8.8 and 11.8 of ITU-T Rec. X.501 | ISO/IEC 9594-2, and 7.3 and 7.6 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

w) Whether conformance is claimed for the use of contexts in RDNs, as defined in 8.5 and 9.3 of ITU-T Rec. X.501 | ISO/IEC 9594-2, 7.7 of ITU-T Rec. X.511 | ISO/IEC 9594-3, and ITU-T Rec. X.518 | ISO/IEC 9594-4.

x) Whether conformance is claimed for the management of the DSA Information Tree, as defined in 7.13 of ITU-T Rec. X.511 | ISO/IEC 9594-3.

y) Whether conformance is claimed for the use of systems management for administration of the Directory, as defined in ITU-T Rec. X.530 | ISO/IEC 9594-10.

z) The selected managed objects and management attribute types defined in ITU-T Rec. X.530 | ISO/IEC 9594-10, and any other managed objects and attributes, for which conformance is claimed.

aa) Whether conformance is claimed to Rule-based Access Control.

NOTE 3 – The support of security labels requires the following minimal support of contexts: Context lists as per 8.7 of ITU-T X.501 | ISO/IEC 9594-2 and returnContexts per 7.6 of ITU-T X.511 | ISO/IEC 9594-3.

bb) Whether conformance is claimed to integrity of Directory operations.

cc) Whether conformance is claimed to integrity and confidentiality of Directory operations.

dd) Whether conformance is claimed that the DSA can hold and provide access to encrypted and digitally signed information.

### 9.2.2 Static requirements

A DSA shall:

a) have the capability of supporting the application-contexts for which conformance is claimed as defined by their abstract syntax in clause 7;

b) have the capability of supporting the information framework defined by its abstract syntax in ITU-T Rec. X.501 | ISO/IEC 9594-2;

c) conform to the minimal knowledge requirements defined in ITU-T Rec. X.518 | ISO/IEC 9594-4;

d) if conformance is claimed as a first-level DSA, conform to the requirements support of the root context, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4;

e) have the capability of supporting the attribute types for which conformance is claimed; as defined by their abstract syntaxes;

f) have the capability of supporting the object classes for which conformance is claimed, as defined by their abstract syntaxes;

g) conform to the extensions for which conformance was claimed in 9.2.1 i);

h) if the capability to administer subschema as defined in ITU-T Rec. X.501 | ISO/IEC 9594-2 is claimed, the DSA shall be able to do this administration;

i) if conformance is claimed for collective attributes, have the capability of performing the related procedures defined in 7.6, 7.8.2 and 9.2.2 of ITU-T Rec. X.511 | ISO/IEC 9594-3;

j) if conformance is claimed for hierarchical attributes, have the capability of performing the related procedures defined in 7.6, 7.8.2 and 9.2.2 of ITU-T Rec. X.511 | ISO/IEC 9594-3;

k) have the capability of supporting the operational attribute types for which conformance is claimed;

l) if conformance is claimed to Basic Access Control, have the capability of holding ACI items that conform to the definitions of Basic Access Control;

m) if conformance is claimed to Simplified Access Control, have the capability of holding ACI items that conform to the definitions of Simplified Access Control;

n) have the capability of supporting the context types for which conformance is claimed, as defined by their abstract syntaxes;

o) if conformance is claimed for contexts, have the capability of performing the related procedures defined in ITU-T Rec. X.511 | ISO/IEC 9594-3;

p) if conformance is claimed for the use of contexts in RDNs, have the capability of performing the related procedures as defined in 9.3 of ITU-T Rec. X.501 | ISO/IEC 9594-2, 7.7 of ITU-T Rec. X.511 | ISO/IEC 9594-3, and ITU-T Rec. X.518 | ISO/IEC 9594-4;

q) if conformance is claimed for the management of the DSA Information Tree, have the capability of performing the related procedures as defined in 7.5 and 7.13 of ITU-T Rec. X.511 | ISO/IEC 9594-3;

r) if conformance is claimed for the use of systems management for administration of the Directory, have the capability of performing the related procedures as defined in ITU-T Rec. X.530 | ISO/IEC 9594-10 for the managed objects for which conformance is claimed;

s) if conformance is claimed to Rule-Based Access Control, have the capability of holding ACI items that conform to the definition of Rule-Based Access Control;

t) if conformance is claimed to integrity of Directory operations, be capable of signing all directory operations supported;

u) if conformance is claimed to integrity and confidentiality of Directory operations, be capable of signing and encrypting all directory operations supported;

v) if conformance is claimed to integrity of directory information in storage be capable to supporting the **attributeValueIntegrityInfoContext** to protect directory information;

w) if conformance is claimed to cryptographic protection of Directory information in storage be able to encrypt attributes for which conformance is claimed.

### 9.2.3 Dynamic requirements

A DSA shall:

a) Conform to the mapping onto used services defined in clause 8.

b) Conform to the procedures for distributed operation of the Directory related to referrals, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

c) If conformance is claimed to the **directoryAccessAC** or **directoryAccessWith2or3seAC** application-context, conform to the procedures of ITU-T Rec. X.518 | ISO/IEC 9594-4 as they relate to the referral mode of the DAP.

d) If conformance is claimed to the **directorySystemAC** or **directorySystemWith2or3seAC** application-context, conform to the referral mode of interaction, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

e) If conformance is claimed to the chained mode of interaction, conform to the chained mode of interaction, as defined in ITU-T Rec. X.518 | ISO/IEC 9594-4.

NOTE – Only in this case is it necessary for a DSA to be capable of invoking operations of the **directorySystemAC**.

f) Conform to the rules of extensibility procedures defined in 7.5.2.

g) If conformance is claimed to Basic Access Control, have the capability of protecting information within the DSA in accordance with the procedures of Basic Access Control.

h) If conformance is claimed to Simplified Access Control, have the capability of protecting information within the DSA in accordance with the procedures of Simplified Access Control.

i) If conformance is claimed for the **shadowOperationalBindingID**, conform to the procedures of ITU-T Rec. X.525 | ISO/IEC 9594-9 and ITU-T Rec. X.501 | ISO/IEC 9594-2 as they relate to the DOP.

j) If conformance is claimed for the **specificHierarchicalBindingID**, conform to the procedures of ITU-T Rec. X.518 | ISO/IEC 9594-4 and ITU-T X.501 | ISO/IEC 9594-2 as they relate to specific hierarchical operational bindings.

k) If conformance is claimed for the **non-specificHierarchicalBindingID**, conform to the procedures of ITU-T Rec. X.518 | ISO/IEC 9594-4 and ITU-T Rec. X.501 | ISO/IEC 9594-2 as they relate to non-specific hierarchical operational bindings.

l) If conformance is claimed for the use of contexts in RDNs, conform to name resolution involving contexts as defined in 9.4 of ITU-T Rec. X.501 | ISO/IEC 9594-2, and 10.3, 10.4, 10.6, 10.9, 10.10 and 15.5.4 of ITU-T Rec. X.518 | ISO/IEC 9594-4.

m) If conformance is claimed to Rule-based Access Control, have the capability of protecting information within the DSA in accordance with the procedures of Rule-based Access Control.

## 9.3 Conformance by a shadow supplier

A DSA implementation claiming conformance to this Directory Specification in the role of shadow supplier shall satisfy the requirements specified in 9.3.1 through 9.3.3.

### 9.3.1 Statement requirements

The following shall be stated:

a) The application context(s) for which conformance is claimed as a shadow supplier: **shadowSupplierInitiatedAC**, **shadowConsumerInitiatedAC**, **shadowSupplierInitiatedAsynchronousAC**, **shadowConsumerInitiatedAsynchronousAC**, **reliableShadowSupplierInitiatedAC**, **reliableShadowConsumerInitiatedAC**, **shadowSupplierInitiatedWith2or3seAC**, and **shadowConsumerInitiatedWith2or3seAC**.

A DSA shall, at a minimum, support either the **shadowSupplierInitiatedAC** or the **shadowConsumerInitiatedAC**. If the DSA supports the **shadowSupplierInitiatedAC**, it may optionally support one or both of the **shadowSupplierInitiatedAsynchronousAC** or **reliableShadowSupplierInitiatedAC**. If the DSA supports the **shadowConsumerInitiatedAC**, it may optionally support one or both of the **shadowConsumerInitiatedAsynchronousAC** or **reliableShadowConsumerInitiatedAC**.

b) The security-level(s) for which conformance is claimed (none, simple, strong).

c) To which degree the **UnitOfReplication** is supported. Specifically, which (if any) of the following optional features are supported:

- Entry filtering on **objectClass**;
- Selection/Exclusion of attributes via **AttributeSelection**;
- The inclusion of subordinate knowledge in the replicated area;
- The inclusion of extended knowledge in addition to subordinate knowledge;
- Selection/Exclusion of attribute values based on contexts.

### 9.3.2 Static requirements

A DSA shall:

a) have the capability of supporting the application-context(s) for which conformance is claimed as defined in their abstract syntax in clause 7;

b) provide support for **modifyTimestamp** and **createTimestamp** operational attributes.

### 9.3.3 Dynamic requirements

A DSA shall:

a) conform to the mapping onto used services defined in clause 8;

b) conform to the procedures of ITU-T Rec. X.525 | ISO/IEC 9594-9 as they relate to the DISP.

## 9.4 Conformance by a shadow consumer

A DSA implementation claiming conformance to this Directory Specification as a shadow consumer shall satisfy the requirements specified in 9.4.1 through 9.4.3.

### 9.4.1 Statement requirements

The following shall be stated:

a) The application context(s) for which conformance is claimed as a shadow consumer: **shadowSupplierInitiatedAC**, **shadowConsumerInitiatedAC**, **shadowSupplierInitiatedAsynchronousAC**, **shadowConsumerInitiatedAsynchronousAC**, **reliableShadowSupplierInitiatedAC**, **reliableShadowConsumerInitiatedAC**, **shadowSupplierInitiatedWith2or3seAC**, and **shadowConsumerInitiatedWith2or3seAC**.

A DSA shall, at a minimum, support either the **shadowSupplierInitiatedAC** or the **shadowConsumerInitiatedAC**. If the DSA supports the **shadowSupplierInitiatedAC**, it may optionally support one or both of the **shadowSupplierInitiatedAsynchronousAC** or **reliableShadowSupplierInitiatedAC**. If the DSA supports the **shadowConsumerInitiatedAC** it may optionally support one or both of the **shadowConsumerInitiatedAsynchronousAC** or **reliableShadowConsumerInitiatedAC**.

b) the security-level(s) for which conformance is claimed (none, simple, strong);

c) whether the DSA can act as a secondary shadow supplier (i.e. participate in secondary shadowing as an intermediate DSA);

d) whether the DSA supports shadowing of overlapping units of replication.

### 9.4.2 Static requirements

A DSA shall:

a) have the capability of supporting the application-context(s) for which conformance is claimed as defined in their abstract syntax in clause 7;

b) provide support for **modifyTimestamp** and **createTimestamp** operational attributes if overlapping units of replication is supported;

c) provide support for the **copyShallDo** service control.

### 9.4.3 Dynamic requirements

A DSA shall:

a) conform to the mapping onto used services defined in clause 8;

b) conform to the procedures of ITU-T Rec. X.525 | ISO/IEC 9594-9 as they relate to the DISP.

# Annex A

# DAP in ASN.1

(This annex forms an integral part of this Recommendation | International Standard)

This annex includes all of the ASN.1 type and value definitions contained in this Directory Specification, in the form of the ASN.1 module, "**DirectoryAccessProtocol**".

---

```
DirectoryAccessProtocol {joint-iso-itu-t ds(5) module(1) dap(11) 3}
DEFINITIONS ::=
BEGIN

-- EXPORTS All --

-- The types and values defined in this module are exported for use in the other ASN.1 modules contained
-- within the Directory Specifications, and for the use of other applications which will use them to access
-- Directory services. Other applications may use them for their own purposes, but this will not constrain
-- extensions and modifications needed to maintain or improve the Directory service.

IMPORTS
        directoryAbstractService , protocolObjectIdentifiers, directorySecurityExchanges
                FROM  UsefulDefinitions  {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 3}

        ROS-OBJECT-CLASS, CONTRACT, OPERATION-PACKAGE, CONNECTION-PACKAGE,
        Code, OPERATION
                FROM Remote-Operations-Information-Objects
                        {joint-iso-ccitt remote-operations(4) informationObjects(5) version1(0)}

        ROS{}, Bind{}, Unbind{}, InvokeId
                FROM Remote-Operations-Generic-ROS-PDUs
                        {joint-iso-ccitt remote-operations(4) generic-ROS-PDUs(6) version1(0)}

        APPLICATION-CONTEXT
                FROM Remote-Operations-Information-Objects-extensions {joint-iso-ccitt
                        remote-operations(4) informationObjects-extensions(8) version1(0)}

        acse, pData
                FROM Remote-Operations-Realisations
                        {joint-iso-ccitt remote-operations(4) realisations(9) version1(0)}

        acse-abstract-syntax
                FROM Remote-Operations-Abstract-Syntaxes {joint-iso-ccitt remote-operations(4)
                        remoteOperationsAabstractSyntaxes(12) version1(0)}

        id-ac-directoryAccessAC, id-ac-directoryAccessWith2or3seAC, id-rosObject-dua,
        id-rosObject-directory, id-rosObject-dapDSA,
        id-contract-dap, id-package-dapConnection, id-package-read, id-package-search,
        id-package-modify, id-as-directoryAccessAS
                FROM ProtocolObjectIdentifiers protocolObjectIdentifiers

        directoryBind, directoryUnbind, read, compare, abandon, list, search, addEntry, removeEntry,
        modifyEntry, modifyDN
                FROM DirectoryAbstractService directoryAbstractService
```

```
        dir2or3se
            FROM DirectorySecurityExchanges directorySecurityExchanges  ;

-- application contexts --

directoryAccessAC  APPLICATION-CONTEXT  ::=  {
        CONTRACT                        dapContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryAccessAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-directoryAccessAC }

directoryAccessWith2or3seAC  APPLICATION-CONTEXT  ::=  {
        CONTRACT                        dapContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        directoryAccessAbstractSyntax |
                                        dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-directoryAccessWith2or3seAC }

-- ROS objects --

dua  ROS-OBJECT-CLASS  ::=  {
        INITIATES       { dapContract }
        ID              id-rosObject-dua }

directory  ROS-OBJECT-CLASS  ::=  {
        RESPONDS        { dapContract }
        ID              id-rosObject-directory }

dap-dsa  ROS-OBJECT-CLASS  ::=  {
        RESPONDS        { dapContract }
        ID              id-rosObject-dapDSA }

-- contracts --

dapContract  CONTRACT  ::=  {
        CONNECTION                  dapConnectionPackage
        INITIATOR CONSUMER OF       { readPackage | searchPackage | modifyPackage }
        ID                          id-contract-dap }

-- connection package --

dapConnectionPackage  CONNECTION-PACKAGE  ::=  {
        BIND        directoryBind
        UNBIND      directoryUnbind
        ID          id-package-dapConnection }

--  read package --

readPackage  OPERATION-PACKAGE  ::=  {
        CONSUMER INVOKES            { read | compare | abandon }
        ID                          id-package-read }

--  search package --

searchPackage  OPERATION-PACKAGE  ::=  {
        CONSUMER INVOKES            { list | search }
        ID                          id-package-search }

--  modify Package  --

modifyPackage  OPERATION-PACKAGE  ::=  {
        CONSUMER INVOKES            { addEntry | removeEntry | modifyEntry | modifyDN }
        ID                          id-package-modify }

-- abstract syntaxes --

directoryAccessAbstractSyntax  ABSTRACT-SYNTAX  ::=  {
        DAP-PDUs
        IDENTIFIED BY    id-as-directoryAccessAS }
```

```
DAP-PDUs  ::=  CHOICE  {
        basicRos    ROS { { DAP-InvokeIDSet }, { DAP-Invokable }, { DAP-Returnable } },
        bind        Bind { directoryBind },
        unbind      Unbind { directoryUnbind } }

DAP-InvokeIDSet  ::=  InvokeId (ALL EXCEPT absent:NULL)

DAP-Invokable  OPERATION  ::=    { read | compare | abandon |
                                 list | search |
                                 addEtry | removeEntry | modifyEntry | modifyDN }

DAP-Returnable  OPERATION  ::=   { read | compare | abandon |
                                 list | search |
                                 addEntry | removeEntry | modifyEntry | modifyDN }

-- remote operation codes --

id-opcode-read              Code        ::=    local : 1
id-opcode-compare           Code        ::=    local : 2
id-opcode-abandon           Code        ::=    local : 3
id-opcode-list              Code        ::=    local : 4
id-opcode-search            Code        ::=    local : 5
id-opcode-addEntry          Code        ::=    local : 6
id-opcode-removeEntry       Code        ::=    local : 7
id-opcode-modifyEntry       Code        ::=    local : 8
id-opcode-modifyDN          Code        ::=    local : 9

-- remote error codes  --

id-errcode-attributeError                Code        ::=    local : 1
id-errcode-nameError                     Code        ::=    local : 2
id-errcode-serviceError                  Code        ::=    local : 3
id-errcode-referral                      Code        ::=    local : 4
id-errcode-abandoned                     Code        ::=    local : 5
id-errcode-securityError                 Code        ::=    local : 6
id-errcode-abandonFailed                 Code        ::=    local : 7
id-errcode-updateError                   Code        ::=    local : 8

-- remote error code for DSP --

id-errcode-dsaReferral                   Code        ::=    local : 9

END
```

# Annex B

# DSP in ASN.1

(This annex forms an integral part of this Recommendation | International Standard)

This annex includes all of the ASN.1 type and value definitions contained in this Directory Specification, in the form of the ASN.1 module, "**DirectorySystemProtocol**".

---

```
DirectorySystemProtocol {joint-iso-itu-t ds(5) module(1) dsp(12) 3}
DEFINITIONS ::=
BEGIN

-- EXPORTS All --

-- The types and values defined in this module are exported for use in the other ASN.1 modules contained
-- within the Directory Specifications, and for the use of other applications which will use them to access
-- Directory services. Other applications may use them for their own purposes, but this will not constrain
-- extensions and modifications needed to maintain or improve the Directory service.

IMPORTS

    distributedOperations, protocolObjectIdentifiers, directorySecurityExchanges
        FROM UsefulDefinitions  {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 3}

    ROS-OBJECT-CLASS, CONTRACT, OPERATION-PACKAGE, CONNECTION-PACKAGE,
    Code, OPERATION
        FROM Remote-Operations-Information-Objects
            {joint-iso-ccitt remote-operations(4) informationObjects(5) version1(0)}

    ROS{}, Bind{}, Unbind{}, InvokeId
        FROM Remote-Operations-Generic-ROS-PDUs
            {joint-iso-ccitt remote-operations(4) generic-ROS-PDUs(6) version1(0)}

    APPLICATION-CONTEXT
        FROM Remote-Operations-Information-Objects-extensions {joint-iso-ccitt
            remote-operations(4) informationObjects-extensions(8) version1(0)}

    acse, pData
        FROM Remote-Operations-Realisations
            {joint-iso-ccitt remote-operations(4) realisations(8) version1(0)}

    acse-abstract-syntax
        FROM  Remote-Operations-Abstract-Syntaxes {joint-iso-ccitt remote-operations(4)
            remoteOperationsAabstractSyntaxes(12) version1(0)}

    id-ac-directorySystemAC, id-ac-directorySystemWith2or3seAC, id-rosObject-dspDSA,
    id-contract-dsp,
    id-package-dspConnection, id-package-chainedRead, id-package-chainedSearch,
    id-package-chainedModify, id-as-directorySystemAS
        FROM ProtocolObjectIdentifiers protocolObjectIdentifiers

    dSABind, dSAUnbind, chainedRead, chainedCompare, chainedAbandon, chainedList,
    chainedSearch, chainedAddEntry, chainedRemoveEntry, chainedModifyEntry,
    chainedModifyDN
        FROM DistributedOperations distributedOperations
```

```
        dir2or3se
            FROM DirectorySecurityExchanges  directorySecurityExchanges  ;

-- application contexts --

directorySystemAC  APPLICATION-CONTEXT  ::=  {
        CONTRACT                        dspContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directorySystemAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-directorySystemAC }

directorySystemWith2or3seAC  APPLICATION-CONTEXT  ::=  {
        CONTRACT                        dspContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        directorySystemAbstractSyntax |
                                        dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-directorySystemWith2or3seAC }

-- ROS objects --

dsp-dsa  ROS-OBJECT-CLASS  ::=  {
        BOTH            { dspContract }
        ID              id-rosObject-dspDSA }

-- contracts --

dspContract  CONTRACT  ::=  {
        CONNECTION          dspConnectionPackage
        OPERATIONS OF       { chainedReadPackage | chainedSearchPackage | chainedModifyPackage }
        ID                  id-contract-dsp }

-- connection package --

dspConnectionPackage  CONNECTION-PACKAGE  ::=  {
        BIND        dSABind
        UNBIND      dSAUnbind
        ID          id-package-dspConnection }

--  chained read package --

chainedReadPackage  OPERATION-PACKAGE  ::=  {
        OPERATIONS      { chainedRead | chainedCompare | chainedAbandon }
        ID              id-package-chainedRead }

--  chained search package --

chainedSearchPackage  OPERATION-PACKAGE  ::=  {
        OPERATIONS      { chainedList | chainedSearch }
        ID              id-package-chainedSearch }

--  chained modify package  --

chainedModifyPackage  OPERATION-PACKAGE  ::=  {
        OPERATIONS      { chainedAddEntry | chainedRemoveEntry |
                        chainedModifyEntry | chainedModifyDN }
        ID              id-package-chainedModify }

-- abstract syntaxes --

directorySystemAbstractSyntax  ABSTRACT-SYNTAX  ::=  {
        DSP-PDUs
        IDENTIFIED BY   id-as-directorySystemAS }

DSP-PDUs  ::=  CHOICE  {
        basicRos    ROS { {DSP-InvokeIDSet }, { DSP-Invokable }, { DSP-Returnable } },
        bind        Bind { dSABind },
        unbind      Unbind { dSAUnbind } }
```

```
DSP-InvokeIDSet  ::=  InvokeId (ALL EXCEPT absent:NULL)

DSP-Invokable  OPERATION  ::=    { chainedRead | chainedCompare | chainedAbandon |
                                   chainedList | chainedSearch |
                                   chainedAddEntry | chainedRemoveEntry | chainedModifyEntry |
                                   chainedModifyDN }

DSP-Returnable  OPERATION  ::=   { chainedRead | chainedCompare | chainedAbandon |
                                   chainedList | chainedSearch |
                                   chainedAddEntry | chainedRemoveEntry | chainedModifyEntry |
                                   chainedModifyDN }

END
```

# Annex C

## DISP in ASN.1

(This annex forms an integral part of this Recommendation | International Standard)

This annex includes all of the relevant ASN.1 type and value definitions contained in this Directory Specification in the form of the ASN.1 module, "**DirectoryInformationShadowProtocol**"

---

```
DirectoryInformationShadowProtocol {joint-iso-itu-t ds(5) module(1) disp(16) 3}
DEFINITIONS  ::=
BEGIN

-- EXPORTS All --

-- The types and values defined in this module are exported for use in the other ASN.1 modules contained
-- within the Directory Specifications, and for the use of other applications which will use them to access
-- Directory services. Other applications may use them for their own purposes, but this will not constrain
-- extensions and modifications needed to maintain or improve the Directory service.

IMPORTS
        directoryShadowAbstractService, protocolObjectIdentifiers, directorySecurityExchanges
                FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 3}

        ROS-OBJECT-CLASS, CONTRACT, OPERATION-PACKAGE, CONNECTION-PACKAGE,
        Code, OPERATION
                FROM Remote-Operations-Information-Objects
                        {joint-iso-ccitt remote-operations(4) informationObjects(5) version1(0)}

        ROS{}, Bind{}, Unbind{}, InvokeId
                FROM Remote-Operations-Generic-ROS-PDUs
                        {joint-iso-ccitt remote-operations(4) generic-ROS-PDUs(6) version1(0)}

        APPLICATION-CONTEXT
                FROM Remote-Operations-Information-Objects-extensions {joint-iso-ccitt
                        remote-operations(4) informationObjects-extensions(8) version1(0)}

        acse, pData, association-by-RTSE, transfer-by-RTSE
                FROM Remote-Operations-Realisations
                        {joint-iso-ccitt remote-operations(4) realisations(9) version1(0)}

        acse-abstract-syntax
                FROM Remote-Operations-Abstract-Syntaxes {joint-iso-ccitt remote-operations(4)
                        remoteOperationsAabstractSyntaxes(12) version1(0)}

        id-ac-shadowSupplierInitiatedAC, id-ac-shadowSupplierInitiatedAsynchronousAC,
        id-ac-shadowConsumerInitiatedAC, id-ac-shadowConsumerInitiatedAsynchronousAC,
        id-ac-shadowSupplierInitiatedWith2or3seAC, id-ac-shadowConsumerInitiatedWith2or3seAC,
        id-ac-reliableShadowSupplierInitiatedAC, id-ac-reliableShadowConsumerInitiatedAC,
        id-ac-reliableShadowSupplierInitiatedWith2or3seAC,
        id-ac-reliableShadowConsumerInitiatedWith2or3seAC,
        id-rosObject-initiatingConsumerDSA, id-rosObject-respondingSupplierDSA,
        id-rosObject-initiatingSupplierDSA, id-rosObject-respondingConsumerDSA,
        id-contract-shadowConsumer, id-contract-shadowSupplier, id-package-dispConnection,
        id-package-shadowConsumer, id-package-shadowSupplier, id-as-directoryShadowAS,
        id-as-directoryReliableShadowAS, id-as-reliableShadowBindingAS
                FROM ProtocolObjectIdentifiers protocolObjectIdentifiers

        dSAShadowBind, dSAShadowUnbind, requestShadowUpdate, updateShadow,
        coordinateShadowUpdate
                FROM DirectoryShadowAbstractService directoryShadowAbstractService
```

```
        RTSE-apdus
                FROM Reliable-Transfer-APDUs {joint-iso-ccitt reliable-transfer(3) apdus(0)}

        dir2or3se
                FROM DirectorySecurityExchanges  directorySecurityExchanges        ;

-- application contexts --

shadowSupplierInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowSupplierInitiatedAC }

shadowSupplierInitiatedAsynchronousAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               {acse-abstract-syntax |
                                        directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowSupplierInitiatedAsynchronousAC }

shadowSupplierInitiatedWith2or3seAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax | dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-shadowSupplierInitiatedWith2or3seAC }

shadowConsumerInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax | directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowConsumerInitiatedAC }

shadowConsumerInitiatedAsynchronousAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               {acse-abstract-syntax |
                                        directoryShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-shadowConsumerInitiatedAsynchronousAC }

shadowConsumerInitiatedWith2or3seAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  acse
        INFORMATION TRANSFER BY         pData
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        directoryShadowAbstractSyntax |
                                        dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-shadowConsumerInitiatedWith2or3seAC }

reliableShadowSupplierInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowSupplierInitiatedAC }
```

```
reliableShadowSupplierInitiatedWith2or3seAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowSupplierContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax |
                                        dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowSupplierInitiatedWith2or3seAC }

reliableShadowConsumerInitiatedAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowConsumerInitiatedAC }

reliableShadowConsumerInitiatedWith2or3seAC  APPLICATION-CONTEXT  ::= {
        CONTRACT                        shadowConsumerContract
        ESTABLISHED BY                  association-by-RTSE
        INFORMATION TRANSFER BY         transfer-by-RTSE
        ABSTRACT SYNTAXES               { acse-abstract-syntax |
                                        reliableShadowBindingAbstractSyntax |
                                        directoryReliableShadowAbstractSyntax |
                                        dir2or3se }
        APPLICATION CONTEXT NAME        id-ac-reliableShadowConsumerInitiatedWith2or3seAC }

-- ROS objects --

initiating-consumer-dsa  ROS-OBJECT-CLASS  ::= {
        INITIATES       { shadowConsumerContract }
        ID              id-rosObject-initiatingConsumerDSA }

responding-supplier-dsa  ROS-OBJECT-CLASS  ::= {
        RESPONDS        { shadowConsumerContract }
        ID              id-rosObject-respondingSupplierDSA }
initiating-supplier-dsa  ROS-OBJECT-CLASS  ::= {
        INITIATES       { shadowSupplierContract }
        ID              id-rosObject-initiatingSupplierDSA }

responding-consumer-dsa  ROS-OBJECT-CLASS  ::= {
        RESPONDS        { shadowSupplierContract }
        ID              id-rosObject-respondingConsumerDSA }

-- contracts --

shadowConsumerContract  CONTRACT  ::= {
        CONNECTION                      dispConnectionPackage
        INITIATOR CONSUMER OF           { shadowConsumerPackage }
        ID                              id-contract-shadowConsumer }

shadowSupplierContract  CONTRACT  ::= {
        CONNECTION                      dispConnectionPackage
        RESPONDER CONSUMER OF           { shadowSupplierPackage }
        ID                              id-contract-shadowSupplier }

-- connection package --

dispConnectionPackage  CONNECTION-PACKAGE  ::= {
        BIND                    dSAShadowBind
        UNBIND                  dSAShadowUnbind
        ID                      id-package-dispConnection }

-- packages --

shadowConsumerPackage  OPERATION-PACKAGE  ::= {
        CONSUMER INVOKES        { requestShadowUpdate }
        SUPPLIER INVOKES        { updateShadow }
        ID                      id-package-shadowConsumer }
```

```
shadowSupplierPackage  OPERATION-PACKAGE  ::= {
        SUPPLIER INVOKES      { coordinateShadowUpdate |
                                updateShadow }
        ID                    id-package-shadowSupplier }

-- abstract syntaxes --

directoryShadowAbstractSyntax  ABSTRACT-SYNTAX  ::= {
        DISP-PDUs
        IDENTIFIED BY     id-as-directoryShadowAS }

directoryReliableShadowAbstractSyntax  ABSTRACT-SYNTAX  ::= {
        Reliable-DISP-PDUs
        IDENTIFIED BY     id-as-directoryReliableShadowAS }

reliableShadowBindingAbstractSyntax     ABSTRACT-SYNTAX  ::= {
        ReliableShadowBinding-PDUs
        IDENTIFIED BY     id-as-reliableShadowBindingAS }

DISP-PDUs  ::=  CHOICE  {
        basicROS    ROS { { DISP-InvokeIDSet }, { DISP-Invokable }, {DISP-Returnable } },
        bind        Bind { dSAShadowBind },
        unbind      Unbind { dSAShadowUnbind } }

Reliable-DISP-PDUs      ::=     ROS { { DISP-InvokeIDSet }, { DISP-Invokable },
        {DISP-Returnable } }

ReliableShadowBinding-PDUs  ::=  CHOICE  {
        rTS         [0] RTSE-apdus,
        bind        Bind { dSAShadowBind },
        unbind      Unbind { dSAShadowUnbind } }

DISP-InvokeIDSet  ::=  InvokeID (ALL EXCEPT absent:NULL)

DISP-Invokable  OPERATION  ::=    { requestShadowUpdate | updateShadow |
                                  coordinateShadowUpdate }

DISP-Returnable  OPERATION  ::=   { requestShadowUpdate | updateShadow |
                                  coordinateShadowUpdate }

--  remote operation codes --

id-opcode-requestShadowUpdate       Code        ::=     local : 1
id-opcode-updateShadow              Code        ::=     local : 2
id-opcode-coordinateShadowUpdate    Code        ::=     local : 3

-- remote error codes  --

id-errcode-shadowError              Code        ::=     local : 1

END
```