INTERNATIONAL STANDARD

ISO/IEC 9594-5

Second edition
1995-09-15

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 9594-5:1995(E)

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.

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.

International Standard ISO/IEC 9594-5 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 21, *Open systems interconnection, data management and open distributed processing*, in collaboration with ITU-T. The identical text is published as ITU-T Recommendation X.519.

Implementors should note that a defect resolution process exists and that corrections may be applied to this part of ISO/IEC 9594 in the form of technical corrigenda. A list of approved technical corrigenda for this part of ISO/IEC 9594 can be obtained from the subcommittee secretariat. Published technical corrigenda are available from your national standards organization.

This second edition technically revises and enhances ISO/IEC 9594-5:1990. It also incorporates technical corrigendum 1:1992. Implementations may still claim conformance to the first edition of this part of ISO/IEC 9594. However, at some point, the first edition will no longer be supported (i.e. reported defects will no longer be resolved). It is recommended that implementations conform to this second edition as soon as possible.

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*

Annexes A to F form an integral part of this part of ISO/IEC 9594. Annex G 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 second edition technically revises and enhances, but does not replace, the first edition of this Recommendation | International Standard. Implementations may still claim conformance to the first edition.

This second edition specifies version 1 of the Directory service and protocols. The first edition also specifies version 1. Differences between the services and between the protocols defined in the two editions are accommodated using the rules of extensibility defined in this edition of this Recommendation | International Standard.

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 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 CCITT 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 editions 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.500 (1993) | ISO/IEC 9594-1:1995, *Information technology – Open Systems Interconnection – The Directory: Overview of concepts, models and services.*
- ITU-T Recommendation X.501 (1993) | ISO/IEC 9594-2:1995, *Information technology – Open Systems Interconnection – The Directory: Models.*
- ITU-T Recommendation X.511 (1993) | ISO/IEC 9594-3:1995, *Information technology – Open Systems Interconnection – The Directory: Abstract service definition.*
- ITU-T Recommendation X.518 (1993) | ISO/IEC 9594-4:1995, *Information technology – Open Systems Interconnection – The Directory: Procedures for distributed operation.*
- ITU-T Recommendation X.520 (1993) | ISO/IEC 9594-6:1995, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types.*
- ITU-T Recommendation X.521 (1993) | ISO/IEC 9594-7:1995, *Information technology – Open Systems Interconnection – The Directory: Selected object classes.*
- ITU-T Recommendation X.509 (1993) | ISO/IEC 9594-8:1995, *Information technology – Open Systems Interconnection – The Directory: Authentication framework.*
- ITU-T Recommendation X.525 (1993) | ISO/IEC 9594-9:1995, *Information technology – Open Systems Interconnection – The Directory: Replication*
- ITU-T Recommendation X.680 (1994) | ISO/IEC 8824-1:1995, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*
- ITU-T Recommendation X.681 (1994) | ISO/IEC 8824-2:1995, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.*
- ITU-T Recommendation X.682 (1994) | ISO/IEC 8824-3:1995, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.*
- ITU-T Recommendation X.683 (1994) | ISO/IEC 8824-4:1995, *Information technology – Abstract Syntax Notation One (ASN.1): Parametrization of ASN.1 specifications.*

- ITU-T Recommendation X.690 (1994) | ISO/IEC 8825-1:1995, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*
- ITU-T Recommendation X.880 (1994) | ISO/IEC 13712-1:1995, *Information technology – Remote Operations: Concepts, model and notation.*
- 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.882 (1994) | ISO/IEC 13712-3:1995, *Information technology – Remote Operations: OSI realizations – Remote Operations Service Element (ROSE) protocol specification.*

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.200 (1988), *Reference Model of Open Systems Interconnection for CCITT Applications.*

  ISO 7498:1984, *Information processing systems — Open Systems Interconnection — Basic Reference Model.*
- CCITT Recommendation X.216 (1988), *Presentation Service Definition for Open Systems Interconnection for CCITT Applications.*

  ISO 8822:1988, *Information processing systems – Open Systems Interconnection – Presentation service definition.*
- CCITT Recommendation X.217 (1988), *Service Definition for the Association Control Service Element.*

  ISO 8649:1988, *Information processing systems – Open Systems Interconnection – Service Definition for the Association Control Service Element.*
- CCITT Recommendation X.218 (1988) *Reliable Transfer: Model and Service Definition.*

  ISO/IEC 9066-1:1989, *Information processing systems – Text communication – Reliable Transfer: Model and service definition.*
- CCITT Recommendation X.227 (1988), *Protocol Specification for the Association Control Service Element.*

  ISO 8650:1988, *Information processing systems – Open Systems Interconnection – Protocol specification for the Association Control Service Element.*

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 CCITT Rec. X.200 | ISO 7498:

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 9072-1:

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*.

4 Abbreviations

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

AC	Application Context
ACSE	Association Control Service Element
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
ROS	Remote Operations Service
ROSE	Remote Operations Service Element

5 Conventions

With minor exceptions this Directory Specification has been prepared according to the "Presentation of ITU-TS/ISO/IEC common text" guidelines in the Guide for ITU-TS and ISO/IEC JTC 1 Cooperation, March 1993.

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 previous (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. Systems conforming to the current Directory Specifications are referred to as "1993 edition systems".

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 9072-1.

6 Protocol overview

6.1 Remote Operations – Specification and OSI Realization

ITU-T Rec. X.880 | ISO/IEC 9072-1 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 9072-1 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 9072-1 are mapped to OSI concepts in ITU-T Rec. X.881 | ISO/IEC 9072-2 and ITU-T Rec. X.882 | ISO/IEC 9072-3.

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.

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 modeled, 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.

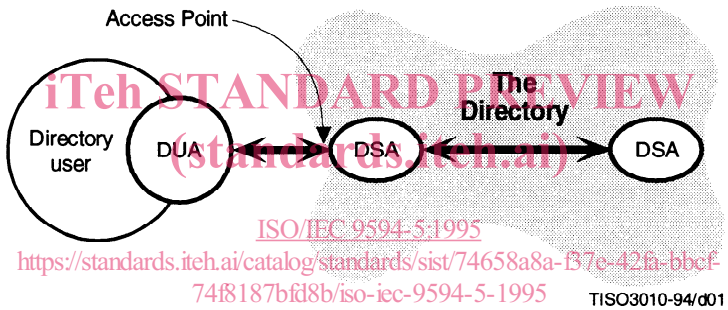

Figure 1 – Directory Interactions

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 9072-2.

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 CCITT 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 CCITT Rec. X.217 | ISO 8649.

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

If the RTSE 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 is 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.

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.