

This Standard is obsolescent.
See the BSI Catalogue for details.

BS 7299 : Part 5 : 1990
ISO 8613-5 : 1989

British Standard

Office document architecture (ODA) and interchange format for text and office systems

Part 5. Office document interchange format (ODIF)

Architecture des documents de bureau (ODA) et format d'échange pour la bureautique
Partie 5. Format d'échange de documents de bureau

Büro Dokumenten-Architektur und Austauschformat für Text- und Bürosysteme
Teil 5. Büro-Dokumenten-Austauschformat

Contents

	Page
National foreword	Inside front cover
Committees responsible	Back cover
1 Scope	1
2 Normative references	2
3 Definitions	2
4 Document representations	3
5 Office document interchange format (ODIF)	3
Annexes	
A Coded representation	24
B Application class tag assignments	26
C Summary of object identifiers	27
D Examples	28
E Office document language (ODL)	58
F Examples of office document language representations	81

National foreword

This Part of BS 7299 has been prepared under the direction of the Information Systems Technology Standards Policy Committee. It is the UK implementation of, and is identical with, ISO 8613-5 : 1989 'Information processing – Text and office systems – Office Document Architecture (ODA) and interchange format – Part 5 : Office document interchange format (ODIF)', published by the International Organization for Standardization (ISO).

The following list includes all the Parts of this British Standard.

- Part 1 Introduction and general principles
- Part 2 Document structures
- Part 4 Document profile
- Part 5 Office document interchange format (ODIF)
- Part 6 Character content architectures
- Part 7 Raster graphics content architectures
- Part 8 Geometric graphics content architectures

NOTE. To maintain alignment with the numbering of the Parts of ISO 8613, there is at present no Part 3 of this British Standard.

The UK actively participated in the development of the international standard through BSI Technical Committee IST/18 and fully approved publication of the international standard.

There are no plans at present to supplement this British Standard with any additional guidance, and any comments or enquiries on the international standard or its application should be addressed to the Secretary of IST/18 at the BSI headquarters address.

Reasons for implementation. This Part of BS 7299 implements the international standard for the following reasons.

- (a) Because information technology equipment, services and systems tend to be, and generally need to be, designed, purchased, and used in a world-wide context, BSI policy is to agree, wherever possible, standards at international level and not to prepare unique national standards. At the time of its approval, the text of the international standard was considered to be the best that could be achieved, taking into account both the needs of the 'market place' and the 'state of the art' in this subject area, and the international standardization process.
- (b) The Commission of the European Communities has mandated the European Committee for Standardization (CEN) to produce European Standards (ENs) (or, where this cannot be achieved, Harmonization Documents (HDs) or Draft European Standards (ENVs)) in this subject area for application throughout the European Community by 1992. This British Standard anticipates that ISO 8613-5 : 1989 will be adopted without change as the Reference Document for the EN (or HD or ENV) and, therefore, fulfils in advance the UK obligation under CEN/CENELEC Common Rules to implement nationally each agreed European Standard.
- (c) Publication of British Standards corresponding to the international standards in which IST/- resources were invested will ensure that the BSI Catalogue becomes, and then remains, a comprehensive first source of information for UK users of IT equipment and services about the availability of international standards on IT subjects. In turn, this will facilitate UK use of, and access to, those standards.

Cross-references. The cross-references relating to all Parts of this British Standard are given in the national foreword to Part 1.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Information processing — Text and office systems — Office Document Architecture (ODA) and interchange format —

Part 5 : Office Document Interchange Format (ODIF)

1 Scope

The purpose of ISO 8613 is to facilitate the interchange of documents.

In the context of ISO 8613, documents are considered to be items such as memoranda, letters, invoices, forms and reports, which may include pictures and tabular material. The content elements used within the documents may include graphic characters, geometric graphics elements and raster graphics elements, all potentially within one document.

NOTE – ISO 8613 is designed to allow for extensions, including typographical features, colour, spreadsheets and additional types of content such as sound.

ISO 8613 applies to the interchange of documents by means of data communications or the exchange of storage media.

ISO 8613 provides for the interchange of documents for either or both of the following purposes:

- to allow presentation as intended by the originator;
- to allow processing such as editing and reformatting.

The composition of a document in interchange can take several forms:

- formatted form, allowing presentation of the document;
- processable form, allowing processing of the document;
- formatted processable form, allowing both presentation and processing.

ISO 8613 also provides for the interchange of ODA information structures used for the processing of interchanged documents.

Furthermore, ISO 8613 allows for the interchange of documents containing one or more different types of content such as character text, images, graphics and sound.

This part of ISO 8613 defines

- the format of the data stream used to interchange documents structured in accordance with ISO 8613-2;
- the representation of the constituents which may appear in an interchanged document.

NOTES

- 1 This part of ISO 8613 does not specify the coded representation of content elements.
- 2 Data formats for presentation attributes and coding attributes are defined in other parts of ISO 8613.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO 8613. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO 8613 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 8601 : 1988, *Data elements and interchange formats – Information interchange – Representation of dates and times*.

ISO 8613 : 1989, *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format –*

Part 1 – Introduction and general principles;

Part 2 – Document structures;

Part 4 – Document profile;

Part 6 – Character content architectures;

Part 7 – Raster graphics content architectures;

Part 8 – Geometric graphics content architectures.

ISO 8824 : 1987, *Information processing – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*.

ISO 8824 Add.1 : ¹⁾, *Information processing systems – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1) – Addendum 1 : ASN.1 extensions*.

ISO 8825 : 1987, *Information processing – Open Systems Interconnection – Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1)*.

ISO 8825 Add.1 : ¹⁾, *Information processing systems – Open Systems Interconnection – Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1) – Addendum 1 : ASN.1 extensions*.

ISO 8879 : 1986, *Information processing – Text and office systems – Standard Generalized Markup Language (SGML)*.

ISO 9069 : 1988, *Information processing – SGML support facilities – SGML Document Interchange Format (SDIF)*.

ISO 9541-6 : ¹⁾, *Information processing – Font and character information interchange – Part 6: Font and character attribute subsets and applications*.

3 Definitions

For the purpose of this part of ISO 8613, the definitions given in ISO 8613-1 apply.

¹⁾ To be published.

4 Document representations

A document structured in accordance with ISO 8613 is represented for interchange by either the Office Document Interchange Format (ODIF), or the Office Document Language (ODL) in conjunction with the SGML Document Interchange Format (SDIF). The ODIF and ODL/SDIF representations are technically equivalent; a document can be transformed from one to the other without loss of information about the document constituents and attributes.

NOTE – Both data structure (ODIF) and language (ODL) representations have been standardized in order to meet the document representation requirements of distinct application environments. ODIF, being a data structure specified using ASN.1, is particularly intended for use in an OSI environment. ODL is particularly appropriate for systems that share information through marked-up text files, especially where human users can access the markup directly.

4.1 ODIF

ODIF is an abstract data syntax in which the constituents and attributes of the document are represented by a hierarchy of data structures and data items, specified using the abstract syntax notation ASN.1 defined in ISO 8824.

The coded representation of each data structure or data item is obtained by applying a set of encoding rules.

ODIF is specified in clause 5.

NOTE – ASN.1 is a formal description method that allows data types relevant to an application to be specified in terms of other data types, including basic data types such as “integer” and “octet string” which are defined in ISO 8824 itself. Basic encoding rules for ASN.1 are defined in ISO 8825 and are summarized in annex A.

4.2 ODL and SDIF

ODL is a language in which the constituents and attributes of the document are identified by descriptive tags, and are grouped into one or more storage entities (e.g. files) as the user may require.

For interchange, each ODL entity is represented as a single data structure or data item, specified using ASN.1, in a data stream constructed according to the SGML Document Interchange Format defined in ISO 9069.

ODL is specified in annex E (normative).

NOTE – ODL is an SGML application conforming to ISO 8879.

5 Office Document Interchange Format (ODIF)

5.1 General description

A document structured in accordance with ISO 8613 is represented by a data stream consisting of one or more data structures of the following types:

- document profile descriptor;
- layout object descriptor;
- layout object class descriptor;
- logical object descriptor;
- logical object class descriptor;
- presentation style descriptor;
- layout style descriptor;
- text unit.

These data structures are called *interchange data elements*. Within a data stream, the interchange data elements are ordered in accordance with certain rules which are specified below. This part of ISO 8613 defines two such sets of rules; they are called *interchange format class A* and *interchange format class B*.

Which of these sets of rules applies to a given data stream is indicated in the document profile descriptor. In all cases, a data stream contains one and only one document profile descriptor which is always the first interchange data element in the data stream. The document profile descriptor may be the only data structure in the data stream.

When an ODIF data stream is used as part of an ASN.1 external data type, the abstract syntax shall be formed by an ASN.1 SEQUENCE-OF type referencing the Interchange-Data-Element type; the encoding of the data value shall consist of an integral number of octets, formed by applying the ASN.1 basic encoding rules; and the value of the associated ASN.1 object identifier shall be { 2 8 0 0 }.

NOTE – The manner of incorporating the interchange data elements, or the external data type, in an application protocol or the manner of mapping them on service data units (in an OSI environment) is not defined by this part of ISO 8613.

5.2 Interchange format class A

According to interchange format class A, a data stream consists of one document profile descriptor and, optionally, one or more interchange data elements of the following types:

- layout object descriptor;
- layout object class descriptor;
- logical object descriptor;
- logical object class descriptor;
- presentation style descriptor;
- layout style descriptor;
- text unit.

The order of the interchange data elements is as follows:

- a) document profile descriptor;
- b) layout object class descriptors;
- c) logical object class descriptors;
- d) text units representing generic content portions;
- e) presentation style descriptors;
- f) layout style descriptors;
- g) layout object descriptors;
- h) logical object descriptors;
- i) text units representing specific content portions.

Within each of the groups of layout object descriptors and logical object descriptors, the order of the descriptors is equal to the sequential order defined in ISO 8613–2.

If the data stream contains layout object descriptors, the text units representing specific content portions are ordered according to the sequential layout order; otherwise, they are ordered according to the sequential logical order.

Within each of the other groups of interchange data elements, the order is arbitrary.

5.3 Interchange format class B

According to interchange format class B, a data stream consists of one document profile descriptor and, optionally, one or more interchange data elements of the following types:

- layout object descriptor;
- layout object class descriptor;
- presentation style descriptor;
- text unit.

Interchange format class B can be used only to represent documents that do not contain any specific or generic logical structure, i.e. documents that conform to the formatted document architecture class.

The order of the interchange data elements is as follows:

- a) document profile descriptor;
- b) layout object class descriptors and associated text units;
- c) presentation style descriptors;
- d) layout object descriptors and associated text units.

Within the group of layout object class descriptors and associated text units, the order is such that a group of descriptors that have identical identifiers, except for the last number in each identifier, follow each other in the data stream without any other descriptor between them. However, each descriptor of an object class for a basic layout object is followed immediately by the associated text units.

Within the group of layout object descriptors and associated text units, the order of the descriptors is equal to the sequential order defined in ISO 8613–2. However, each descriptor of a basic layout object is followed immediately by the associated text units.

Within the group of presentation style descriptors, the order is arbitrary.

5.4 Descriptors and text units

A document profile descriptor, layout object descriptor, layout object class descriptor, logical object descriptor, logical object class descriptor, presentation style descriptor or layout style descriptor consists of simple and composite data items representing the attributes of the constituent concerned.

The document profile, each object class, each style and each object is represented by one descriptor.

A text unit consists of two parts:

- a) an attribute field, i.e. a data structure consisting of simple and composite data items representing the attributes of the content portion concerned;
- b) an information field, i.e. a data structure that is either a data item or a set of data items representing the content elements making up the content portion concerned.

Each content portion is represented by one text unit.

The data formats of the interchange data elements are specified in 5.5 to 5.12, using the abstract syntax notation ASN.1 defined in ISO 8824.

NOTE – 5.5 to 5.12 by themselves do not completely define the data stream format; additional rules are specified in 5.1 to 5.4 of this part and in other parts of ISO 8613. For example, the keyword OPTIONAL merely indicates that a particular data structure or data item is not part of every instance of the containing data structure; the conditions controlling the presence or absence of the data structure or data item are specified in part 2 or part 4.

5.5 Interchange data elements

Interchange-Data-Elements { 2 8 1 5 5 }

DEFINITIONS ::= BEGIN

EXPORTS Interchange-Data-Element;

IMPORTS Document-Profile-Descriptor
 FROM Document-Profile-Descriptor - - see 5.6
 Layout-Class-Descriptor, Layout-Object-Descriptor
 FROM Layout-Descriptors - - see 5.8
 Logical-Class-Descriptor, Logical-Object-Descriptor
 FROM Logical-Descriptors - - see 5.9
 Presentation-Style-Descriptor, Layout-Style-Descriptor
 FROM Style-Descriptors - - see 5.10
 Text-Unit
 FROM Text-Units; - - see 5.12

Interchange-Data-Element ::= CHOICE {
 document-profile [0] IMPLICIT Document-Profile-Descriptor,
 layout-object-class [1] IMPLICIT Layout-Class-Descriptor,
 layout-object [2] IMPLICIT Layout-Object-Descriptor,
 content-portion [3] IMPLICIT Text-Unit,
 logical-object-class [5] IMPLICIT Logical-Class-Descriptor,
 logical-object [6] IMPLICIT Logical-Object-Descriptor,
 presentation-style [7] IMPLICIT Presentation-Style-Descriptor,
 layout-style [8] IMPLICIT Layout-Style-Descriptor}
 END

5.6 Document profile descriptor

Document-Profile-Descriptor { 2 8 1 5 6 }

DEFINITIONS ::= BEGIN

EXPORTS Document-Profile-Descriptor;

IMPORTS Resource-Name, Object-or-Class-Identifier
 FROM Identifiers-and-Expressions - - see 5.7
 Measure-Pair, Transparency, Colour, Dimension-Pair, One-Of-Four-Angles,
 Border, Medium-Type
 FROM Layout-Descriptors - - see 5.8
 Protection
 FROM Logical-Descriptors - - see 5.9
 Content-Architecture-Class, Content-Type, Block-Alignment, Fill-Order
 FROM Style-Descriptors - - see 5.10
 Type-of-Coding
 FROM Text-Units - - see 5.12
 Character-Content-Defaults, Char-Presentation-Feature,
 Character-Coding-Attribute
 FROM Character-Profile-Attributes { 2 8 1 6 4 } - - see ISO 8613-6
 Raster-Gr-Content-Defaults, Ra-Gr-Presentation-Feature,
 Ra-Gr-Coding-Attribute
 FROM Raster-Gr-Profile-Attributes { 2 8 1 7 4 } - - see ISO 8613-7
 Geo-Gr-Content-Defaults, Geo-Gr-Presentation-Feature,
 Geo-Gr-Coding-Attribute
 FROM Geo-Gr-Profile-Attributes { 2 8 1 8 4 } - - see ISO 8613-8
 Font-Reference FROM ISO9541-FONTS { 1 9541 6 1 0 }; - - see ISO 9541-6

Document-Profile-Descriptor ::= SET {
 generic-layout-structure [0] IMPLICIT NumericString OPTIONAL,
 specific-layout-structure [1] IMPLICIT NumericString OPTIONAL,
 generic-logical-structure [4] IMPLICIT NumericString OPTIONAL,
 specific-logical-structure [5] IMPLICIT NumericString OPTIONAL,
 presentation-styles [6] IMPLICIT NumericString OPTIONAL,
 layout-styles [7] IMPLICIT NumericString OPTIONAL,

- - for the generic structures;
- - 'partial generator-set' is represented by "0", 'complete-generator-set'
- - is represented by "1", 'factor-set' is represented by "2";
- - for the other cases, the numeric string has the value 'present'
- - represented by "1"

external-document-class [9] Document-Reference OPTIONAL,
 resource-document [10] Document-Reference OPTIONAL,
 resources [11] IMPLICIT SET OF SET {
 resource-identifier Resource-Name,
 object-class-identifier Object-or-Class-Identifier} OPTIONAL,
 document-characteristics [2] IMPLICIT Document-Characteristics OPTIONAL,
 document-management-attributes [3] IMPLICIT Document-Management-Attributes OPTIONAL}

Document-Characteristics ::= SET {
 document-application-profile CHOICE {
 [0] IMPLICIT INTEGER {
 group-4-facsimile (2)},
 [4] IMPLICIT OBJECT IDENTIFIER} OPTIONAL,
 doc-appl-profile-defaults [10] IMPLICIT Doc-Appl-Profile-Defaults OPTIONAL,
 document-architecture-class [1] IMPLICIT INTEGER {
 formatted (0),
 processable (1),
 formatted-processable (2)} OPTIONAL,
 content-architecture-classes [5] IMPLICIT SET OF OBJECT IDENTIFIER OPTIONAL,

interchange-format-class	[6] IMPLICIT INTEGER { if-a (0), if-b (1)} OPTIONAL,
oda-version	[8] IMPLICIT SEQUENCE {
standard-or-recommendation	Character-Data,
publication-date	Date-and-Time} OPTIONAL,
non-basic-doc-characteristics	[2] IMPLICIT Non-Basic-Doc-Characteristics OPTIONAL,
non-basic-struc-characteristics	[3] IMPLICIT Non-Basic-Struc-Characteristics OPTIONAL,
additional-doc-characteristics	[9] IMPLICIT Additional-Doc-Characteristics OPTIONAL}
Doc-Appl-Profile-Defaults	::= SET {
document-architecture-defaults	[0] IMPLICIT Document-Architecture-Defaults OPTIONAL,
character-content-defaults	[1] IMPLICIT Character-Content-Defaults OPTIONAL,
raster-gr-content-defaults	[2] IMPLICIT Raster-Gr-Content-Defaults OPTIONAL,
geo-gr-content-defaults	[3] IMPLICIT Geo-Gr-Content-Defaults OPTIONAL,
- - the following tags are reserved for additional types	
- - of content defaults:	
- - [4] videotex, for use in conjunction with CCITT Recommendations	
- - [5] audio	
- - [6] dynamic-graphics	
external-content-architecture-defaults	[7] IMPLICIT SEQUENCE OF EXTERNAL OPTIONAL }
Document-Architecture-Defaults	::= SET {
content-architecture-class	[0] IMPLICIT Content-Architecture-Class OPTIONAL,
content-type	[1] IMPLICIT Content-Type OPTIONAL,
page-dimensions	[2] IMPLICIT Measure-Pair OPTIONAL,
transparency	[3] IMPLICIT Transparency OPTIONAL,
colour	[4] IMPLICIT Colour OPTIONAL,
layout-path	[5] IMPLICIT One-Of-Four-Angles OPTIONAL,
medium-type	[6] IMPLICIT Medium-Type OPTIONAL,
block-alignment	[7] IMPLICIT Block-Alignment OPTIONAL,
border	[8] IMPLICIT Border OPTIONAL,
page-position	[9] IMPLICIT Measure-Pair OPTIONAL,
type-of-coding	[10] Type-of-Coding OPTIONAL}
Non-Basic-Doc-Characteristics	::= SET {
profile-character-sets	[5] IMPLICIT OCTET STRING OPTIONAL,
comments-character-sets	[1] IMPLICIT OCTET STRING OPTIONAL,
alternative-repr-char-sets	[6] IMPLICIT OCTET STRING OPTIONAL,
- - each of these octet strings represents a string of escape sequences	
page-dimensions	[2] IMPLICIT SET OF Dimension-Pair OPTIONAL,
medium-types	[8] IMPLICIT SET OF Medium-Type OPTIONAL,
layout-paths	[21] IMPLICIT SET OF One-Of-Four-Angles OPTIONAL,
transparencies	[22] IMPLICIT SET OF Transparency OPTIONAL,
protections	[23] IMPLICIT SET OF Protection OPTIONAL,
block-alignments	[24] IMPLICIT SET OF Block-Alignment OPTIONAL,
fill-orders	[25] IMPLICIT SET OF Fill-Order OPTIONAL,
colours	[26] IMPLICIT SET OF Colour OPTIONAL,
borders	[27] IMPLICIT SET OF Border OPTIONAL,
page-positions	[28] IMPLICIT SET OF Measure-Pair OPTIONAL,
types-of-coding	[29] IMPLICIT SET OF Type-of-Coding OPTIONAL,
char-presentation-features	[9] IMPLICIT SET OF Char-Presentation-Feature OPTIONAL,
ra-gr-presentation-features	[4] IMPLICIT SET OF Ra-Gr-Presentation-Feature OPTIONAL,
geo-gr-presentation-features	[12] IMPLICIT SET OF Geo-Gr-Presentation-Feature OPTIONAL,

- - the following tags are reserved for additional types
- - of presentation features:
 - - [13] videotex, for use in conjunction with CCITT Recommendations
 - - [14] audio
 - - [15] dynamic-graphics

character-coding-attributes	[16] IMPLICIT SET OF Character-Coding-Attribute OPTIONAL,
ra-gr-coding-attributes	[3] IMPLICIT SET OF Ra-Gr-Coding-Attribute OPTIONAL,
geo-gr-coding-attributes	[17] IMPLICIT SET OF Geo-Gr-Coding-Attribute OPTIONAL,

- - the following tags are reserved for additional types
- - of coding attributes:
 - - [18] videotex, for use in conjunction with CCITT Recommendations
 - - [19] audio
 - - [20] dynamic-graphics

ext-non-basic-pres-features	[10] IMPLICIT SEQUENCE OF EXTERNAL OPTIONAL,
ext-non-basic-coding-attributes	[11] IMPLICIT SEQUENCE OF EXTERNAL OPTIONAL}

Non-Basic-Struc-Characteristics	::= SET {
number-of-objects-per-page	[0] IMPLICIT INTEGER OPTIONAL}

Additional-Doc-Characteristics	::= SET {
unit-scaling	[3] IMPLICIT SEQUENCE {INTEGER,INTEGER} OPTIONAL,
fonts-list	[2] IMPLICIT Fonts-List OPTIONAL}

Fonts-List	::= SET OF SET {
font-identifier	INTEGER,
font-reference	Font-Reference}

Document-Management-Attributes	::= SET {
document-description	[7] IMPLICIT Document-Description OPTIONAL,
dates-and-times	[0] IMPLICIT Dates-and-Times OPTIONAL,
originators	[1] IMPLICIT Originators OPTIONAL,
other-user-information	[2] IMPLICIT Other-User-Information OPTIONAL,
external-references	[3] IMPLICIT External-References OPTIONAL,
local-file-references	[4] IMPLICIT Local-File-References OPTIONAL,
content-attributes	[5] IMPLICIT Content-Attributes OPTIONAL,
security-information	[6] IMPLICIT Security-Information OPTIONAL}

Document-Description	::= SET {
title	[0] IMPLICIT Character-Data OPTIONAL,
subject	[1] IMPLICIT Character-Data OPTIONAL,
document-type	[2] IMPLICIT Character-Data OPTIONAL,
abstract	[3] IMPLICIT Character-Data OPTIONAL,
keywords	[4] IMPLICIT SET OF Character-Data OPTIONAL,
document-reference	[5] Document-Reference OPTIONAL}

Character-Data	::= [APPLICATION 3] IMPLICIT OCTET STRING
----------------	---

- - string of characters from the sets designated by the attribute
- - "profile character sets", plus carriage return and line feed

Document-Reference	::= CHOICE {
unique-reference	OBJECT IDENTIFIER,
descriptive-reference	Character-Data}

Dates-and-Times document-date-and-time creation-date-and-time local-filing-date-and-time expiry-date-and-time start-date-and-time purge-date-and-time release-date-and-time revision-history revision-date-and-time version-number revisors names position organization version-reference user-comments	::= SET { [0] IMPLICIT Date-and-Time OPTIONAL, [1] IMPLICIT Date-and-Time OPTIONAL, [2] IMPLICIT SEQUENCE OF Date-and-Time OPTIONAL, [3] IMPLICIT Date-and-Time OPTIONAL, [4] IMPLICIT Date-and-Time OPTIONAL, [5] IMPLICIT Date-and-Time OPTIONAL, [6] IMPLICIT Date-and-Time OPTIONAL, [7] IMPLICIT SEQUENCE OF SET { [0] IMPLICIT Date-and-Time OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL, [2] IMPLICIT SET OF SET { [0] IMPLICIT SET OF Personal-Name OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL, [2] IMPLICIT Character-Data OPTIONAL} OPTIONAL, [3] Document-Reference OPTIONAL, [4] IMPLICIT Character-Data OPTIONAL} OPTIONAL}
Date-and-Time - - string of characters representing a date and, optionally, a time - - in accordance with ISO 8601	::= [APPLICATION 4] IMPLICIT PrintableString
Originators organizations preparers personal-name organization owners personal-name organization authors personal-name organization	::= SET { [0] IMPLICIT SET OF Character-Data OPTIONAL, [1] IMPLICIT SEQUENCE OF SET { [0] IMPLICIT Personal-Name OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL} OPTIONAL, [2] IMPLICIT SEQUENCE OF SET { [0] IMPLICIT Personal-Name OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL} OPTIONAL, [3] IMPLICIT SEQUENCE OF SET { [0] IMPLICIT Personal-Name OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL} OPTIONAL}
Personal-Name surname givenname initials title	::= [APPLICATION 6] IMPLICIT SET { [0] IMPLICIT Character-Data, [1] IMPLICIT Character-Data OPTIONAL, [2] IMPLICIT Character-Data OPTIONAL, [3] IMPLICIT Character-Data OPTIONAL}
Other-User-Information copyright copyright-information copyright-dates status user-specific-codes distribution-list personal-name organization additional-information	::= SET { [0] IMPLICIT SET OF SET { [0] IMPLICIT SET OF Character-Data OPTIONAL, [1] IMPLICIT SET OF Date-and-Time OPTIONAL} OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL, [2] IMPLICIT SET OF Character-Data OPTIONAL, [3] IMPLICIT SEQUENCE OF SET { [0] IMPLICIT Personal-Name OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL} OPTIONAL, [5] ANY OPTIONAL}
External-References references-to-other-documents superseded-documents	::= SET { [0] IMPLICIT SET OF Document-Reference OPTIONAL, [1] IMPLICIT SET OF Document-Reference OPTIONAL}
Local-File-References file-name location user-comments	::= SET OF SET { [0] IMPLICIT Character-Data OPTIONAL, [1] IMPLICIT Character-Data OPTIONAL, [2] IMPLICIT Character-Data OPTIONAL}
Content-Attributes document-size number-of-pages languages	::= SET { [1] IMPLICIT INTEGER OPTIONAL, [2] IMPLICIT INTEGER OPTIONAL, [4] IMPLICIT SET OF Character-Data OPTIONAL}

Security-Information
 authorization
 person
 organization
 security-classification
 access-rights

```
 ::= SET {
    CHOICE {
      [0] IMPLICIT Personal-Name,
      [4] IMPLICIT Character-Data} OPTIONAL,
      [1] IMPLICIT Character-Data OPTIONAL,
      [2] IMPLICIT SET OF Character-Data OPTIONAL}
  END
```

5.7 Identifiers and expressions

Identifiers-and-Expressions { 2 8 1 5 7 }

DEFINITIONS ::= BEGIN

EXPORTS Content-Portion-Identifier, Object-or-Class-Identifier,
Style-Identifier, Layout-Category-Name,
Resource-Name, Binding-Name,
Construction-Expression, Object-Id-Expression,
Numeric-Expression, String-Expression;

IMPORTS Layout-Object-Type
FROM Layout-Descriptors - - see 5.8
Logical-Object-Type
FROM Logical-Descriptors; - - see 5.9

Content-Portion-Identifier ::= [APPLICATION 0] IMPLICIT PrintableString

- - only digits and space are used in the present version
- - of the standard; other characters are reserved for extensions

Object-or-Class-Identifier ::= [APPLICATION 1] IMPLICIT PrintableString

- - only digits and space are used in the present version
- - of the standard; other characters are reserved for extensions;
- - a "null" value is represented by an empty string

Style-Identifier ::= [APPLICATION 5] IMPLICIT PrintableString

- - only digits and space are used in the present version
- - of the standard; other characters are reserved for extensions;
- - a "null" value is represented by an empty string

Layout-Category-Name ::= PrintableString

- - a "null" value is represented by an empty string

Resource-Name ::= PrintableString

Binding-Name ::= PrintableString

Construction-Expression ::= CHOICE {
construction-type Construction-Type,
single-term-construction [3] Construction-Term }

Construction-Type ::= CHOICE {
sequence-construction [0] IMPLICIT Term-Sequence,
aggregate-construction [1] IMPLICIT Term-Sequence,
choice-construction [2] IMPLICIT Term-Sequence }

Term-Sequence ::= SEQUENCE OF Construction-Term

Construction-Term ::= CHOICE {
required-construction-factor [0] Construction-Factor,
optional-construction-factor [1] Construction-Factor,
repetitive-construction-factor [2] Construction-Factor,
optional-repetitive-factor [3] Construction-Factor }

Construction-Factor ::= CHOICE {
object-class-identifier Object-or-Class-Identifier,
construction-type Construction-Type }

Object-Id-Expression current-object-function preceding-object-function superior-object-function current-instance-function	::= CHOICE { [0] IMPLICIT NULL, [1] Object-Id-Expression, [3] Object-Id-Expression, [4] Current-Instance-Function}
Numeric-Expression numeric-literal increment-application decrement-application ordinal-application identifier expression binding-reference	::= CHOICE { [0] IMPLICIT INTEGER, [1] Numeric-Expression, [2] Numeric-Expression, [3] CHOICE { Object-or-Class-Identifier, Object-Id-Expression}, [4] IMPLICIT Binding-Reference}
Binding-Reference object-reference identifier expression binding-identifier	::= SET { CHOICE { Object-or-Class-Identifier, Binding-Selection-Function}, Binding-Name}
Binding-Selection-Function current-object-function preceding-function superior-function current-instance-function	::= CHOICE { [0] IMPLICIT NULL, [1] Object-Id-Expression, [3] Object-Id-Expression, [4] Current-Instance-Function}
Current-Instance-Function first-parameter identifier layout-object-type logical-object-type second-parameter identifier expression	::= SEQUENCE { CHOICE { [0] IMPLICIT Object-or-Class-Identifier, [1] IMPLICIT Layout-Object-Type, [2] IMPLICIT Logical-Object-Type}, CHOICE { Object-or-Class-Identifier, Object-Id-Expression}}
String-Expression	::= SEQUENCE OF Atomic-String-Expression
Atomic-String-Expression string-literal binding-reference make-string-application upper-alpha-application lower-alpha-application upper-roman-application lower-roman-application	::= CHOICE { [0] IMPLICIT OCTET STRING, [2] IMPLICIT Binding-Reference, [3] Numeric-Expression, [4] Numeric-Expression, [5] Numeric-Expression, [6] Numeric-Expression, [7] Numeric-Expression}
	END

5.8 Layout descriptors

Layout-Descriptors { 2 8 1 5 8 }

DEFINITIONS ::= BEGIN

EXPORTS Layout-Object-Descriptor, Layout-Class-Descriptor,
Layout-Object-Type, Transparency, Comment-String,
Binding-Pair, One-Of-Four-Angles, Measure-Pair, Dimension-Pair,
Medium-Type, Colour, Border;

IMPORTS Object-or-Class-Identifier, Style-Identifier,
Layout-Category-Name, Resource-Name, Binding-Name,
Construction-Expression, Object-Id-Expression,
Numeric-Expression, String-Expression
FROM Identifiers-and-Expressions - - see 5.7
Presentation-Attributes
FROM Style-Descriptors - - see 5.10
Default-Value-Lists-Layout
FROM Default-Value-Lists; - - see 5.11

Position-Spec ::= SET {
offset [0] IMPLICIT SET {
leading [0] IMPLICIT INTEGER OPTIONAL,
trailing [1] IMPLICIT INTEGER OPTIONAL,
left-hand [2] IMPLICIT INTEGER OPTIONAL,
right-hand [3] IMPLICIT INTEGER OPTIONAL} OPTIONAL,
separation [1] IMPLICIT SET {
leading [0] IMPLICIT INTEGER OPTIONAL,
trailing [1] IMPLICIT INTEGER OPTIONAL,
centre [2] IMPLICIT INTEGER OPTIONAL} OPTIONAL,
alignment [2] IMPLICIT INTEGER {
right-hand (0), centred (1),
left-hand (2)} OPTIONAL,
fill-order [3] IMPLICIT INTEGER {
normal (0), reverse (1)} OPTIONAL}

Dimension-Pair ::= SEQUENCE {
horizontal [0] IMPLICIT INTEGER,
vertical CHOICE {
fixed [0] IMPLICIT INTEGER,
variable [1] IMPLICIT INTEGER}

Dimension-Spec ::= SEQUENCE {
horizontal Dimension,
vertical Dimension}

Dimension ::= CHOICE {
fixed [0] IMPLICIT INTEGER,
rule-a [1] IMPLICIT SET {
minimum [0] IMPLICIT INTEGER OPTIONAL,
maximum [1] IMPLICIT INTEGER OPTIONAL},
rule-b [2] IMPLICIT SET {
minimum [0] IMPLICIT INTEGER OPTIONAL,
maximum [1] IMPLICIT INTEGER OPTIONAL},
maximum-size [3] IMPLICIT NULL}

Transparency ::= INTEGER {transparent (0), opaque (1)}

Comment-String ::= OCTET STRING

- - string of characters from the sets designated by
- - the document profile attribute "comments character sets",
- - plus code extension control functions,
- - carriage return and line feed

Binding-Pair binding-identifier binding-value	::= SET { [0] IMPLICIT Binding-Name, CHOICE { [1] Object-Id-Expression, [2] Numeric-Expression, [3] String-Expression, [4] IMPLICIT Object-or-Class-Identifier, [5] IMPLICIT INTEGER, [6] IMPLICIT OCTET STRING}}
One-Of-Four-Angles	::= INTEGER {d0 (0), d90 (1), d180 (2), d270 (3)}
Measure-Pair horizontal vertical	::= SEQUENCE { [0] IMPLICIT INTEGER, [0] IMPLICIT INTEGER}
Medium-Type nominal-page-size side-of-sheet	::= SEQUENCE { Measure-Pair OPTIONAL, INTEGER {unspecified (0), recto (1), verso (2)} OPTIONAL}
Colour	::= INTEGER {colourless (0), white (1)}
Border left-hand-edge right-hand-edge trailing-edge leading-edge	::= SET { [0] IMPLICIT Border-Edge OPTIONAL, [1] IMPLICIT Border-Edge OPTIONAL, [2] IMPLICIT Border-Edge OPTIONAL, [3] IMPLICIT Border-Edge OPTIONAL}
Border-Edge line-width line-type freespace-width	::= SET { [0] IMPLICIT INTEGER OPTIONAL, [1] IMPLICIT INTEGER { invisible (0), solid (1), dashed (2), dot (3), dash-dot (4), dash-dot-dot (5)} OPTIONAL, [2] IMPLICIT INTEGER OPTIONAL}
- - a "null" border edge is represented by an empty set	
Layout-Object-Descriptor object-type descriptor-body	::= SEQUENCE { Layout-Object-Type OPTIONAL, Layout-Object-Descriptor-Body OPTIONAL}
Layout-Object-Type	::= INTEGER {document-layout-root (0), page-set (1), page (2), frame (3), block (4)}
Layout-Object-Descriptor-Body object-identifier subordinates content-portions object-class position dimensions transparency presentation-attributes default-value-lists user-readable-comments bindings layout-path imaging-order permitted-categories	::= SET { Object-or-Class-Identifier OPTIONAL, [0] IMPLICIT SEQUENCE OF NumericString OPTIONAL, [1] IMPLICIT SEQUENCE OF NumericString OPTIONAL, [2] IMPLICIT Object-or-Class-Identifier OPTIONAL, [3] IMPLICIT Measure-Pair OPTIONAL, [4] IMPLICIT Dimension-Pair OPTIONAL, [5] IMPLICIT Transparency OPTIONAL, [6] IMPLICIT Presentation-Attributes OPTIONAL, [7] IMPLICIT Default-Value-Lists-Layout OPTIONAL, [8] IMPLICIT Comment-String OPTIONAL, [9] IMPLICIT SET OF Binding-Pair OPTIONAL, [11] IMPLICIT One-Of-Four-Angles OPTIONAL, [12] IMPLICIT SEQUENCE OF NumericString OPTIONAL, [13] IMPLICIT SET OF Layout-Category-Name OPTIONAL,
- - a "null" value is represented by an empty set	
user-visible-name page-position	[14] IMPLICIT Comment-String OPTIONAL, [15] IMPLICIT Measure-Pair OPTIONAL,

medium-type	[16] IMPLICIT Medium-Type OPTIONAL,
presentation-style	[17] IMPLICIT Style-Identifier OPTIONAL,
balance	[21] IMPLICIT SEQUENCE OF Object-or-Class-Identifier OPTIONAL,

- - a "null" value is represented by an empty set

colour	[22] IMPLICIT Colour OPTIONAL,
border	[23] IMPLICIT Border OPTIONAL,
application-comments	[25] IMPLICIT OCTET STRING OPTIONAL}

Layout-Class-Descriptor	::= SEQUENCE {
object-type	Layout-Object-Type,
descriptor-body	Layout-Class-Descriptor-Body}

Layout-Class-Descriptor-Body	::= SET {
object-class-identifier	Object-or-Class-Identifier,
generator-for-subordinates	[0] Construction-Expression OPTIONAL,
content-portions	[1] IMPLICIT SEQUENCE OF NumericString OPTIONAL,
position	CHOICE {
fixed-position	[3] IMPLICIT Measure-Pair,
variable-position	[26] IMPLICIT Position-Spec} OPTIONAL,
dimensions	[4] IMPLICIT Dimension-Spec OPTIONAL,
transparency	[5] IMPLICIT Transparency OPTIONAL,
presentation-attributes	[6] IMPLICIT Presentation-Attributes OPTIONAL,
default-value-lists	[7] IMPLICIT Default-Value-Lists-Layout OPTIONAL,
user-readable-comments	[8] IMPLICIT Comment-String OPTIONAL,
bindings	[9] IMPLICIT SET OF Binding-Pair OPTIONAL,
content-generator	[10] IMPLICIT String-Expression OPTIONAL,
layout-path	[11] IMPLICIT One-Of-Four-Angles OPTIONAL,
permitted-categories	[13] IMPLICIT SET OF Layout-Category-Name OPTIONAL,

- - a "null" value is represented by an empty set

user-visible-name	[14] IMPLICIT Comment-String OPTIONAL,
page-position	[15] IMPLICIT Measure-Pair OPTIONAL,
medium-type	[16] IMPLICIT Medium-Type OPTIONAL,
presentation-style	[17] IMPLICIT Style-Identifier OPTIONAL,
logical-source	[18] IMPLICIT Object-or-Class-Identifier OPTIONAL,
balance	[21] IMPLICIT SEQUENCE OF Object-or-Class-Identifier OPTIONAL,

- - a "null" value is represented by an empty sequence

colour	[22] IMPLICIT Colour OPTIONAL,
border	[23] IMPLICIT Border OPTIONAL,
resource	[24] IMPLICIT Resource-Name OPTIONAL,
application-comments	[25] IMPLICIT OCTET STRING OPTIONAL}

END

5.9 Logical descriptors

Logical-Descriptors { 2 8 1 5 9 }

DEFINITIONS ::= BEGIN

EXPORTS Logical-Object-Descriptor, Logical-Class-Descriptor,
Logical-Object-Type, Protection;

IMPORTS Object-or-Class-Identifier, Style-Identifier,
Resource-Name, Construction-Expression, String-Expression
FROM Identifiers-and-Expressions - - see 5.7
Comment-String, Binding-Pair
FROM Layout-Descriptors - - see 5.8
Presentation-Attributes FROM Style-Descriptors - - see 5.10
Default-Value-Lists-Logical
FROM Default-Value-Lists; - - see 5.11

Logical-Object-Descriptor ::= SEQUENCE {
 object-type
 descriptor-body
 Logical-Object-Type OPTIONAL,
 Logical-Object-Descriptor-Body OPTIONAL}

Logical-Object-Type ::= INTEGER {document-logical-root (0),
 composite-logical-object (1),
 basic-logical-object (2)}

Logical-Object-Descriptor-Body ::= SET {
 object-identifier
 subordinates
 content-portions
 object-class
 presentation-attributes
 Object-or-Class-Identifier OPTIONAL,
 [0] IMPLICIT SEQUENCE OF NumericString OPTIONAL,
 [1] IMPLICIT SEQUENCE OF NumericString OPTIONAL,
 [2] IMPLICIT Object-or-Class-Identifier OPTIONAL,
 [6] IMPLICIT Presentation-Attributes OPTIONAL,

- - only for use for the attribute content-architecture-class
- - the content architecture specific attributes can only be referenced by
- - use of presentation style

default-value-lists [7] IMPLICIT Default-Value-Lists-Logical OPTIONAL,
user-readable-comments [8] IMPLICIT Comment-String OPTIONAL,
bindings [9] IMPLICIT SET OF Binding-Pair OPTIONAL,
content-generator [10] IMPLICIT String-Expression OPTIONAL,
user-visible-name [14] IMPLICIT Comment-String OPTIONAL,
presentation-style [17] IMPLICIT Style-Identifier OPTIONAL,
layout-style [19] IMPLICIT Style-Identifier OPTIONAL,
protection [20] IMPLICIT Protection OPTIONAL,
application-comments [25] IMPLICIT OCTET STRING OPTIONAL}

Logical-Class-Descriptor ::= SEQUENCE {
 object-type
 descriptor-body
 Logical-Object-Type,
 Logical-Class-Descriptor-Body}

Logical-Class-Descriptor-Body ::= SET {
 object-class-identifier
 generator-for-subordinates
 content-portions
 presentation-attributes
 Object-or-Class-Identifier,
 [0] Construction-Expression OPTIONAL,
 [1] IMPLICIT SEQUENCE OF NumericString OPTIONAL,
 [6] IMPLICIT Presentation-Attributes OPTIONAL,

- - only for use for the attribute content-architecture-class
- - the content architecture specific attributes can only be referenced by
- - use of presentation style

default-value-lists [7] IMPLICIT Default-Value-Lists-Logical OPTIONAL,
user-readable-comments [8] IMPLICIT Comment-String OPTIONAL,
bindings [9] IMPLICIT SET OF Binding-Pair OPTIONAL,
content-generator [10] IMPLICIT String-Expression OPTIONAL,

user-visible-name	[14] IMPLICIT Comment-String OPTIONAL,
presentation-style	[17] IMPLICIT Style-Identifier OPTIONAL,
layout-style	[19] IMPLICIT Style-Identifier OPTIONAL,
protection	[20] IMPLICIT Protection OPTIONAL,
resource	[24] IMPLICIT Resource-Name OPTIONAL,
application-comments	[25] IMPLICIT OCTET STRING OPTIONAL}

Protection ::= INTEGER {unprotected (0), protected (1)}

END

5.10 Style descriptors

Style-Descriptors { 2 8 1 5 10 }

DEFINITIONS ::= BEGIN

EXPORTS Presentation-Style-Descriptor, Presentation-Attributes,
Content-Type, Content-Architecture-Class,
Layout-Style-Descriptor, Fill-Order, Block-Alignment;

IMPORTS Object-or-Class-Identifier, Style-Identifier,
Layout-Category-Name, Object-Id-Expression,
FROM Identifiers-and-Expressions - - see 5.7
Comment-String, Transparency, Colour, Border,
Layout-Object-Type
FROM Layout-Descriptors - - see 5.8
Character-Attributes
FROM Character-Presentation-Attributes { 2 8 1 6 2 } - - see ISO 8613-6
Raster-Graphics-Attributes
FROM Raster-Gr-Presentation-Attributes { 2 8 1 7 2 } - - see ISO 8613-7
Geometric-Graphics-Attributes
FROM Geo-Gr-Presentation-Attributes { 2 8 1 8 2 }; - - see ISO 8613-8

Presentation-Style-Descriptor ::= SET {
style-identifier Style-Identifier,
user-readable-comments [0] IMPLICIT Comment-String OPTIONAL,
user-visible-name [1] IMPLICIT Comment-String OPTIONAL,
transparency [2] IMPLICIT Transparency OPTIONAL,
presentation-attributes [3] IMPLICIT Presentation-Attributes OPTIONAL,
colour [4] IMPLICIT Colour OPTIONAL,
border [5] IMPLICIT Border OPTIONAL}

Presentation-Attributes ::= SET {
content-type Content-Type OPTIONAL,
content-architecture-class Content-Architecture-Class OPTIONAL,
character-attributes [0] IMPLICIT Character-Attributes OPTIONAL,
raster-graphics-attributes [1] IMPLICIT Raster-Graphics-Attributes OPTIONAL,
geometric-graphics-attributes [2] IMPLICIT Geometric-Graphics-Attributes OPTIONAL,

- - the following tags are reserved for additional types
- - of presentation attributes:
 - - [3] videotex, for use in conjunction with CCITT Recommendations
 - - [4] audio
 - - [5] dynamic-graphics

ext-cont-arch-pres-attributes [6] IMPLICIT SEQUENCE OF EXTERNAL OPTIONAL}

Content-Type ::= [APPLICATION 2] IMPLICIT INTEGER {
formatted-raster-graphics (1)}

Content-Architecture-Class ::= OBJECT IDENTIFIER

Layout-Style-Descriptor ::= SET {
style-identifier Style-Identifier,
user-readable-comments [0] IMPLICIT Comment-String OPTIONAL,
user-visible-name [1] IMPLICIT Comment-String OPTIONAL,
layout-directives [4] IMPLICIT Layout-Directives OPTIONAL}

Layout-Directives ::= SET {
indivisibility CHOICE {
to-layout-object-class [0] IMPLICIT Object-or-Class-Identifier,
to-layout-category [1] IMPLICIT Layout-Category-Name,
to-layout-object-type [2] IMPLICIT Layout-Object-Type,
null [15] IMPLICIT NULL} OPTIONAL,

separation	[3] IMPLICIT Separation OPTIONAL,
offset	[4] IMPLICIT Offset OPTIONAL,
fill-order	[5] IMPLICIT Fill-Order OPTIONAL,
concatenation	[6] IMPLICIT Concatenation OPTIONAL,
new-layout-object	CHOICE {
to-layout-object-class	[7] IMPLICIT Object-or-Class-Identifier,
to-layout-category	[8] IMPLICIT Layout-Category-Name,
to-layout-object-type	[9] IMPLICIT Layout-Object-Type,
null	[16] IMPLICIT NULL} OPTIONAL,
same-layout-object	[10] IMPLICIT Same-Layout-Object OPTIONAL,
layout-object-class	[11] IMPLICIT Object-or-Class-Identifier OPTIONAL,
layout-category	[12] IMPLICIT Layout-Category-Name OPTIONAL,
synchronization	CHOICE {
	[13] IMPLICIT Object-or-Class-Identifier,
	[17] Object-Id-Expression,
	[18] IMPLICIT NULL} OPTIONAL,
block-alignment	[14] IMPLICIT Block-Alignment OPTIONAL}
Separation	::= SET {
leading	[0] IMPLICIT INTEGER OPTIONAL,
trailing	[1] IMPLICIT INTEGER OPTIONAL,
centre	[2] IMPLICIT INTEGER OPTIONAL}
Offset	::= SET {
right-hand	[0] IMPLICIT INTEGER OPTIONAL,
left-hand	[1] IMPLICIT INTEGER OPTIONAL,
trailing	[2] IMPLICIT INTEGER OPTIONAL,
leading	[3] IMPLICIT INTEGER OPTIONAL}
Fill-Order	::= INTEGER {normal (0), reverse (1)}
Concatenation	::= INTEGER {non-concatenated (0), concatenated (1)}
Same-Layout-Object	::= SET {
to-logical-object	CHOICE {
	[0] IMPLICIT Object-or-Class-Identifier,
	[4] Object-Id-Expression,
	[5] IMPLICIT NULL},
	CHOICE {
to-layout-object-class	[1] IMPLICIT Object-or-Class-Identifier,
to-layout-category	[2] IMPLICIT Layout-Category-Name,
to-layout-object-type	[3] IMPLICIT Layout-Object-Type} OPTIONAL}
Block-Alignment	::= INTEGER {
	right-hand (0), left-hand (1),
	centred (2), null (3)}
	END

5.11 Default value lists

Default-Value-Lists { 2 8 1 5 11 }

DEFINITIONS ::= BEGIN

EXPORTS Default-Value-Lists-Logical, Default-Value-Lists-Layout;

IMPORTS Style-Identifier, Layout-Category-Name
 FROM Identifiers-and-Expressions - - see 5.7
 Measure-Pair, One-Of-Four-Angles, Medium-Type,
 Dimension-Pair, Transparency, Colour, Border
 FROM Layout-Descriptors - - see 5.8
 Protection FROM Logical-Descriptors - - see 5.9
 Presentation-Attributes
 FROM Style-Descriptors; - - see 5.10

Default-Value-Lists-Layout ::= SET {
 page-attributes [2] IMPLICIT Page-Attributes OPTIONAL,
 frame-attributes [3] IMPLICIT Frame-Attributes OPTIONAL,
 block-attributes [4] IMPLICIT Block-Attributes OPTIONAL}

Default-Value-Lists-Logical ::= SET {
 composite-logical-attributes [5] IMPLICIT Composite-Logical-Attributes OPTIONAL,
 basic-logical-attributes [6] IMPLICIT Basic-Logical-Attributes OPTIONAL}

Page-Attributes ::= SET {
 dimensions < Attribute OPTIONAL,
 transparency < Attribute OPTIONAL,
 presentation-attributes < Attribute OPTIONAL,
 page-position < Attribute OPTIONAL,
 medium-type < Attribute OPTIONAL,
 presentation-style < Attribute OPTIONAL,
 colour < Attribute OPTIONAL}

Frame-Attributes ::= SET {
 position < Attribute OPTIONAL,
 dimensions < Attribute OPTIONAL,
 transparency < Attribute OPTIONAL,
 layout-path < Attribute OPTIONAL,
 permitted-categories < Attribute OPTIONAL,
 colour < Attribute OPTIONAL,
 border < Attribute OPTIONAL}

Block-Attributes ::= SET {
 position < Attribute OPTIONAL,
 dimensions < Attribute OPTIONAL,
 transparency < Attribute OPTIONAL,
 presentation-attributes < Attribute OPTIONAL,
 presentation-style < Attribute OPTIONAL,
 colour < Attribute OPTIONAL,
 border < Attribute OPTIONAL}

Composite-Logical-Attributes ::= SET {
 protection < Attribute OPTIONAL,
 layout-style < Attribute OPTIONAL}

Basic-Logical-Attributes ::= SET {
 presentation-attributes < Attribute OPTIONAL,

- - only for use for the attribute content-architecture-class
- - the content architecture specific attributes can only be referenced by
- - use of presentation style

protection < Attribute OPTIONAL,

presentation-style	< Attribute OPTIONAL,
layout-style	< Attribute OPTIONAL}
Attribute	::= CHOICE {
position	[0] IMPLICIT Measure-Pair,
dimensions	[1] IMPLICIT Dimension-Pair,
transparency	[2] IMPLICIT Transparency,
presentation-attributes	[3] IMPLICIT Presentation-Attributes,
layout-path	[4] IMPLICIT One-Of-Four-Angles,
page-position	[5] IMPLICIT Measure-Pair,
medium-type	[6] IMPLICIT Medium-Type,
permitted-categories	[7] IMPLICIT SET OF Layout-Category-Name,
protection	[8] IMPLICIT Protection,
presentation-style	[9] IMPLICIT Style-Identifier,
layout-style	[10] IMPLICIT Style-Identifier,
colour	[11] IMPLICIT Colour,
border	[12] IMPLICIT Border}
	END

5.12 Text units

Text-Units { 2 8 1 5 12 }

DEFINITIONS ::= BEGIN

EXPORTS Text-Unit, Type-Of-Coding;

IMPORTS Content-Portion-Identifier
 FROM Identifiers-and-Expressions - - see 5.7
 Character-Coding-Attributes
 FROM Character-Coding-Attributes { 2 8 1 6 3 } - - see ISO 8613-6
 Raster-Gr-Coding-Attributes
 FROM Raster-Gr-Coding-Attributes { 2 8 1 7 3 } - - see ISO 8613-7
 Geo-Gr-Coding-Attributes
 FROM Geo-Gr-Coding-Attributes { 2 8 1 8 3 }; - - see ISO 8613-8

Text-Unit ::= SEQUENCE {
 content-portion-attributes Content-Portion-Attributes OPTIONAL,
 content-information Content-Information OPTIONAL}

Content-Portion-Attributes ::= SET {
 content-identifier-layout Content-Portion-Identifier OPTIONAL,
 content-identifier-logical [4] IMPLICIT Content-Portion-Identifier OPTIONAL,
 type-of-coding Type-Of-Coding OPTIONAL,
 coding-attributes CHOICE {
 character-coding-attributes [1] IMPLICIT Character-Coding-Attributes,
 raster-gr-coding-attributes [2] IMPLICIT Raster-Gr-Coding-Attributes,
 geo-gr-coding-attributes [7] IMPLICIT Geo-Gr-Coding-Attributes,

- - the following data element is used in CCITT Recommendations T.415

- - videotex-coding-attributes
 - - [8] IMPLICIT Videotex-Coding-Attributes,

- - the following tags are reserved for additional types

- - of coding attributes:
 - - [9] audio
 - - [10] dynamic-graphics

ext-cont-arch-coding-attributes [11] IMPLICIT EXTERNAL} OPTIONAL,
 alternative-representation [3] IMPLICIT Alternative-Representation OPTIONAL}

Content-Information ::= OCTET STRING

Type-Of-Coding ::= CHOICE {
 [0] IMPLICIT INTEGER { t6 (1)},
 [6] IMPLICIT OBJECT IDENTIFIER}

Alternative-Representation ::= OCTET STRING

- - string of characters from the sets designated by the document
 - - profile attribute "alternative representation character sets",
 - - plus carriage return and line feed

END

Annex A

(informative)

Coded representation

This annex is a summary of the basic encoding rules for the abstract syntax notation ASN.1 defined in ISO 8825.

The coded representation of each data structure or data item that constitutes, or constitutes part of, a descriptor or a text unit consists of a type field, a length field and a value field.

If the data item concerned is an elementary data item, then the type field specifies the elementary data type, the length field specifies the length of the value field, and the value field represents the value of the data item.

If the data structure or data item concerned is not elementary, then the type field identifies the attribute or group of attributes corresponding to the data structure or data item, the length field specifies the length of the value field, and the value field consists of one or more triplets, each of which is composed of a type field, a length field and a value field, representing the subordinate data structures and data items.

The *type field* (which is called "identifier octets" in ISO 8825) consists of one or more bytes. The bits of the first byte are used as follows:

- bits 8 and 7: tag class (00: universal,
01: application,
10: context-specific,
11: private);
- bit 6: contents encoding form (0: simple,
1: structured);
- bits 5 to 1: 00000 to 11110: tag number;
11111 indicates a multi-octet type field.

The following tag numbers for universal tags have been assigned in ISO 8824 and ISO 8825:

Built-in data types

- 0: End-of-contents
- 1: Boolean
- 2: Integer
- 3: Bit String
- 4: Octet String
- 5: Null
- 6: Object Identifier
- 7: Object Descriptor
- 8: External
- 9: Real
- 10: Enumerated
- 11: Encrypted
- 16: Sequence
- 17: Set

Defined data types

- 18: Numeric String
- 19: Printable String
- 20: Teletex String
- 21: Videotex String
- 22: IA5 String
- 23: UTC Time
- 24: Generalized Time
- 25: Graphic Character String
- 26: General String
- 27: Visible String

Data items of type End-of-contents, Boolean, Integer or Null are simple (elementary data items). Sequences and Sets are structured (data structures with subordinate data items). Data items of type Bit String, Octet String or any of the defined data types can be either simple or structured.

The *length* field consists of one or more bytes. It takes one of three forms: short, long and indefinite. The bits of the first byte are used as follows:

bit 8: length field form (0: short, 1: long or indefinite)

bits 7 to 1: if bit 8 = 0: number of bytes of the value field;
if bit 8 = 1: number of bytes of the length field following the first byte;
0000000 indicates the indefinite form of the length field.

A data structure or data item with an indefinite length field must be structured and must be terminated by a delimiter consisting of an End-of-contents (EOC) item. An EOC item consists of two bytes: a type field of one byte and a length field of one byte. Both are equal to zero. An EOC item has no value field.

Annex B

(informative)

Application class tag assignments

The application class tag assignments made in various clauses of this part of ISO 8613 are summarized in the table below:

Tag	Data type	Clause
APPLICATION 0	Content-Portion-Identifier	5.7
APPLICATION 1	Object-or-Class-Identifier	5.7
APPLICATION 2	Content-Type	5.10
APPLICATION 3	Character-Data	5.6
APPLICATION 4	Date-and-Time	5.6
APPLICATION 5	Style-Identifier	5.7
APPLICATION 6	Personal-Name	5.6