INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

T.502

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(11/94)

TERMINALS FOR TELEMATIC SERVICES

DOCUMENT APPLICATION PROFILE PM-11 FOR THE INTERCHANGE OF SIMPLE STRUCTURE, CHARACTER CONTENT DOCUMENTS IN PROCESSABLE AND FORMATTED FORMS

ITU-T Recommendation T.502

(Previously “CCITT Recommendation”)

FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation T.502 was revised by ITU-T Study Group 8 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 11th of November 1994.

NOTE

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

CONTENTS

Page

1 Scope 1

2 Normative references 1

2.1 Paired Recommendations | International Standards equivalent in technical content 1

2.2 Additional references 2

3 Definitions 3

4 Relationship with other Profiles 3

5 Conformance 3

5.1 Data stream conformance 4

5.2 Implementation conformance 4

6 Characteristics supported by this document application profile 4

6.1 Overview 4

6.1.1 General 4

6.1.2 Formatted form documents 4

6.1.3 Processable form documents 5

6.1.4 Formatted processable form documents 5

6.2 Logical characteristics 5

6.2.1 Introduction 5

6.2.2 Overview of the logical structure 6

6.2.3 Body part of the logical structure 6

6.2.4 Common content part of the logical structure 6

6.3 Layout characteristics 7

6.3.1 Overview of the layout characteristics 7

6.3.2 DocumentLayoutRoot 9

6.3.3 PageSet 9

6.3.4 Page characteristics 9

6.3.5 Body area characteristics 11

6.3.6 Header and footer area characteristics 11

6.3.7 SpecificBlock 11

6.4 Document layout characteristics 11

6.4.1 Flow controls 12

6.4.2 Layout of the document content 13

6.4.3 Layout controls applicable in the absence of a generic layout structure 13

6.5 Content layout and imaging characteristics 14

6.5.1 Introduction 14

6.5.2 Character content architecture class 15

6.5.3 Character repertoire 15

6.5.4 Code extension techniques 16

6.5.5 Line spacing 16

6.5.6 Character spacing 16

6.5.7 Character path and line progression 18

6.5.8 Character orientation 18

Page

6.5.9 Emphasis ... 18
6.5.10 Tabulation ... 19
6.5.11 Indentation ... 19
6.5.12 Alignment ... 19
6.5.13 First line format ... 19
6.5.14 Widow and orphan sizes ... 20
6.5.15 Reverse character string ... 20
6.5.16 Superscripts and subscripts ... 20
6.5.17 Line breaks ... 20
6.5.18 Substitution of characters ... 20
6.5.19 Initial point ... 21
6.5.20 Use of control functions ... 21
6.5.21 Formatting the content ... 21

6.6 Miscellaneous features ... 21
6.6.1 Application comments ... 21
6.6.2 Alternative representation ... 22
6.6.3 Page numbering ... 22
6.6.4 User readable comments ... 23
6.6.5 User visible name ... 23

6.7 Document management features ... 24
6.7.1 Document constituent information ... 24
6.7.2 Document identification information ... 24
6.7.3 Document default information ... 25
6.7.4 Non-basic characteristics information ... 25
6.7.5 Document management attributes ... 26

7 Specification of constituent constraints ... 27

7.1 Introduction ... 27
7.1.1 Diagrams of relationships of logical constituents ... 27
7.1.2 Diagrams of relationships of layout constituents ... 28
7.1.3 Notation ... 29

7.2 Document profile constituent constraints ... 30
7.2.1 Macro definitions ... 30
7.2.2 Constituent constraints ... 34

7.3 Logical constituent constraints ... 36
7.3.1 Macro definitions ... 36
7.3.2 Factor constraints ... 37
7.3.3 Constituent constraints ... 37

7.4 Layout constituent constraints ... 39
7.4.1 Macro definitions ... 39
7.4.2 Factor constraints ... 40
7.4.3 Constituent constraints ... 41

7.5 Layout style constituent constraints ... 44
7.5.1 Macro definitions ... 44
7.5.2 Factor constraints ... 44
7.5.3 Constituent constraints ... 44

7.6 Presentation style constituent constraints ... 45
7.6.1 Macro definitions ... 45
7.6.2 Factor constituent constraints ... 45
7.6.3 Constituent constraints ... 45

Page

7.7 Content portion constituent constraints 46
7.7.1 Macro definitions 46
7.7.2 Factor constraints 46
7.7.3 Constituent constraints 46

8 Interchange format 47
8.1 Interchange format 47
8.2 Document application profile object identifier 48
8.3 Encoding of application comments 48
8.4 Data lengths 48

Annex A – Amendments and technical corrigenda 48
A.1 Amendments 48
A.1.1 Amendments to the base standard 48
A.1.2 Proposed changes to standards due to defects 49
A.2 Technical corrigenda 49
A.2.1 Technical corrigenda to this Recommendation 49
A.3 Versions of the ODA International Standard 49

Annex B – Recommended practices 49
B.1 Transfer methods for ODA 49
B.1.1 Conveyance of ODA over CCITT X.400-1984 49
B.1.2 Conveyance of ODA over FTAM 50
B.1.3 Conveyance of ODA over DTAM 50
B.1.4 Conveyance of ODA over flexible disks 50

Annex C – Bibliography 50

SUMMARY

This Recommendation specifies an ODA Document Application Profile (DAP) named PM-11.

This Profile specifies interchange format for the transfer of structured documents between equipment designed for word or document processing. Such documents contain character content.

The documents that can be interchanged using this Profile range from memos and letters to simple structured documents. This Profile provides a comprehensive level of features for the transfer of documents between these systems.

A document structured in accordance with this Profile is represented for interchange by the Open Document Interchange Format (ODIF), as defined in Recommendation T.415.

INTRODUCTION

This Recommendation specifies an ODA Document Application Profile (DAP) named PM-11. The purpose of this Profile is to facilitate the interworking of applications interchanging documents based on ODA, T.410-Series CCITT Recommendations | ISO/IEC 8613. This Profile is suitable for interchanging documents in formatted form, processable form or formatted processable form and has been defined in accordance with CCITT Recommendation T.411 | ISO/IEC 8613-1. The format of this Profile is in accordance with the standardized proforma and notation defined in Annex F of CCITT Recommendation T.411 | ISO/IEC 8613-1.

Recommendation T.502

DOCUMENT APPLICATION PROFILE PM-11 FOR THE INTERCHANGE OF SIMPLE STRUCTURE, CHARACTER CONTENT DOCUMENTS IN PROCESSABLE AND FORMATTED FORMS

(revised in 1994)

1 Scope

This Profile specifies interchange format for the transfer of structured documents between equipment designed for word or document processing. Such documents contain character content.

The documents that can be interchanged using this Profile range from memos and letters to simple structured documents. This Profile provides a comprehensive level of features for the transfer of documents between these systems.

This Profile allows documents to be interchanged in the following forms:

- formatted form;
- processable form;
- formatted processable form.

The architecture levels defined for these three forms have matching functionalities so that the interchange formats of a document are convertible from a processable form to any other form.

This Profile is independent of the processes carried out in an end system to create, edit or reproduce documents. It is also independent of the means to transfer documents which, for example, may be by means of communication links or storage media.

A document structured in accordance with this Profile is represented for interchange by the Open Document Interchange Format (ODIF), as defined in Recommendation T.415 | ISO/IEC 8613-5.

2 Normative references

The following Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision: all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of currently valid ITU-T Recommendations is regularly published.

2.1 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation T.411 (1988), *Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles*.

  ISO 8613-1:1989, *Information processing – Text and office systems – Office Document – Architecture (ODA) and Interchange Format – Part 1: Introduction and General Principles*.

- CCITT Recommendation T.411 Annex F (1991), *Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles – Annex F: Document Application Profile Proforma and Notation*.

  ISO 8613-1 Add. 1: *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format – Part 1: Introduction and General Principles – Addendum 1: Document Application Profile Proforma and Notation*.

– CCITT Recommendation T.412 (1988), *Open Document Architecture (ODA) and Interchange Format: Document Structures*.

  ISO 8613-2:1989, *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format – Part 2: Document Structures*.

– CCITT Recommendation T.414 (1988), *Open Document Architecture (ODA) and Interchange Format: Document Profile*.

  ISO 8613-4:1989, *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format – Part 4: Document Profile*.

– CCITT Recommendation T.415 (1988), *Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format (ODIF)*.

  ISO 8613-5:1989, *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format – Part 5: Office Document Interchange Format (ODIF)*.

– CCITT Recommendation T.416 (1988), *Open Document Architecture (ODA) and Interchange Format: Character Content Architecture*.

  ISO 8613-6:1989, *Information processing – Text and office systems – Office Document Architecture (ODA) and Interchange Format – Part 6: Character content architectures*.

– CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1)*.

  ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*.

– CCITT Recommendation X.209 (1988), *Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1)*.

  ISO/IEC 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.

– ITU-T Recommendation T.505 (1994), *Document Application Profile PM-26 for the Interchange of Enhanced Structure, Mixed Content Documents in Processable and Formatted Forms*.

  ISO/IEC ISP 11181-1:1993, *Information technology – International Standardized Profile FOD26 – Open Document Format: Enhanced document structure – Character, raster graphics and geometric graphics content architectures – Part 1: Document Application Profile (DAP)*.

– ITU-T Recommendation T.506 (1993), *Document Application Profile PM-36 for the Interchange of Extended Structure, Mixed Content Documents in Processable and Formatted Forms*,

  ISO/IEC ISP 11182-1:1993, *Information technology – International Standardized Profile FOD36 – Open Document Format: Extended document structure – Character, raster graphics and geometric graphics content architectures – Part 1: Document Application Profile (DAP)*.

– CCITT Recommendation T.50 (1992), *International Reference Alphabet (IRA). Information technology – 7-bit coded character set for information interchange*.

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

– CCITT Recommendation T.51 (1992), *Latin Based Coded Character Sets for Telematic Services*.

  ISO/IEC 6937:1994, *Information technology – Coded graphic character set for the communication of texts using the Latin alphabet*.

2.2 Additional references

– ISO 2022:1986, *Information processing – ISO 7-bit and 8-bit coded character sets – Code extension techniques*.

– ISO 2375:1985, *Data processing – Procedure for registration of escape sequences*.

– ISO/IEC 7350:1991, *Information technology – Registration of repertoires of graphic characters from ISO/IEC 10367*.

– ISO 8859-1:1987, *Information processing – 8-bit single-byte coded graphic character sets – Part 1: Latin Alphabet No. 1*.

– ISO 9293:1987, *Information processing – Volume and file structure of flexible disk cartridges for information interchange*.

– ISO/IEC TR 10000-1:1992, *Information technology – Framework and taxonomy of International Standardized Profiles – Part 1 : Framework*.

– ISO/IEC TR 10000-2:1992, *Information technology – Framework and taxonomy of International Standardized Profiles – Part 2: Taxonomy of OSI Profiles*.

– CCITT Recommendation T.400 (1988), *Introduction to Document Architecture, Transfer and Manipulation*.

– ISO/IEC ISP 10610-1:1993, *Information technology – International Standardized Profile FOD11 – Open Document Format: Simple document structure – Character content architecture only*.

3 Definitions

For the purposes of this Profile, the following definitions apply.

The definitions given in CCITT Rec. T.411 | ISO 8613-1 are applicable to this Profile.

constituent constraint names: Each constituent that may be included in a document that conforms to this Profile has been given a unique name which serves to associate that constituent with a constituent constraint defined in this Profile.

The convention is that full names are used (i.e. no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent constraint names used in this Profile are BodyText and RectoPage.

In clause 6, each constituent constraint provided by this Profile is italicized once at the point in the text at which the purpose of that constituent constraint is defined. This also serves to identify all the constituent constraints provided by this Profile.

The same constituent constraint names are also used in the technical specification in clause 7 so that there is a one-to-one correspondence between the use of these names in clauses 6 and 7.

Although the constituent constraint names relate to the purpose of the constituent constraints, the semantics of constituents shall not be implied from the names that are used. Also, these names do not appear in an interchanged document but a mechanism for associating constituents in a document with constituent constraints is provided (see 6.6.1). Thus, in an application using this Profile, the constituents may be known to the user by different names.

4 Relationship with other Profiles

This Profile belongs to a series of hierarchically related Profiles which include PM-26 and PM-36.

The features supported by this Profile are a subset of the features supported by the Profiles PM-26 and PM-36 and thus, all data streams conformant to this Profile are also conformant to PM-26 and PM-36, apart from the document application profile identifier.

NOTE – This Profile is technically aligned with (but not identical to) the Specification defined in the ISO International Standardized Profile FOD11, except that FOD11 not only defines the use of the ODIF interchange format, but also the use of the SDIF interchange format.

5 Conformance

In order to conform to this Profile, a data stream representing a document shall meet the requirements specified in 5.1.

This Recommendation does not define implementation or service requirements.

5.1 Data stream conformance

The following requirements apply to the encoding of data streams which conform to this Profile:

a) the data stream shall be encoded either in accordance with the ASN.1 encoding rules defined in CCITT Rec. X.209 | ISO/IEC 8825;

b) the data stream shall be structured in accordance with the interchange formats defined in clause 8;

c) the document, as represented by the data stream after resolution of any external references, shall be structured in accordance with one of the documents architecture classes as defined in 6.1 and shall contain all mandatory constituents specified for that class; other constituents may be included, provided that they are permitted for that class, as specified in clause 7;

d) each constituent shall contain all those attributes specified as required for that constituent in this Profile; other attributes may be specified provided that they are permitted for that constituent;

e) the attribute values specified shall be within the range of permissible values specified in this Profile;

f) the encoded document shall be constructed in accordance with the abstract document architecture defined in CCITT Rec. T.412 | ISO 8613-2;

g) the document shall be structured in accordance with the characteristics and constraints specified in clause 6.

5.2 Implementation conformance

This subclause states the requirements for implementations claiming conformance to this Profile.

A conforming receiving implementation shall be capable of receiving either any data streams conforming to this Profile structured in accordance with ODIF. Receiving usually, but not always, involves recognizing and further processing the data stream elements.

6 Characteristics supported by this document application profile

This clause describes the characteristics of documents which may be represented by data streams conforming to this Profile. This clause also describes how these characteristics are represented in term of constituent constraints.

6.1 Overview

6.1.1 General

This Profile supports the interchange of documents in the following forms:

– processable form, which facilitates the revision of a document by a recipient;

– formatted form, which facilitates the reproduction of a document as intended by the originator;

– formatted processable form, which facilitates the reproduction of a document as intended by the originator or facilitates the revision of a document by a recipient.

The constituents that make up these three classes of data stream are defined in 6.1.2, 6.1.3 and 6.1.4. Constituents defined as “required” shall occur in any data stream that conforms to this Profile. Constituents listed as “optional” may or may not be present in the data stream depending on the requirements of the particular data stream.

6.1.2 Formatted form documents

Required constituents:

– a document profile;

– layout object descriptions representing a specific layout structure;

– content portion descriptions associated to the basic objects in the specific layout structure.

Optional constituents:

– layout object class descriptions representing a factor generic layout structure;

– presentation styles.

6.1.3 Processable form documents

Required constituents:

– a document profile;

– logical object class descriptions representing a complete or partial generic logical structure;

– logical object descriptions representing a specific logical structure;

– content portion descriptions associated to the basic objects in the specific logical structure.

Optional constituents:

– layout object class descriptions representing a complete generic layout structure;

– layout styles;

– presentation styles;

– content portion descriptions associated to the basic object classes in the generic logical structure.

In the case of processable form documents, when the generic layout structure is not present, additional restrictions are placed on the layout directives that may be included in layout styles. These restrictions are defined in 6.4.3.

Note that when the generic layout structure is present, a layout style is required by the constituent constraint of the type Passage.

6.1.4 Formatted processable form documents

Required constituents:

– a document profile;

– logical object class descriptions representing a complete or partial generic logical structure;

– logical object descriptions representing a specific logical structure;

– layout object class descriptions representing a complete generic layout structure;

– layout object descriptions representing a specific layout structure;

– content portion descriptions associated to the basic objects in the specific logical/layout structure;

– layout styles.

Optional constituents:

– presentation styles;

– content portion descriptions associated to the basic object classes generic logical structure.

6.2 Logical characteristics

6.2.1 Introduction

This subclause defines the logical constituent constraints provided by this Profile to represent the characteristics of documents containing logical component descriptions.

Different constituent constraints may be used to represent and distinguish parts of a document that have different logical characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the logical characteristics represented by each of the constituent constraints are provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.2.2 Overview of the logical structure

From the logical point of view, the document consists of two parts, namely a body part and a common part.

The body part represents the main content of a document and is intended to be reproduced in the body area of the pages that make up the document. The body part shall be included in all documents that are interchanged in accordance with this Profile.

The common part represents common content that is to be placed in reserved header and footer areas on each page of a document. Header and footer content are independently optional and so may be included in an interchanged document only if required.

6.2.3 Body part of the logical structure

6.2.3.1 DocumentLogicalRoot

DocumentLogicalRoot is a constituent constraint representing the top level in the document logical structure. Its immediate subordinates consist of a sequence of one or more constituent constraints of the type Passage.

6.2.3.2 Passage

Passage is a constituent constraint that represents the first level of logical subdivision of a document. It may be used to indicate a logical grouping of subordinate parts of a document that are to be regarded as an entity for reading or that have common layout and presentation characteristics.

Passages are typically used to represent:

- the contents to be placed on the title page of a report;
- the front matter in the table of contents or foreword;
- the main matter of the document;
- the back matter, consisting of appendices, glossary or index.

The immediate subordinates of a Passage consist of a sequence of one or more constituent constraints of the type BodyText.

A document contains only one class definition of the type Passage, which defines the common characteristics of sets of Passages within the document such as layout properties. For example, when the generic layout structure is present, Passage shall be entirely laid out in the pages of one page set.

6.2.3.3 BodyText

BodyText is a constituent constraint which represents the lowest level of logical subdivision of a document. This constituent constraint is a subdivision of Passages. This allows the layout and presentation requirements of different parts of the document to be specified.

This is a basic logical constituent that directly refers to content portions that contain character content. BodyText in the specific logical structure shall refer to one or more content portions each containing processable, formatted or formatted processable character content. However, this constituent in the generic logical structure shall not refer to generic content.

6.2.4 Common content part of the logical structure

6.2.4.1 CommonContent

CommonContent is a constituent constraint that represents common content that is to be laid out in the header and footer areas of the pages of a document. Common content consists of character content.

Any number of constituent constraints of the type CommonContent may be contained in a document. CommonContent is a composite logical object class whose immediate subordinates consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

– CommonText;

– PageNumber.

When the generic layout structure is present, constituents of the type CommonContent and their associated subordinate constituents are constrained to be laid out in frames representing header or footer areas using the "logical source" mechanism (see 6.3.6).

6.2.4.2 CommonText

CommonText is a constituent constraint that represents the common character content that is to be laid out in the header or footer area of a document. For example, header or footer content that appears on each page in a sequence of pages may be represented by this constituent.

CommonText is a constituent constraint for a basic logical object class that references one content portion containing processable, formatted or formatted processable character content.

6.2.4.3 PageNumber

PageNumber is a constituent constraint that represents the common character content that is to be laid out in the header or footer area of a document. This constituent is specifically used when it is required to present a header or footer content which contains an automatically generated page number.

PageNumber is a basic logical object class that contains a content generator. This content generator contains a reference to a page number which is automatically evaluated when the document is laid out. This provides the means of representing the page numbers that are displayed on the consecutive pages of a document.

Each page number consists of a single number which may be represented in the form of Arabic or Roman numerals or in its alphabetic equivalent. Page numbering schemes may start at 0 or any value greater than 0 at the document root or page set level.

The format of the content generators is defined in 6.6.3.

6.3 Layout characteristics

This subclause defines the constituent constraints for layout components which are provided by this Profile to represent the characteristics of documents.

Different constituent constraints may be used to represent and distinguish parts of a document that have different layout characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the layout characteristics represented by each of the constituent constraints are provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.3.1 Overview of the layout characteristics

The document structure allows the document content to be laid out and presented in one or more page sets. Each page set may be used for different parts of the document, for example, the title page, foreword, table of contents, document body and appendices.

Each page set consists of a series of pages. In general, each page may be subdivided into three areas; the body area, which is used to layout the document body; and the header and footer areas, which may be used to layout the common content.

Page layout type supported by this Profile is used when the character content is to be laid out horizontally (from left to right or from right to left) and from top to bottom within the body area, the header area and footer area. Portrait and landscape orientations of this page layout are illustrated in Figures 1 and 2 respectively.

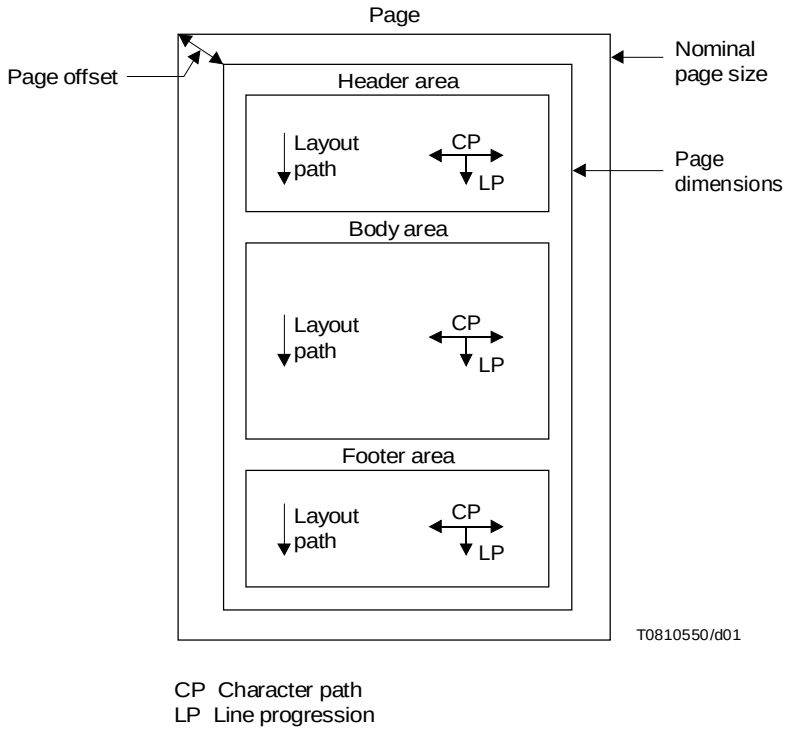

FIGURE 1/T.502

Page layout (portrait orientation)

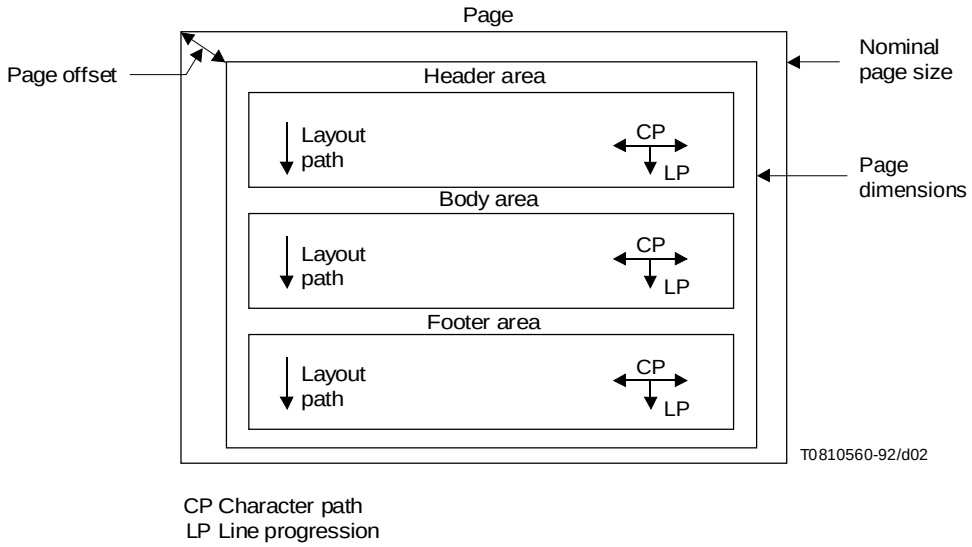

FIGURE 2/T.502

Page layout (landscape orientation)

6.3.2 DocumentLayoutRoot

DocumentLayoutRoot is a constituent constraint represents the top level in the document layout structure. Its immediate subordinates consist of a sequence of one or more constituent constraints of the type PageSet. The numbering schemes for pages may be initialized on this constituent constraint.

6.3.3 PageSet

PageSet is a constituent constraint that represents a grouping of pages within a document. A PageSet is typically used to represent a part of a document that has different layout requirements from other parts of a document. Also, a PageSet may correspond to a part of a document that has a certain logical significance, for example, a PageSet might represent the front matter in a document or an individual chapter.

Only one level of PageSet is allowed in a document. However, a document may contain any number of class definitions of type PageSet which may be used, for example, to provide a choice of alternative layouts for different parts of a document or to specify the exact layout requirements for each successive part of a document.

The immediate subordinates of a PageSet consist of a combination of constituent constraints of the types Page, RectoPage and VersoPage as described in 6.3.4.1.

6.3.4 Page characteristics

6.3.4.1 Page constituents

Three constituent constraints are provided to represent the pages within a document, namely *Page*, *RectoPage* and *VersoPage*.

The only difference in the characteristics of these types of constituent constraints concerns the values that may be specified for the parameter “side of sheet” in the attribute “medium type”. In the case of Page, the value of this parameter may be specified as 'recto', 'verso' or 'unspecified'. In the case of RectoPage, the value of this parameter may be specified as 'recto' or 'unspecified'. In the case of VersoPage, the value of this parameter may be specified as 'verso' or 'unspecified'. The values 'recto' and 'verso' of the “side of sheet” parameter of the “medium type” attribute are non-basic.

The pages that make up a page set consist of an optional initial page which is represented by the constituent constraint Page and which is optionally followed by either:

a) A sequence of pages represented by the constituent constraint Page. All pages in this sequence shall have the same layout characteristics but these characteristics may differ from those of the initial page.

b) A sequence of pages which are intended to be laid out alternatively on the 'recto' and 'verso' (or on the 'verso' and 'recto') sides of the presentation medium and are represented by the constituent constraints RectoPage and VersoPage respectively. All pages in this sequence shall have the same layout characteristics but these characteristics may differ from those of the initial page.

Pages having the same layout characteristics are pages that have the same page layout (see 6.3.4.5) and for which the body area, header area (if present) and footer area (if present) have the same dimensions and positions within the page (see 6.3.4.3). Pages having the same layout characteristics do not necessarily have the same position on the presentation medium (see 6.3.4.4).

A page set shall contain at least one page.

An initial page is typically used at the beginning of a document or of a section within a document. It may be used, for example, for a title page whose layout requirements differ from the following pages.

The following restrictions also apply to the pages within a page set:

– all the pages must have the same dimensions and orientation (see 6.3.4.2);

– all pages are to be laid out on the same size of presentation medium (see 6.3.4.3).

6.3.4.2 Page dimensions

The dimensions of the pages may be specified as any value (in BMUs) that is equivalent to or less than ISO A3 or ANSI B paper sizes in portrait or landscape orientation. The dimensions may be specified in portrait or landscape orientation. Japanese page sizes B4 and B5 are also supported but the dimensions of these pages lie within the range of dimensions given above.

Dimensions equivalent to or less than the common assured reproduction area of ISO A4 and ANSI A in portrait or landscape orientation are basic values. Larger page sizes are non-basic and their use shall be indicated in the document profile.

Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above.

NOTE – The size termed "North American Letter (NAL)" in CCITT Rec. T.410-Series | ISO 8613 (e.g. in CCITT Rec. T.412 | ISO 8613-2, clause 7) is in this Specification called "ANSI A" in order to be consistent with the other reference to ANSI standard paper sizes.

6.3.4.3 Nominal Page sizes

The nominal page sizes that may be specified are listed in Table 1. These may be specified in portrait or landscape orientation. All values of nominal page size are non-basic and hence all values used in a document must be indicated in the document profile.

Any value of nominal page size defined in Table 1, subject to the restrictions specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended assured reproduction area (ARA). Information loss may occur when a document is reproduced if the dimensions of constituent constraints of the type page exceed the ARA for the specified nominal page size.

TABLE 1/T.502

Nominal page sizes

Page type	Size in inches or millimetres	Size in BMUs	ARA in BMUs
ISO A5	148 mm × 210 mm	7 015 × 9 920	Not defined
ISO A4	210 mm × 297 mm	9 920 × 14 030	9 240 × 13 200
ISO A3	297 mm × 420 mm	14 030 × 19 840	13 200 × 18 480
ANSI legal	8.5 × 14 inches	10 200 × 16 800	9 240 × 15 480
ANSI A	8.5 × 11 inches	10 200 × 13 200	9 240 × 12 400
ANSI B	11 × 17 inches	13 200 × 20 400	12 744 × 19 656
Japanese legal	257 mm × 364 mm	12 141 × 17 196	11 200 × 15 300
Japanese letter	182 mm × 257 mm	8 598 × 12 141	7 600 × 10 200

6.3.4.4 Page offset

The page offset is the distance of the position of the left and top edges of the page relative to the left and top edges respectively of the presentation medium on which each page is reproduced. Any value of page offset may be specified provided that no part of the page area lies outside the area of the nominal page. Also, page offsets specified for the initial, recto and verso pages within a given page set may differ. The default page offset may be specified in the document profile.

6.3.4.5 Page layout characteristics

Each page in a document may be subdivided into three rectangular areas, as follows:

- a body area which is reserved for content that belongs to the body part of the document (see 6.3.5);
- a header area which is reserved for common header content (see 6.3.6);
- a footer area which is reserved for common footer content (see 6.3.6).

The body area is mandatory and shall occur on every page in a document. The header and footer areas are both optional.

Also, these three areas shall be entirely contained within the page area and shall not overlap.

For the style of page layout supported by this Profile, the header and footer areas are placed above and below the body area respectively. The layout paths in the header, body, and footer area always have a value of 270 degrees, as shown in Figures 1 and 2. As this Profile only supports a layout path of 270 degrees, which is the standard default value specified in CCITT Rec. T.412 | ISO 8613-2, the layout path shall not be explicitly specified in a document.

6.3.5 Body area characteristics

6.3.5.1 General characteristics

The body area is the area within a page where the main matter of the document, that is the body part of the document, is laid out.

The body area consists of a single frame into which the content is directly laid out. This body area is represented by a BasicBody frame.

6.3.5.2 BasicBody

BasicBody is a constituent constraint which defines a lowest level frame into which the content is directly laid out.

The position and dimensions of this frame are fixed. The layout path of BasicBody is implicitly specified as 270 degrees (see 6.3.4.5).

6.3.6 Header and footer area characteristics

6.3.6.1 General characteristics

The header area consists of a basic area and the footer area consists of a basic area.

A basic header or footer area is an area into which the content is directly laid out. This type of area is represented by a constituent constraint of the type BasicHeader or BasicFooter respectively.

The content allocated to these areas is derived from the common part of the logical structure of a document.

6.3.6.2 BasicHeader and BasicFooter

BasicHeader and *BasicFooter* are constituent constraints that define lowest level frames that represent areas within a page that are reserved for common content.

These types of frame have fixed positions and dimensions. The layout path of these frames is implicitly specified as 270 degrees (see 6.3.4.5).

The content that is laid out in these frames is derived, using the logical source mechanism, from the content associated with the composite logical object classes of the type CommonContent.

6.3.7 SpecificBlock

SpecificBlock is a constituent constraint that defines a specific block.

Objects of the type SpecificBlock may only occur in the specific layout structure. They are created during the document layout process and result from the layout of basic logical objects into lowest level frames that constitute the body, header and footer areas.

Each SpecificBlock in a BasicBody frame shall refer to one content portion. A SpecificBlock in a BasicHeader frame or BasicFooter frame may refer to one or more content portions.

6.4 Document layout characteristics

Mechanisms for controlling the allocation of logical constituents to various areas in the layout structure are defined in 6.4.1. Mechanisms for controlling the layout of the content within the allocated areas are defined in 6.4.2.

These mechanisms relate to documents for which a generic layout structure is specified. When a generic layout structure is not present, then these mechanisms are restricted as described in 6.4.3.

6.4.1 Flow controls

Various mechanisms are provided to control the allocation of constituent constraints representing the body parts of the logical structure of a document to page sets, pages and body areas. These are described in 6.4.1.1, 6.4.1.2 and 6.4.1.3. The mechanisms for controlling the layout of the common parts of a document are described in 6.4.1.4.

6.4.1.1 Allocation of content to page sets

The following method of allocating the constituent constraint associated with the body part of the document to page sets is provided.

This provides the ability to specify that a part of a document is to be laid out entirely within a specified page set. This shall be specified for a constituent constraint of the type Passage using the attribute “layout object class” which specifies the object class identifier of the required class of page set.

6.4.1.2 Allocation of content to page

The following method of allocating the constituent constraint associated with the body part of the document to pages is provided.

6.4.1.2.1 New layout object

New layout object provides the ability to specify that a particular logical constituent constraint in a document is to be laid out starting at the beginning of a new page. The page specified must belong to the page set in which the immediate preceding logical constituent constraint is laid out.

This may be specified for the logical constituent constraint of the type BodyText.

This is achieved using the attribute “new layout object”. This attribute may specify the value 'page' indicating that the logical constituent constraint is to be laid out starting on the next available page which may be of any class. Alternatively, the attribute may specify that the logical constituent constraint is to be laid out starting on a page of a particular class; this is achieved by specifying the object class identifier of the required page class.

The specification of a page break must not be used to layout part of a document in a new page set. If a new page set is required, then this should be explicitly specified as described in 6.4.1.1.

6.4.1.2.2 Indivisibility

Indivisibility provides the means to specify whether or not a logical object derived from a basic or composite logical constituent constraints is allowed to be split over more than one page. It may be specified for logical constituent constraints of the types Passage and BodyText. The attribute “indivisibility” is used to specify this feature.

6.4.1.2.3 Same layout object

Same layout object provides the means to specify that the start of the content associated with a logical object and the end of the content associated with the previous logical object are to be laid out within a single page. This may be specified for basic logical objects of the type BodyText. The attribute “same layout object” is used to specify this feature.

6.4.1.3 Allocation of content to body areas

The page to which the content is allocated contains a basic body area. The content is laid out in sequential order in that body area in the form of a single column.

6.4.1.4 Allocation of content to header-footer areas

The frame representing a basic header or footer area (see 6.3.6) specifies the attribute “logical source” which indicates the particular instance of the logical constituent constraint of the type CommonContent (see 6.2.4.1) that is to be laid out in that area. The basic logical constituents subordinate to CommonContent are then laid out in accordance with their sequential order.

6.4.1.4.1 Concatenation

Concatenation provides the means to specify that the content associated with a logical object derived from a basic logical constituent constraint and the content associated with the logical object derived from the previous basic logical constituent constraint are to be regarded as an unbroken stream of content. This may be specified for basic logical constituent constraints of the type CommonText and PageNumber. The attribute "concatenation" is used to specify this feature.

6.4.2 Layout of the document content

Various constraints may be specified to control the layout of the content into the body, header and footer areas. These constraints are described below.

6.4.2.1 Margins

The margins are the minimum distances, or offsets, between a part of the document content and the edge of the particular area in which that content is laid out. The margins define the maximum extents of the available area into which the content shall be positioned.

Margins may be specified for the basic logical constituent constraints of the type BodyText, CommonText and PageNumber; different margin values may be specified for different basic logical constituent constraints without restriction.

Four margins may be independently specified for each logical constituent constraint (see Figure 3), namely:

- trailing edge margin;
- leading edge margin;
- right hand edge margin;
- left hand edge margin.

Any combination of the above margins may be specified for a particular logical constituent constraint. These margins are specified by the attribute "offset". Any value may be specified in units of BMUs. If a particular margin is not specified then it is assumed to be 0 BMUs.

6.4.2.2 Separation

Leading separation is the minimum distance between one basic logical object and the next one, if any, when they are laid out. Trailing separation is the minimum distance between one basic logical object and the previous one, if any, when they are laid out. Both may be specified for basic logical components of the constituent constraint types BodyText, CommonText and PageNumber. These distances are specified in BMUs by the attribute "separation". If no value is specified, then the minimum distance is assumed to be 0 BMUs.

6.4.3 Layout controls applicable in the absence of a generic layout structure

In processable form documents the generic layout structure is optional. If the generic layout structure is omitted, then it is the responsibility of the receiver to define an appropriate layout structure. No limitations are placed on the layout structure that is used.

When a generic layout structure is not specified within a processable form document, then restrictions are placed on the layout control functions described in 6.4.1 and 6.4.2 that may be specified within the document. These restrictions are indicated below:

- It is not possible to specify that certain logical parts of a document are to be allocated to a given page set as defined in 6.4.1.1.
- It is possible to specify page break as defined in 6.4.1.2.1 but it is only possible to indicate that the layout should begin on a new page. It is not possible to specify a particular page class.

– The logical parts of the document that are intended to be laid out in the body area and in the header/footer areas of each page may be distinguished from each other by means of application comments specified for them (see 6.6.1). An exception is that it is not possible to distinguish whether a particular portion of the common content is to be placed in a header or a footer area (or both).

– Margins and separation as defined in 6.4.2 may all be specified. Indivisibility as defined in 6.4.1.2.2 and same layout object as defined in 6.4.1.2.3 may all be specified. Concatenation as defined in 6.4.1.4.1 may all be specified.

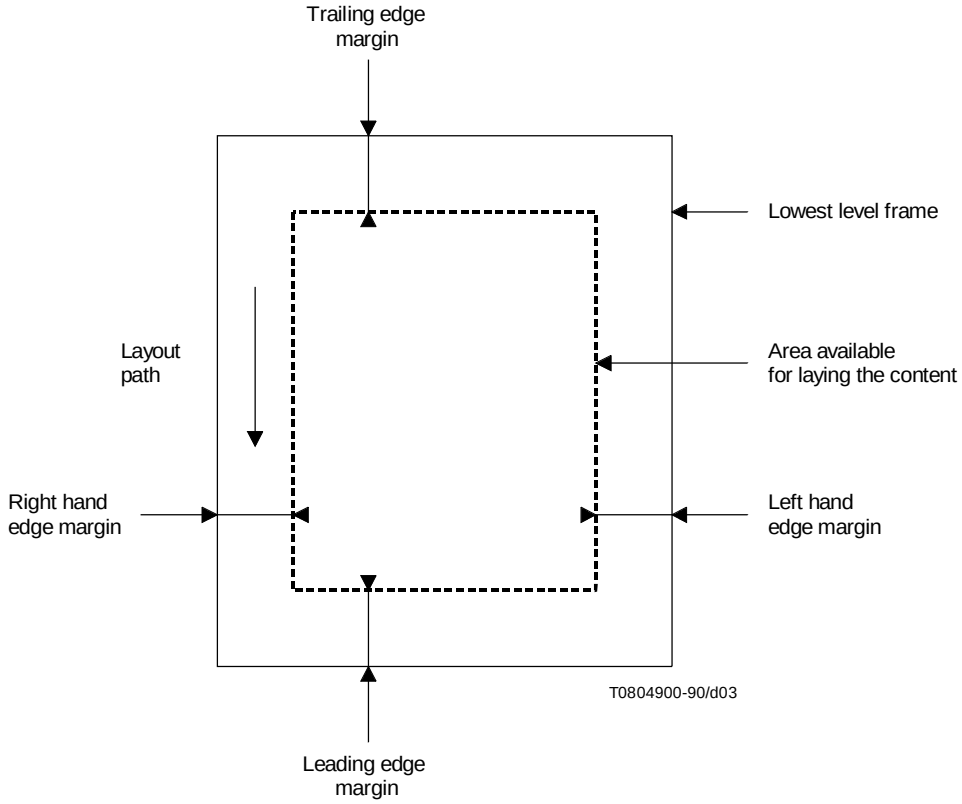

FIGURE 3/T.502

Specification of margins

6.5 Content layout and imaging characteristics

A document contains character content.

The content architectures that may be specified using the attribute “content architecture class” are formatted character, processable character and formatted processable character. Any of these may be specified as the default in the document profile.

6.5.1 Introduction

This subclause defines the features that are applicable to the character content contained in a document and the presentation attributes and control functions that may be used to specify these features. These features may apply to basic logical and layout components unless otherwise indicated.

The default values for the following features may be specified in the document profile:

- graphic character sets;
- graphic character subrepertoire;
- code extension announcers;
- line spacing;
- character spacing;
- character path;
- line progression;
- graphic rendition, including the parameter values:

  default rendition, increased intensity (bold), italicized, underlined, crossed-out, normal intensity, not italicized, not underlined, not crossed-out;
- line layout table;
- indentation;
- alignment;
- first line offset;
- itemization;
- widow size;
- orphan size;
- initial offset.

The specification in a document of a non-basic feature by a presentation attribute or control function shall be indicated in the document profile.

6.5.2 Character content architecture class

Processable and formatted processable form documents may contain processable, formatted or formatted processable character content. Formatted form documents may contain formatted or formatted processable character content.

6.5.3 Character repertoire

The basic character repertoire supported by this Profile is composed of the 94 characters of ISO-IR 6 (the IRV of ISO/IEC 646) plus the character space.

Any other graphic character set which is registered in accordance with ISO 2375 may be designated and invoked at any point in the document provided its use is indicated in the document profile as a non-basic value using the character presentation feature “graphic character sets”. No locking shift functions are specified in this presentation feature.

The code extension techniques allowed for the designation and invocation of character sets to the left hand side and right hand side of the 8-bit code table (GL and GR respectively) are defined in 6.5.4.

Using these code extension techniques, the graphic character sets designated and/or invoked at the beginning of a content portion containing character content are specified by the presentation attribute “graphic character sets”. The graphic character sets may be changed at any point within a content portion.

The default graphic character sets which apply to the content portions within a document may be specified in the document profile using the presentation attribute “graphic character sets”.

If the character set defined in ISO 6937 is designated and invoked, then the use of any of its subrepertoires registered according to ISO/IEC 7350 may be specified using the presentation attribute “graphic character subrepertoire”. All subrepertoires are non-basic and their use must be indicated in the document profile. The subrepertoire shall not be changed within a content portion.

NOTE – The basic character repertoire supported by this Profile is not the standard default value specified in CCITT Rec. T.416 | ISO 8613-6; hence it may be necessary to specify, in the document profile of a particular document, that this is the default value being used for that document.

6.5.4 Code extension techniques

The code extension techniques specified in ISO 2022 may be used subject to the following restrictions:

a) G0 set – Only ISO-IR 6 (the IRV of ISO/IEC 646), ISO-IR 2 (the primary set of ISO 6937-2), or any other version of ISO 646 may be designated for this set; these graphic character sets may only be invoked in GL.

b) G1, G2, G3 sets – No restrictions are placed on the character sets that may be designated for these sets; these graphic character sets may only be invoked in GR.

c) The locking and single shift functions allowed are as follows:

   – LS0 to invoke the G0 set into GL;

   – LS1R to invoke the G1 set into GR;

   – LS2R to invoke the G2 set into GR;

   – LS3R to invoke the G3 set into GR;

   – SS2 to invoke one character from the G2 set into GL;

   – SS3 to invoke one character from the G3 set into GL.

   (Here GL and GR refer to the left and right hand parts respectively of the 8-bit code table.)

d) When specifying the presentation attribute "graphic character sets", it is necessary to invoke character sets for both GL and GR. Thus, an allowed character set shall be designated into G0 [see item a) above] and invoked into GL. It is also necessary to invoke a character set into GR which has been designated into G1, G2 or G3 set.

e) The empty set shall be designated into G1 and invoked into GR if no other specific character set is invoked in GR.

The code extension techniques allowed are illustrated in Figures 4 and 5.

The announcement and encoding of these functions are to be as specified in ISO 2022.

The code extension techniques that are used or may be used in a basic component shall be specified by the presentation attribute "code extension announcers". The default code extension announcers used throughout a document may be specified in the document profile using the presentation attribute "code extension announcers".

NOTE – In accordance with CCITT Rec. T.416 | ISO 8613-6, there is no restriction concerning the number of graphic character sets which may be designated and/or invoked in the presentation attribute "graphic character sets" providing the restrictions defined in this subclause are applied. Hence designation to a particular G set overrides the previous designation to that set, and invocation to GL or GR overrides the previous invocation to GL or GR respectively. Thus, the sequential order of designation and/or invocation sequences in the attribute "graphic character sets" is significant.

6.5.5 Line spacing

Line spacing values of 100, 150, 200, 300 and 400 BMUs may be specified. The values of 200, 300 and 400 BMUs are basic; the use of any other value in a document is non-basic and shall be indicated in the document profile.

The line spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute "line spacing". The value may be changed anywhere within the content portion using the control function SVS.

6.5.6 Character spacing

Character spacing values of 80, 100, 120, 160 and 200 BMUs may be specified. The value of 120 BMUs is basic; the use of any other value in a document is non-basic and shall be indicated in the document profile.

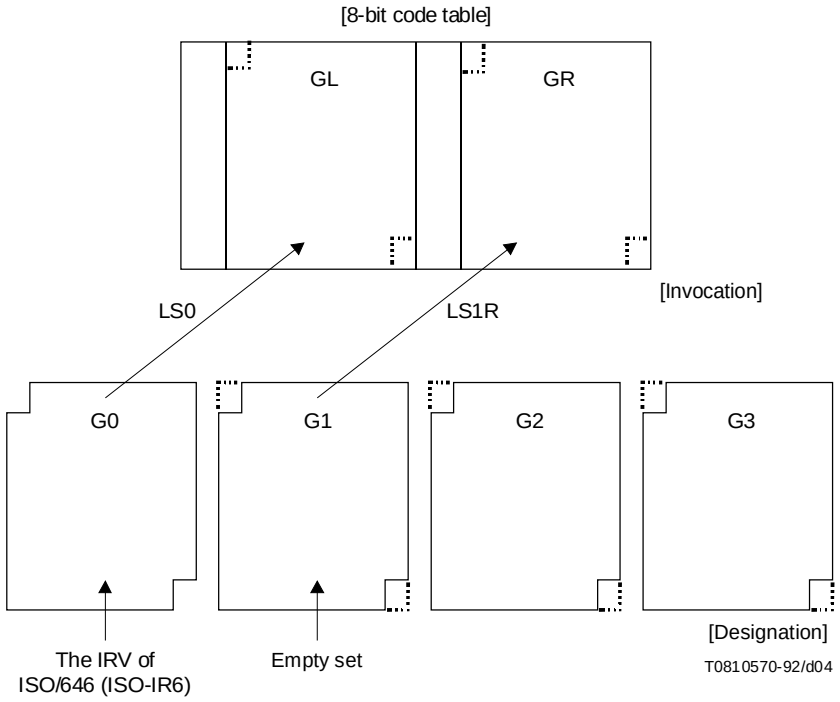

FIGURE 4/T.502

Code extension features (basic case)

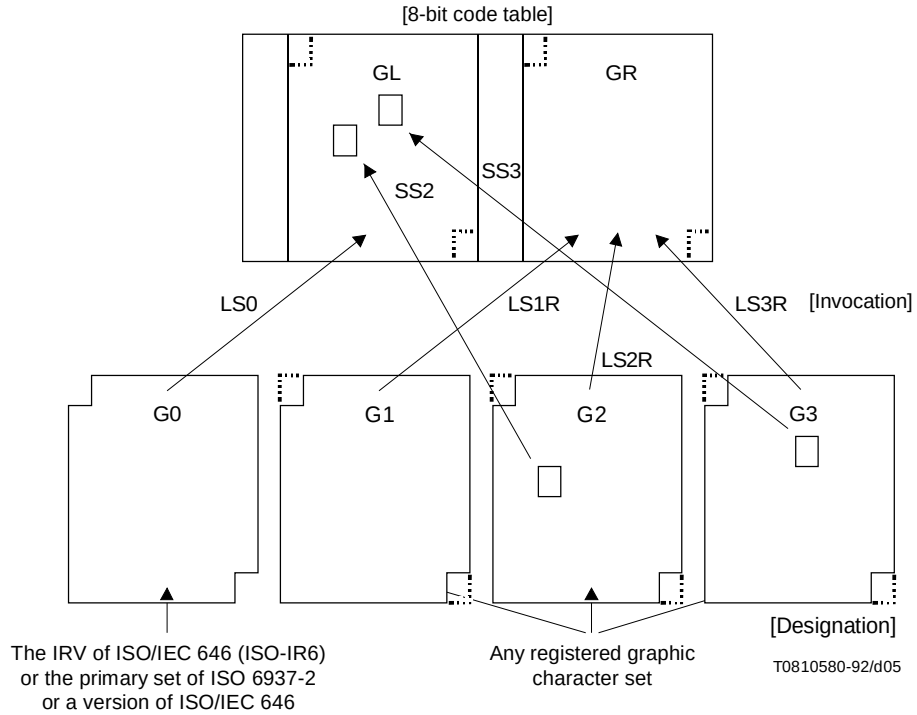

FIGURE 5/T.502

Code extension features (all possible cases)

The character spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute “character spacing”. The value may be changed anywhere within the content portion using the control functions SHS and SCS.

NOTES

1 A character spacing value of 160 BMUs is provided for use with Korean Hangul characters.

2 SHS parameters 0, 1, 2 and 3 are provided. The use of parameters 5 and 6 may be provided in a future edition of this Profile for use with Chinese characters.

6.5.7 Character path and line progression

Both from left to right and from right to left horizontal writing directions may be specified within a document. The line progression is then from top to bottom only on the page.

Character path values of 0 degrees and 180 degrees may be specified. The value of 0 degrees is basic. The value of 180 degrees is non-basic and must be indicated in the document profile.

The values of character path may be specified at the beginning of the content associated with a basic component using the presentation attribute “character path”. The value shall not be changed within a content portion.

The line progression supported by this Profile is 90 degrees and 270 degrees. The value of 270 degrees is basic. The value of 90 degrees is non-basic and must be indicated in the document profile. The line progression value of 270 degrees may be specified in case of the character path 0 degrees, and the line progression value of 90 degrees may be specified in case of the character path value of 180 degrees.

6.5.8 Character orientation

The character orientation supported by this Profile is 0 degrees only which is the standard default value specified in CCITT Rec. T.416 | ISO 8613-6. Thus, the character orientation shall not be specified.

6.5.9 Emphasis

The following modes of emphasising graphic characters may be specified as basic:

- default rendition;
- normal intensity;
- increase intensity (bold);
- italicized;
- not italicized;
- underlined;
- not underlined.

The following modes of emphasising graphic characters may be specified as non-basic:

- crossed-out;
- not crossed-out.

The above mentioned non-basic modes shall be indicated in the document profile. If no default mode is explicitly specified in the document profile, then the default mode is default rendition.

The mode of emphasis may be specified at the beginning of the content associated with a basic component using the presentation attribute “graphic rendition”. The mode may be changed anywhere within the content using the control function SGR.

The mode of emphasis remains in effect within the content associated with a basic component until changed into a mutually exclusive mode or by the specification of 'default rendition'. Mutually exclusive modes are normal/increased intensity, italicized/not italicized, underlined/not underlined and crossed-out/not crossed-out. One mode from each mutually exclusive set may be in operation at any point in the document content.

Default rendition cancels the effect of all modes of emphasis that are currently in operation and specifies that the text shall be displayed in accordance with the default rendition parameters set for the presentation device. Thus, for example, if it is required to ensure that the content is not underlined, then it is necessary to explicitly specify that underlined is not to be used.

6.5.10 Tabulation

Tabulation stop positions may be specified at any character position along the character path. Each stop is specified by means of the following:

a) The tabulation position, relative to the margin position is in the direction opposite to the character path.

b) An optional alignment qualifier that specifies the type of alignment to be used at the designated tabulation position. The type may be specified as one of the following:

- start aligned;
- end aligned;
- centred;
- aligned around.

These alignment qualifiers are defined in CCITT Rec. T.416 | ISO 8613-6. If the alignment qualifier is not explicitly specified, then it is assumed that start aligned is to be used.

Only one set of tabulation stops can be specified to be applicable to the content associated with a basic component. No limit is placed on the number of tabulation stops that may be specified within a given set.

The set of tabulation stop positions associated with the content of a basic component are specified using the presentation attribute “line layout table”. Tabulation stop positions are invoked within the content using the control function STAB.

The tabulation reference numbers used in the control function STAB and associated presentation attribute “line layout table” shall be chosen so that, in any given line layout table the reference numbers are unique, sequential in the direction of the character path and do not include leading zeroes.

6.5.11 Indentation

Indentation is the distance between the first character on a line of content and the position of the margin that is in the direction opposite to the character path. Thus, the value of indentation specified determines the line home position (as defined in CCITT Rec. T.416 | ISO 8613-6).

Indentation acts as temporary alternation in the position of the offset in the direction opposite to the direction of the character path. When text is formatted, it is intended to be laid out between the indentation position and the margin position in the direction of the character path.

Any value of indentation may be specified for basic logical components using the presentation attribute “indentation”. The indentation value shall not be changed within a content portion.

6.5.12 Alignment

This feature is concerned with how the first and last characters on each line of character content are to be laid out during the formatting process.

The following values of alignment may be specified as basic:

- start aligned;
- end aligned;
- centred;
- justified.

The semantics of these values are as defined in CCITT Rec. T.416 | ISO 8613-6.

The presentation attribute “alignment” is used to specify the alignment that is applicable to the content associated with a basic component. The alignment value cannot be changed within a content portion.

6.5.13 First line format

This feature specifies how the first line of the content associated with a basic component is to be laid out and provides for the itemization of paragraphs.

It allows the first character in the content to be positioned at some point along the character path relative to the indentation position (as defined in 6.5.11). This point may be in the direction of the character path or in the direction opposite to the direction of the character path relative to the indentation position.

In addition, this feature provides for the specification of an item identifier on the first line. The item identifier is a string of characters that precedes and is separated from the remaining characters that form the first line. The control function CR (Carriage Return) is used as the separator.

The features provided correspond to examples 10.1 to 10.5 shown in Figure 10 of CCITT Rec. T.416 | ISO 8613-6.

First line format is specified by the presentation attributes “first line offset” and “itemization”, and “indentation”. Only those values of the attributes which combine to form the examples shown in Figure 10 of CCITT Rec. T.416 | ISO 8613-6 may be used.

6.5.14 Widow and orphan sizes

The widow size specifies the minimum number of lines of content that shall be allocated to a following frame or page when the content associated with a basic logical component is laid out such that it flows over two frames or pages. To accommodate this, it may be necessary to move a number of lines of content from one frame or page to the next frame or page.

The orphan size specifies the minimum number of lines of content that shall be placed in the current frame or page when the content associated with a basic logical component is split over two frames or pages. If this minimum cannot be accommodated, then the whole content shall be placed in the next frame or page.

Any value of widow or orphan size may be specified using the presentation attributes “widow size” and “orphan size” respectively.

Widow and orphan size may only be specified for character content placed in the body area of pages.

6.5.15 Reverse character string

Bi-directional writing is supported by this Profile (see 6.5.7). Hence, a string of characters in a content portion associated with a basic component may be specified to be imaged in the reverse direction of the immediately preceding character string. Such strings may be specified by the control function SRS as defined in CCITT Rec. T.416 | ISO 8613-6.

This control function is provided for cases in which the text belongs to different languages and the character content is written, for example, from left to right or from right to left within the same line of characters, dependent upon the language and/or character set being used.

NOTE – The use of this control function cannot be indicated in the document profile. Thus, it is intended that implementations should ignore this control function when reverse character string layout and presentation is not supported.

6.5.16 Superscripts and subscripts

Superscripts and subscripts may be specified anywhere within the content associated with a basic component by using the control functions PLU and PLD. The use of these control functions shall be in accordance with CCITT Rec. T.416 | ISO 8613-6.

6.5.17 Line breaks

The control functions BPH and NBH may be inserted in processable and formatted processable form character content to indicate where line breaks may occur or may not occur respectively, when the content is laid out.

6.5.18 Substitution of characters

The control function SUB is provided to represent characters produced by a local system that cannot be represented by a character within a character set supported by this Profile.

6.5.19 Initial point

The initial point which is applicable to basic layout components may be specified by the attribute "initial offset". Any value may be specified.

6.5.20 Use of control functions

The following is a list of all the control functions and parameter values (where applicable) that may be specified in character content:

SHS	Select character spacing (allowed parameter values: 0, 1, 2, 3)
SCS	Set character spacing (allowed parameter values: 80, 100, 120, 160, 200 BMUs)
SVS	Select line spacing (allowed parameter values: 0, 1, 2, 3, 4)
SGR	Select graphic rendition (allowed parameter values: 0, 1, 3, 4, 9, 22-24, 29)
STAB	Selective tabulation (allowed parameter values: any)
SRS	Start reverse string (allowed parameter values: any)
PLD	Partial line down
PLU	Partial line up
BPH	Break permitted here
NBH	No break here
JFY	No justify
SUB	Substitute
SP	Space
CR	Carriage return
LF	Line feed
SOS	Start of string
ST	String terminator
	Code extension control functions (see 6.5.4)

The use of all these control functions, with exception of SP, CR, LF, SOS and ST, are described in 6.5.3 through 6.5.19.

6.5.21 Formatting the content

The attribute "formatting indicator" shall not be specified within documents that are conformant with this Profile.

6.6 Miscellaneous features

6.6.1 Application comments

Specification of the attribute "application comments" is mandatory for all object classes contained in a document that conforms to this Profile. Specification of this attribute is mandatory for all objects that do not refer to an object class. Specification of this attribute is optional for all objects that refer to object classes.

This attribute is structured so that it contains two fields. The first field is mandatory when the attribute is specified and contains a numeric string which uniquely identifies the constituent constraint applicable to the constituent for which the attribute is specified. This facilitates the processing of documents. A list of these identifiers is given in Table 2.

NOTES

1 The values of the constituent constraint numeric identifiers are not unique between the logical and layout structures, and therefore in order to identify the constituent constraint applicable to a constituent, it is necessary to know the structure of which the constituent is a part.

2 For constituent constraints that correspond to each other between the hierarchically related profiles to which this Profile belongs, the same constituent constraint numeric identifier is specified.

TABLE 2/T.502

List of constituent constraint numeric identifier

Logical constituent constraints	Constituent constraint numeric identifier
DocumentLogicalRoot	0
Passage	1
BodyText	14
CommonContent	19
CommonText	20
PageNumber	40
Layout constituent constraints	Constituent constraint numeric identifier
DocumentLayoutRoot	0
PageSet	1
Page	2
RectoPage	3
VersoPage	4
BasicHeader	27
BasicBody	28
SpecificBlock	30
BasicFooter	33

The second field is optional and may contain any information that is relevant to the application or user. The format of the second field is not defined in this Profile and the interpretation of this field depends upon a private agreement between the originator and recipient of the document.

The encoding of the attribute “application comments” is defined in 8.1.3 and 8.2.3.

6.6.2 Alternative representation

The content information in a content portion may be replaced by a string of characters specified in the attribute “alternative representation”. This attribute may be specified in content portions.

The specification and use of this attribute is optional. The string of characters specified must belong to the character repertoires indicated in the document profile attribute “alternative representation character sets” (see 6.7.4.3). If the latter attribute is not explicitly specified in the document profile, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed. The control functions SP, CR and LF may also be used within the character string but no other control function is allowed; hence graphic character set cannot be changed within the alternative representation.

6.6.3 Page numbering

As described in 6.2.4.3, the constituent constraint PageNumber contains a content generator which may refer to a page number. This content generator is evaluated when the document is laid out and this mechanism provides a means of reproducing the appropriate number of each page of a document.

The content generator has the following format:

<string-literal><num-expr><string-literal>

The format of this content generator is defined in the macro PGNUMBER (see 7.3.1).

The <string-literal> fields are optional and are pre-defined character strings. The basic character repertoire used to specify these strings is ISO-IR 6 (the IRV of ISO/IEC 646). Any other character repertoire, and subrepertoire if appropriate, may be used provided that it is designated and invoked by the appropriate code designation and invocation sequences and indicated in the document profile as a non-basic value. SP and no other control functions may be used in these strings.

The field <num-expr> is a reference to a binding 'PGnum' which specifies the number of the page concerned. This binding is initialized at the document layout root or page set level (see the macro INITIALISEPGNUM in 7.4.1) and automatically incremented on each successive page (see the macro PAGENUMBER in 7.4.1). By placing initialization on the layout root, rather than on the pageset class(es), the pagenumbering may be defined to be continued from one pageset to the next.

The content associated with logical object classes of the type PageNumber is laid out in a frame of one of the following types: BasicHeader or BasicFooter (see 6.3.6) using the logical source mechanism. Thus, when the appropriate frame is being laid out, the field <num-expr> in the content generator contained in a logical object class of the type PageNumber is evaluated and this determines the value of the binding 'PGnum' that is associated with the current page being laid out.

The number associated with the binding 'PGnum' is applied to a string function during its evaluation in order to convert the number into a character string. This enables the number to be represented in the form of an Arabic numeric string, an upper or lower case Roman numeric string or an upper or lower case alphabetic string.

Each page class may refer to a different instance of logical object classes of the type PageNumber and this allows different page numbering formats to be used for different parts of the document.

An example of page numbering is "Page X" which consists of two concatenated character strings. The first is the literal character string 'Page' and this is concatenated to a string function denoted by 'X'. When 'X' is evaluated in a particular instance it may, for example, return the character string 'iv', the Roman numerical (lower case) for the number '4'.

6.6.4 User readable comments

Information which is to be interpreted as comments relevant to constituents and associated content portions may be specified using the attribute "user readable comments". This information is intended for presentation to humans.

The information consists of a string of characters which shall belong to one of the character repertoires indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed. The control functions CR, LF, SP and code extension control functions may also be used within the character string but no other control functions are allowed.

6.6.5 User visible name

Information which may be used to identify constituents within a document may be specified using the attribute "user visible name". This information is intended for presentation to humans, for example, to assist in the editing of documents.

The information consists of a string of characters which shall belong to one of the character repertoires indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed. The control functions CR, LF, SP and code extension control functions may also be used within the character string but no other control functions are allowed.

6.7 Document management features

Information relating to the document as a whole is specified in the document profile which is represented by the constituent *DocumentProfile*. This constituent must be specified in every document.

The information in the document profile is classified into the following categories:

- document constituent information;
- document identification information;
- document default information;
- non-basic characteristics information;
- document management information.

The information in the document profile may be of interest to the user or may be used for machine processing of the document.

6.7.1 Document constituent information

This information specifies which constituents are used to represent the document.

6.7.1.1 Presence of document constituents

This information indicates which constituents are included in the document. That is, this information indicates whether or not the document contains a generic logical structure, a specific logical structure, a generic layout structure, a specific layout structure, layout styles and presentation styles. It is mandatory to specify this information in the document profile.

6.7.2 Document identification information

This information relates to the identification of the document. This information is divided into six categories.

6.7.2.1 Document application profile information

This information indicates the document application profile to which the document belongs. It is mandatory to specify this information using the attribute “document application profile”.

6.7.2.2 Document architecture class information

This information indicates the document architecture class to which the document belongs (see 6.1). It is mandatory to specify this information using the attribute “document architecture class”.

6.7.2.3 Content architecture classes information

This information indicates the content architecture classes used in the document (see 6.5.2). It is mandatory to specify this information using the attribute “content architecture classes”.

6.7.2.4 Interchange format class information

This information indicates the interchange format class used to represent the document (see clause 8). It is mandatory to specify this information using the attribute “interchange format class”.

6.7.2.5 ODA version information

This information indicates the International Standard or CCITT Recommendation to which the document conforms. It also specifies a calendar date, which indicates that the document conforms to the version of the International Standard or CCITT Recommendation and any addenda that are current on that date. It is mandatory to specify this information using the attribute “ODA version”.

6.7.2.6 Document reference

This information serves to identify the document. Typically this information is allocated to the document by the creator of the document. The identifier may consist of an ASN.1 object identifier or string of characters. It is mandatory to specify this information using the attribute "document reference".

6.7.3 Document default information

This information specifies various default values for attributes used in the document. The default values that are allowed are specified in the various subclauses of clause 6. The specification of this information is only required when it is required to specify a default value which is other than the standard default value specified in CCITT Rec. T.410-Series | ISO 8613.

Default values for the following groups of attributes may be specified:

– document architecture attributes;

– character content attributes.

6.7.4 Non-basic characteristics information

This information specifies the non-basic attribute values specified in the document. It is mandatory to specify a non-basic attribute value in the document profile when such a value is used in the document.

The following types of non-basic attribute values may be specified:

– profile character sets;

– comments character sets;

– alternative representation character sets;

– page dimensions;

– medium-types;

– character presentation features.

Further information concerning document profile, comments and alternative representation character sets is given below.

6.7.4.1 Profile character sets

Some document profile attributes have values consisting of character strings, for example, the document management attributes. The character sets used in these character strings are specified by the document profile attribute "profile character sets".

This attribute "profile character sets" specifies a code extension announcer and designations of character sets, which are subject to the following restrictions:

– The code extension announcer shall be 04/03 when specified. This code extension announcer specifies the use of G0 and G1 sets in an 8-bit environment and also the invocation of G0 and G1 sets into GL and GR respectively. Thus, in each attribute to which this attribute applies, invocation shift functions are not necessary, because G0 and G1 sets are implicitly invoked by this code extension announcer.

– G0 set – Only ISO-IR 6 (the IRV of ISO/IEC 646), ISO-IR 2 (the primary set of ISO 6937), or any other version of ISO 646 may be designated for this set; these graphic character sets are implicitly invoked in GL.

– G1 set – No restrictions are placed on the graphic character sets that may be designated for this set; these graphic character sets are implicitly invoked in GR.

– The empty set shall be designated into G1 and invoked into GR if no other specific character set is invoked in GR.

If the attribute "profile character sets" is not specified, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed.

6.7.4.2 Comments character sets

The character sets assumed to have been designated and optionally invoked at the beginning of the character strings specified by the attributes "user readable comments" (see 6.6.4) and "user visible name" (see 6.6.5) are specified using the document profile attribute "comments character sets".

It also specifies code extension techniques and the graphic character sets which may be used in the attributes "user readable comments" and "user visible name".

If this attribute is specified, the code extension techniques which may be used in the "user readable comments" and "user visible name" shall be announced by appropriate code extension announcers. The use of G0 set and GL shall always be announced. Other code extension announcers are to be specified according to the requirements of a particular document.

Two kinds of code extension techniques are permitted for this attribute. One is to use GL and GR without shift functions, and the other is to use various character sets by shift functions. The former is rather restricted but no shift functions are necessary in the "user readable comments" and "user visible name". The same restriction as in 6.7.4.1 is applied in this case. The latter permits various usages of character sets but invocations shall be specified by shift functions in the "user readable comments" and "user visible name". The same restriction as in 6.5.4 is applied in this case.

All the graphic character sets which may be used in the attributes "user readable comments" and "user visible name" shall be designated in the "comments character sets".

There are no restrictions concerning the number of graphic character sets which are designated and/or invoked in the "comments character sets"; hence designation to the same G set overrides the previous G set.

If the attribute "comments character sets" is not specified, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed.

6.7.4.3 Alternative representation character sets

This attribute specifies the graphic character sets designated and invoked at the beginning of the attribute "alternative representation" other than the standard default graphic character sets.

The restriction on profile character sets described in 6.7.4.1 is also applied. If this attribute is not explicitly specified in the document profile, the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed.

6.7.5 Document management attributes

Document management attributes contain information about the content of the document and its purpose. Information relating to the following may be specified:

- document description (see the Note below);
- dates and times;
- originators;
- other user information;
- external references;
- local file references;
- content attributes;
- security information.

The attributes that may be used to specify this information are defined in CCITT Rec. T.414 | ISO 8613-4.

The string of characters used in the document management attributes shall belong to the character sets indicated in the document profile attribute "profile character sets" (see 6.7.4.1). If the latter attribute is not explicitly specified in the document profile, then the default defined in CCITT Rec. T.410-Series | ISO 8613 is assumed.

The control functions SP, CR and LF may also be used within the character strings but no other control functions are allowed. Hence the graphic character set cannot be changed in the document management attributes.

NOTE – The document description includes the specification of the document reference (see 6.7.2.6).

7 Specification of constituent constraints

This clause specifies the definitions of the constituent constraints which may be represented by data streams conforming to this Profile.

7.1 Introduction

The structure diagrams illustrating the relationships between the constituents in the logical structures are shown in 7.1.1. The macros indicated on these diagrams are defined in 7.3.1. These macros define the permissible values for the attribute "generator for subordinates" that are applicable to the constituents, and define the allowed structures that are supported by this Profile.

The structure diagrams illustrating the layout structures are shown in 7.1.2. The macros indicated on these diagrams are defined in 7.4.1.

7.1.1 Diagrams of relationships of logical constituents

See Figures 6 and 7.

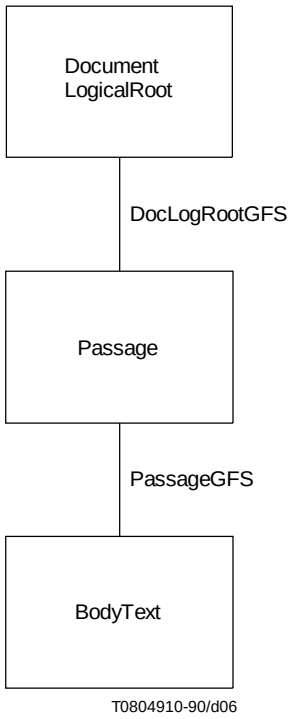

FIGURE 6/T.502

The body part of the generic logical structure

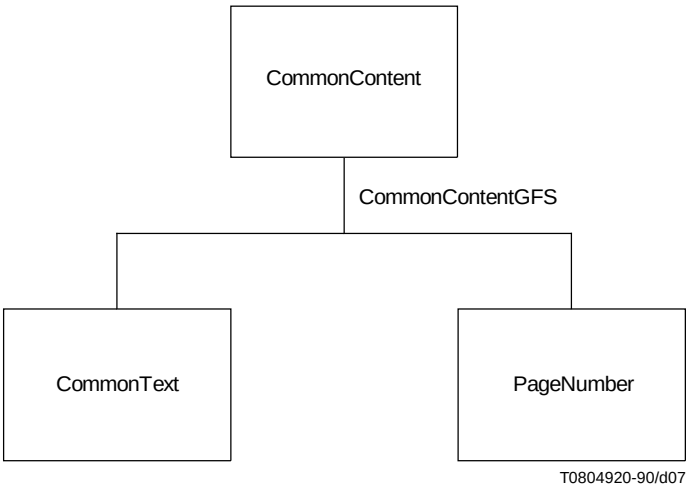

FIGURE 7/T.502

The common part of the generic logical structure

7.1.2 Diagrams of relationships of layout constituents

See Figures 8 and 9.

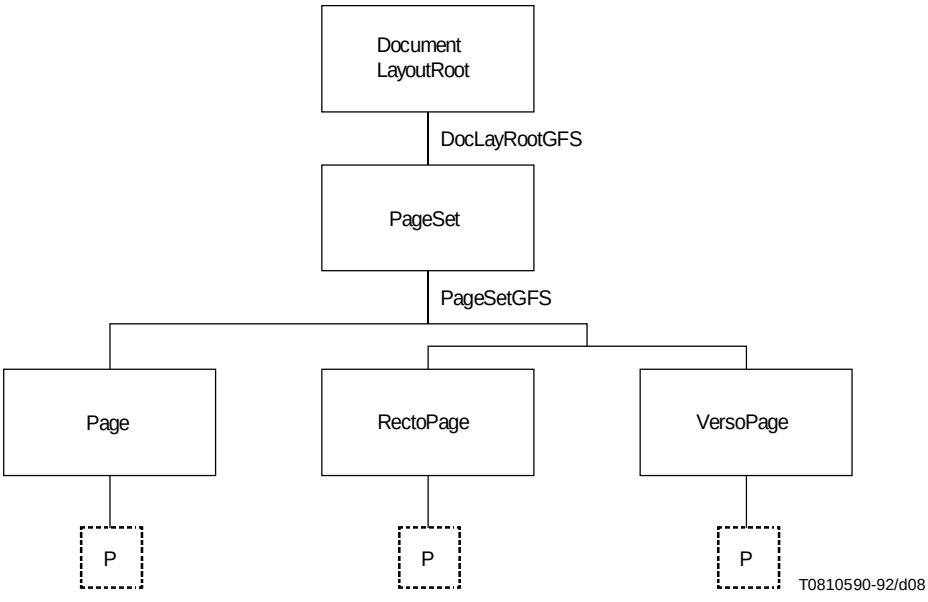

FIGURE 8/T.502

The layout structure – Document root and page sets

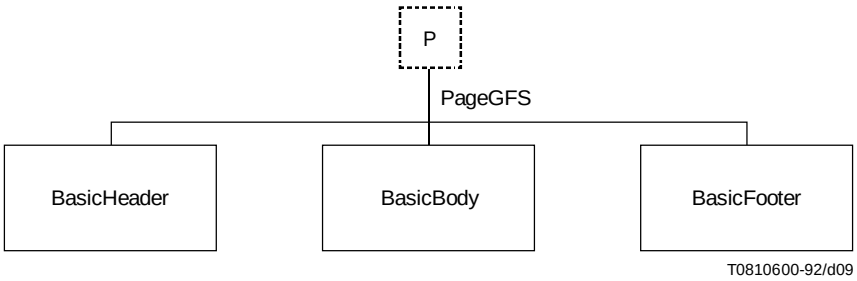

FIGURE 9/T.502

The layout structure – Page structure

7.1.3 Notation

This clause is written in accordance with the Document Application Profile Proforma and Notation (DAPPN) of CCITT Rec. T.411 | ISO 8613-1, Annex F. The following clarifications and minor extensions apply :

a) [Clarification]

The value range definition for the attributes “subordinates” and “imaging order” specify the set of object instances that may occur. The ordering and number (which may be zero) of object instances for the attribute “subordinates” must be in accordance with the value of the attribute “generator for subordinates” in the respective object class.

b) [Clarification]

The value “ANY_STRING” may include code extension control functions as well as graphic characters.

c) [Extension]

In order to write the specification of the usage of character sets and code extension control functions precisely, the following extensions are applied:

1) Table 3 defines the symbols that are introduced to denote shift functions.

2) <escape-sequence> is extended to include shift functions:

<escape-sequence> ::= 'ESC' <octet>... [<invocation-control-function>] ;

<invocation-control-function> ::= 'LS0'|'LS1R'|'LS2R'|'LS3R'|'SS2'|'SS3' ;

3) Data type specification for #ESC in content information is extended as:

<escape-sequence>...

TABLE 3/T.502

Symbols to denote shift functions

Symbol	Shift function	Coded representation
LS0	Locking shift zero	00/15
LS1R	Locking shift one right	ESC 07/14
LS2R	Locking shift two right	ESC 07/13
LS3R	Locking shift three right	ESC 07/12
SS2	Single shift two	08/14
SS3	Single shift three	08/15

7.2 Document profile constituent constraints

7.2.1 Macro definitions

```
DEFINE(FC,          "ASN.1{ 2 8 2 6 0 }"  -- formatted character content --)
DEFINE(PC,          "ASN.1{ 2 8 2 6 1 }"  -- processable character content --)
DEFINE(FPC,         "ASN.1{ 2 8 2 6 2 }"  -- formatted processable character content --)

DEFINE(FDA,             "{'formatted'}")
DEFINE(PDA,             "{'processable'}")
DEFINE(FPDA,            "{'formatted-processable'}")
DEFINE(PDA-FPDA,        "{'processable' | 'formatted-processable'}")

DEFINE(DAC,             "DocumentProfile (Document-architecture-class)")

DEFINE(GLAS,            "DocumentProfile (Generic-layout-structure)")

DEFINE(COMPLETE,    "{'complete-generator-set'}")

DEFINE(BasicPageDimensions, "
                REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=9240}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {<=12400}}
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=12400}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {<=9240}} ")
```

-- Any size equal to or smaller than CARA (Common Assured Reproduction Area) of ISO A4 and ANSI-A. Both
-- Landscape and Portrait may be specified.
-- Note that the above macro is defined for clarification of the specification and is not used in any other part of this DAP
-- specification.

```
DEFINE(NonBasicPageDimensions, "
                REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=14030}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {12401..19840}}
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {9241..14030}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {<=19840}}
                                -- up to ISO A3 portrait --

                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {12401..19840}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {<=14030}}
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=19840}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {9241..14030}}
                                -- up to ISO A3 landscape --

                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=13200}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {12401..20400}}
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {9241..13200}},
                 REQ #vertical-dimension
                    {REQ #fixed-dimension {<=20400}}
                                -- up to ANSI-B portrait --
```

```
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {12401..20400}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {<=13200}}
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=20400}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {9241..13200}} ")
                                    -- up to ANSI-B landscape --

DEFINE(PermissiblePageDimensions, "
                  REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=14030}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {<=19840}}        -- up to ISO A3 portrait --
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=19840}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {<=14030}}        -- up to ISO A3 landscape --
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=13200}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {<=20400}}        -- up to ANSI-B portrait --
                | REQ #horizontal-dimension
                    {REQ #fixed-dimension {<=20400}},
                  REQ #vertical-dimension
                    {REQ #fixed-dimension {<=13200}}        -- up to ANSI-B landscape --        ")

DEFINE(NominalPageSizes,
            "REQ #horizontal-dimension      {14030},
             REQ #vertical-dimension        {19840}              -- ISO A3 portrait --
            | REQ #horizontal-dimension     {19840},
             REQ #vertical-dimension        {14030}              -- ISO A3 landscape --
            | REQ #horizontal-dimension     {9920},
             REQ #vertical-dimension        {14030}              -- ISO A4 portrait --
            | REQ #horizontal-dimension     {14030},
             REQ #vertical-dimension        {9920}               -- ISO A4 landscape --
            | REQ #horizontal-dimension     {7015},
             REQ #vertical-dimension        {9920}               -- ISO A5 portrait --
            | REQ #horizontal-dimension     {9920},
             REQ #vertical-dimension        {7015}               -- ISO A5 landscape --
            | REQ #horizontal-dimension     {12141},
             REQ #vertical-dimension        {17196}              -- JIS B4 (Japanese legal) portrait --
            | REQ #horizontal-dimension     {17196},
             REQ #vertical-dimension        {12141}              -- JIS B4 (Japanese legal) landscape --
            | REQ #horizontal-dimension     {8598},
             REQ #vertical-dimension        {12141}              -- JIS B5 (Japanese letter) portrait --
            | REQ #horizontal-dimension     {12141},
             REQ #vertical-dimension        {8598}               -- JIS B5 (Japanese letter) landscape --
            | REQ #horizontal-dimension     {10200},
             REQ #vertical-dimension        {16800}              -- ANSI legal portrait --
            | REQ #horizontal-dimension     {16800},
             REQ #vertical-dimension        {10200}              -- ANSI legal landscape --
            | REQ #horizontal-dimension     {10200},
             REQ #vertical-dimension        {13200}              -- ANSI-A portrait --
            | REQ #horizontal-dimension     {13200},
             REQ #vertical-dimension        {10200}              -- ANSI-A landscape --
            | REQ #horizontal-dimension     {13200},
             REQ #vertical-dimension        {20400}              -- ANSI-B portrait --
            | REQ #horizontal-dimension     {20400},
             REQ #vertical-dimension        {13200}              -- ANSI-B landscape -- ")
```

```
DEFINE(GRAPHICRENDITIONS, "
        {'cancel' | 'increased-intensity' | 'italicized' | 'underlined' | 'crossed-out'
         | 'normal-intensity' | 'not-italicized' | 'not-underlined' | 'not-crossed-out'}... ")
```

-- Macro defining permissible code extension announcer. Note that all the values are basic. --

```
DEFINE(CDEXTEN,     " ESC 02/00 05/00,          -- Use G0 & LS0 --
                    [ ESC 02/00 05/03 ],        -- Use G1 & LS1R --
                    [ ESC 02/00 05/05 ],        -- Use G2 & LS2R --
                    [ ESC 02/00 05/07 ],        -- Use G3 & LS3R --
                    [ ESC 02/00 05/10 ],        -- Use G2 & SS2 --
                    [ ESC 02/00 05/11 ]         -- Use G3 & SS3 --      ")
```

-- Macro defining code extension announcer for DAP defaults --

```
DEFINE(DAP-DEFAULT-CDEXTEN,     "$CDEXTEN")
```

-- Macros defining final character for designation --

```
DEFINE(FCORE,       "04/02"     -- A final character designating ISO-IR 6 (the IRV of
                                -- ISO/IEC 646, i.e. ASCII) --)

DEFINE(F646,        "-- A final character designating any version of ISO/IEC 646, except ISO-IR 6 --")

DEFINE(F94S,        "-- A final character designating any registered 94 single byte graphic
                     -- character set, optionally preceded by one or more intermediate
                     -- characters as defined in Annex C of ISO 2022 --")

DEFINE(F94M,        "-- A final character designating any registered 94 multi byte graphic
                     -- character set, optionally preceded by one or more intermediate
                     -- characters as defined in Annex C of ISO 2022 --")

DEFINE(F96S,        "-- A final character designating any registered 96 single byte graphic
                     -- character set, optionally preceded by one or more intermediate
                     -- characters as defined in Annex C of ISO 2022 --")

DEFINE(F96M,        "-- A final character designating any registered 96 multi byte graphic
                     -- character set, optionally preceded by one or more intermediate
                     -- characters as defined in Annex C of ISO 2022 --")

DEFINE(FEMPTY,      "07/14"     -- The empty set --)
```

-- Macro defining a revision number of a character set --

```
DEFINE(REV,         "-- An octet between 04/00 and 07/14, which represents a revision number as
                     -- defined in ISO 2022. --")
```

-- Macros defining designation sequences --

```
DEFINE(DEG-CORE-G0, "ESC 02/08 $FCORE")
```

-- Designate 94 characters of ISO-IR 6 (the IRV of ISO/IEC 646) to G0 --

```
DEFINE(DEG-646-G0,  "ESC 02/08 $F646")
```

-- Designate any version of ISO/IEC 646, except ISO-IR 6, to G0 --

```
DEFINE(DEG-ANY-G1, "{[ESC 02/06 $REV]
                    {ESC 02/09 $F94S
                    | ESC 02/04 02/09 $F94M
                    | ESC 02/13 $F96S
                    | ESC 02/04 02/13 $F96M}}")
```

-- Designate any character set to G1 --

```
DEFINE(DEG-ANY-G2, "{[ESC 02/06 $REV]
                    {ESC 02/10 $F94S
                    | ESC 02/04 02/10 $F94M
                    | ESC 02/14 $F96S
                    | ESC 02/04 02/14 $F96M}}")
                -- Designate any character set to G2 --

DEFINE(DEG-ANY-G3, "{[ESC 02/06 $REV]
                    {ESC 02/11 $F94S
                    | ESC 02/04 02/11 $F94M
                    | ESC 02/15 $F96S
                    | ESC 02/04 02/15 $F96M}}")
                -- Designate any character set to G3 --

DEFINE(DEG-EMPTY-G1, "ESC 02/09 $FEMPTY")
                -- Designate the empty set to G1 --

-- Macro defining permissible graphic character sets --

DEFINE(PERMIT-GRCHAR,       "{$DEG-CORE-G0  LS0  |  $DEG-646-G0 LS0},
                            {{$DEG-ANY-G1 LS1R
                            |  $DEG-ANY-G2 LS2R
                            |  $DEG-ANY-G3 LS3R}...
                            |  $DEG-EMPTY-G1 LS1R}")
-- Macro defining graphic character sets for  DAP defaults --

DEFINE(DAP-DEFAULT-GRCHAR,          "$PERMIT-GRCHAR")

-- Macro defining basic graphic character sets. Note that this macro is defined for clarification of the specification and
-- is not used in any other part of this DAP specification. --

DEFINE(BASIC-GRCHAR,        "$DEG-CORE-G0 LS0,
                            $DEG-EMPTY-G1 LS1R")
-- Macro defining non-basic graphic character sets --

DEFINE(NON-BASIC-GRCHAR, "{$DEG-646-G0
                    |  $DEG-ANY-G1
                    |  $DEG-ANY-G2
                    |  $DEG-ANY-G3}... ")
-- Macro defining character sets used in document profile attributes --

DEFINE(PROFCHAR, "
        ESC 02/00 04/03                         -- announcement of use of G0 and G1, and
                                                -- invocation into GL and GR respectively (no shift
                                                -- functions are necessary) --
        { $DEG-CORE-G0 | $DEG-646-G0 }          -- designate G0 --
        { $DEG-ANY-G1 | $DEG-EMPTY-G1 }         -- designate G1 --
")

-- Macro defining comments character sets --

DEFINE(COMCHAR, "
        -- in the case to use both GL and GR without shift functions --
        ESC 02/00 04/03                         -- announcement of use of G0 and G1, and
                                                -- invocation into GL and GR respectively (no shift
                                                -- functions are necessary) --
        { $DEG-CORE-G0 | $DEG-646-G0 }          -- designate G0 --
        { $DEG-ANY-G1 | $DEG-EMPTY-G1 }         -- designate G1 --

        |       -- in the case of use of various character sets (shift functions are necessary) --
        {ESC 02/00 05/00,                       -- announcement to use G0 and LS0 --
        [ESC 02/00 05/03],                      -- announcement to use G1 and LS1R --
        [ESC 02/00 05/05],                      -- announcement to use G2 and LS2R --
        [ESC 02/00 05/07],                      -- announcement to use G3 and LS3R --
        [ESC 02/00 05/10],                      -- announcement to use G2 and SS2 --
        [ESC 02/00 05/11]}                      -- announcement to use G3 and SS3 --
```

```
        { $DEG-CORE-G0 | $DEG-646-G0 }          -- designate G0 --

        {{$DEG-ANY-G1                           -- designate G1 --
        | $DEG-ANY-G2                           -- designate G2 --
        | $DEG-ANY-G3}...                       -- designate G3 --
        | $DEG-EMPTY-G1}
")

-- Macro defining character sets used for alternative representation --

DEFINE(ALTCHAR,     "$PROFCHAR")
```

7.2.2 Constituent constraints

7.2.2.1 DocumentProfile

```
{
CASE $DAC OF  {

    $FDA:   PERM    Generic-layout-structure        {'factor-set'},
            REQ     Specific-layout-structure       {'present'},
            PERM    Presentation-styles             {'present'}

    $PDA:   PERM    Generic-layout-structure        {'complete-generator-set'},
            REQ     Generic-logical-structure       {'complete-generator-set'
                                                    | 'partial-generator-set'},
            REQ     Specific-logical-structure      {'present'},
            PERM    Presentation-styles             {'present'},
            PERM    Layout-styles                   {'present'}

    $FPDA:  REQ     Generic-layout-structure        {'complete-generator-set'},
            REQ     Specific-layout-structure       {'present'},
            REQ     Generic-logical-structure       {'complete-generator-set'
                                                    | 'partial-generator-set'},
            REQ     Specific-logical-structure      {'present'},
            PERM    Presentation-styles             {'present'},
            REQ     Layout-styles                   {'present'}
},

-- Document characteristics --

REQ     Document-application-profile        {-- See clause 8 for a definition of the
                                             -- permitted values for this attribute --},

PERM    Document-application-profile-defaults       {
    CASE  $DAC OF {
        $FDA:       {PERM   #content-architecture-class     {$FC | $FPC}}
        $PDA:       {PERM   #content-architecture-class     {$FC | $PC | $FPC}}
        $FPDA:      {PERM   #content-architecture-class     {$FC | $FPC}}
    },

    PERM    #dimensions         {$PermissiblePageDimensions},

    PERM    #medium-type        {PERM #nominal-page-size    {$NominalPageSizes},
                                 PERM #side-of-sheet        {ANY_VALUE}},

    PERM    #page-position      {ANY_VALUE},

    PERM    #character-content-defaults {
        PERM    #alignment                      {ANY_VALUE},
        PERM    #character-path                 {'0-degrees' | '180-degrees'},
        PERM    #character-spacing              {80 | 100 | 120 | 160 | 200},
        PERM    #code-extension-announcers      {$DAP-DEFAULT-CDEXTEN},
        PERM    #first-line-offset              {ANY_VALUE},
        PERM    #graphic-character-sets         {$DAP-DEFAULT-GRCHAR},
```

```
            PERM    #graphic-character-subrepertoire        {ANY_VALUE},
            PERM    #graphic-rendition                      {$GRAPHICRENDITIONS},
            PERM    #itemization                            {ANY_VALUE},
            PERM    #line-layout-table                      {ANY_VALUE},
            PERM    #line-progression                       {'90-degrees' | '270-degrees'},
            PERM    #line-spacing                           {100 | 150 | 200 | 300 | 400},
            PERM    #initial-offset                         {ANY_VALUE},
            PERM    #indentation                            {ANY_VALUE},
            PERM    #orphan-size                            {ANY_VALUE},
            PERM    #widow-size                             {ANY_VALUE}
            }

},

REQ     Document-architecture-class     {$FDA | $PDA  | $FPDA},
REQ     Content-architecture-classes    {[$FC], [$PC], [$FPC]},
REQ     Interchange-format-class        {-- See clause 8 for a definition of the
                                         -- permitted values for this attribute  --},
REQ     Oda-version     {REQ #standard-or-recommendation
                                {"CCITT Rec. T.410-Series(1988)|ISO 8613(1989); version 1.1"},
                        REQ #publication-date       {"1992-01-01"}},

-- Non-basic document characteristics --

PERM    Profile-character-sets                          {$PROFCHAR},
PERM    Comments-character-sets                         {$COMCHAR},
PERM    Alternative-representation-character-sets       {$ALTCHAR},
PERM    Page-dimensions                                 {PMUL {$NonBasicPageDimensions}},
PERM    Medium-types                                    {PMUL
                                                            {PERM #nominal-page-size
                                                                {$NominalPageSizes},
                                                        -- All permissible page sizes are non-basic. --
                                                            PERM #side-of-sheet         {'recto'|'verso'}}
},

PERM    Presentation-features               {
    PERM    #character-presentation-features {
            PERM    #character-path                         {'180-degrees'},
            PERM    #line-progression                       {'90-degrees'},
            PMUL {PERM      #graphic-character-sets         {$NON-BASIC-GRCHAR}},
            PMUL {PERM      #graphic-character-subrepertoire {ANY_VALUE}},
            PMUL {PERM      #character-spacing              {80 | 100 | 160 | 200}},
            PMUL {PERM      #line-spacing                   {100 | 150}},
            PMUL {PERM      #graphic-rendition              {'crossed-out' | 'not-crossed-out'}}
            }
},

-- Document management attributes --

-- Document description --

PERM    Title                           {ANY_STRING},
PERM    Subject                         {ANY_STRING},
PERM    Document-type                   {ANY_STRING},
PERM    Abstract                        {ANY_STRING},
PERM    Keywords                        {ANY_STRING...},
REQ     Document-reference              {ANY_VALUE},

-- Dates and times --
```

PERM Document-date-and-time {ANY_STRING},
PERM Creation-date-and-time {ANY_STRING},
PERM Local-filing-date-and-time {ANY_VALUE},
PERM Expiry-date-and-time {ANY_STRING},
PERM Start-date-and-time {ANY_STRING},
PERM Purge-date-and-time {ANY_STRING},
PERM Release-date-and-time {ANY_STRING},
PERM Revision-history {ANY_VALUE},

-- Originators --

PERM Organizations {ANY_STRING...},
PERM Preparers {ANY_VALUE},
PERM Owners {ANY_VALUE},
PERM Authors {ANY_VALUE},

-- Other user information --

PERM Copyright {ANY_VALUE},
PERM Status {ANY_STRING},
PERM User-specific-codes {ANY_STRING...},
PERM Distribution-list {ANY_VALUE},
PERM Additional-information {ANY_VALUE},

-- External references --

PERM References-to-other-documents {ANY_VALUE},
PERM Superseded-documents {ANY_VALUE},

-- Local file references --

PERM Local-file-references {ANY_VALUE},

-- Content attributes --

PERM Document-size {ANY_INTEGER},
PERM Number-of-pages {ANY_INTEGER},
PERM Languages {ANY_STRING...},

-- Security information --

PERM Authorization {ANY_VALUE},
PERM Security-classification {ANY_STRING},
PERM Access-rights {ANY_STRING...}
}

7.3 Logical constituent constraints

7.3.1 Macro definitions

DEFINE(DocLogRootGFS, "
<construction-expr> ::= REP OBJECT_CLASS_ID_OF(Passage);")

DEFINE(PassageGFS, "
<construction-expr> ::= REP OBJECT_CLASS_ID_OF(BodyText);")

DEFINE(CommonContentGFS, "
<construction-expr> ::= <construction-factor> | SEQ(<construction-factor>...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(PageNumber)
| OBJECT_CLASS_ID_OF(CommonText);")

```
DEFINE(PGNUMBER, "
        <string-expr> ::=    [ANY_STRING] <str-exp> [ANY_STRING];

        <str-exp>      ::=   MAKE-STRING(<num-exp>)
                           | UPPER-ALPHA(<num-exp>)
                           | LOWER-ALPHA(<num-exp>)
                           | UPPER-ROMAN(<num-exp>)
                           | LOWER-ROMAN(<num-exp>);

        <num-exp>     ::= B_REF(SUP(CURR-INST('frame', (CURR-OBJ))))(""PGnum"");")

DEFINE(DocumentLogicalRoot,    "REQ   #constraint-name      {""0""},
                               PERM   #external-data        {ANY_VALUE}")

DEFINE(Passage,                "REQ   #constraint-name      {""1""},
                               PERM   #external-data        {ANY_VALUE}")

DEFINE(BodyText,               "REQ   #constraint-name      {""14""},
                               PERM   #external-data        {ANY_VALUE}")

DEFINE(CommonContent,          "REQ   #constraint-name      {""19""},
                               PERM   #external-data        {ANY_VALUE}")

DEFINE(CommonText,             "REQ   #constraint-name      {""20""},
                               PERM   #external-data        {ANY_VALUE}")

DEFINE(PageNumber,             "REQ   #constraint-name      {""40""},
                               PERM   #external-data        {ANY_VALUE}")
```

7.3.2 Factor constraints

7.3.2.1 FACTOR ANY-LOGICAL

```
{
GENERIC:
     REQ    Object-type                  {VIRTUAL},
     REQ    Object-class-identifier      {ANY_VALUE}

SPECIFIC:
     PERM   Object-type                  {VIRTUAL},
     REQ    Object-identifier            {ANY_VALUE},
     REQ    Object-class                 {VIRTUAL}

SPECIFIC_AND_GENERIC:
     PERM   User-readable-comments       {ANY_STRING},
     PERM   User-visible-name            {ANY_STRING}
}
```

7.3.3 Constituent constraints

7.3.3.1 DocumentLogicalRoot

```
: ANY-LOGICAL {
GENERIC:
     REQ    Object-type                  {'document-logical-root'},
     REQ    Generator-for-subordinates   {$DocLogRootGFS},
     REQ    Application-comments         {$DocumentLogicalRoot}

SPECIFIC:
     PERM   Object-type                  {'document-logical-root'},
     REQ    Object-class                 {OBJECT_CLASS_ID_OF
                                            (DocumentLogicalRoot)},
     REQ    Subordinates                 {SUB_ID_OF(Passage)+},
     PERM   Application-comments         {$DocumentLogicalRoot}
}
```

7.3.3.2 Passage

```
: ANY-LOGICAL {
GENERIC:
    REQ     Object-type                     {'composite-logical-object'},
    REQ     Generator-for-subordinates      {$PassageGFS},
    REQ     Application-comments            {$Passage}

SPECIFIC:
    PERM    Object-type                     {'composite-logical-object'},
    REQ     Object-class                    {OBJECT_CLASS_ID_OF(Passage)},
    REQ     Subordinates                    {SUB_ID_OF(BodyText)+},
    CASE $GLAS OF {
        $COMPLETE:
            REQ     Layout-style            {STYLE_ID_OF(L-Style1)}
        VOID:
            PERM    Layout-style            {STYLE_ID_OF(L-Style1)}
    },
    PERM    Application-comments            {$Passage}
}
```

7.3.3.3 BodyText

```
: ANY-LOGICAL {
GENERIC:
    REQ     Object-type                     {'basic-logical-object'},
    REQ     Application-comments            {$BodyText}

SPECIFIC:
    PERM    Object-type                     {'basic-logical-object'},
    REQ     Object-class                    {OBJECT_CLASS_ID_OF(BodyText)},
    REQ     Content-portions                {CONTENT_ID_OF
                                                (Character-content-portion)+},
    PERM    Presentation-style              {STYLE_ID_OF(P-Style1)},
    PERM    Content-architecture-class      {$FC | $PC | $FPC},
    PERM    Layout-style                    {STYLE_ID_OF(L-Style2)},
    PERM    Application-comments            {$BodyText}
}
```

7.3.3.4 CommonContent

```
{
GENERIC:
    REQ     Object-type                     {'composite-logical-object'},
    REQ     Object-class-identifier         {ANY_VALUE},
    REQ     Generator-for-subordinates      {$CommonContentGFS},
    REQ     Application-comments            {$CommonContent},
    PERM    User-readable-comments          {ANY_STRING},
    PERM    User-visible-name               {ANY_STRING}
}
```

7.3.3.5 CommonText

```
{
GENERIC:
    REQ     Object-type                     {'basic-logical-object'},
    REQ     Object-class-identifier         {ANY_VALUE},
    REQ     Content-portions                {CONTENT_ID_OF
                                                (Character-content-portion)},
    PERM    Presentation-style              {STYLE_ID_OF(P-Style4)},
    PERM    Content-architecture-class      {$FC | $PC | $FPC},
    PERM    Layout-style                    {STYLE_ID_OF(L-Style3)},
    REQ     Application-comments            {$CommonText},
    PERM    User-readable-comments          {ANY_STRING},
    PERM    User-visible-name               {ANY_STRING}
}
```

7.3.3.6 PageNumber

```
{
GENERIC:
      REQ     Object-type                    {'basic-logical-object'},
      REQ     Object-class-identifier        {ANY_VALUE},
      REQ     Content-generator              {$PGNUMBER},
      PERM    Presentation-style             {STYLE_ID_OF(P-Style4)},
      PERM    Content-architecture-class     {$FC | $PC | $FPC},
      PERM    Layout-style                   {STYLE_ID_OF(L-Style3)},
      REQ     Application-comments           {$PageNumber},
      PERM    User-readable-comments         {ANY_STRING},
      PERM    User-visible-name              {ANY_STRING}
}
```

7.4 Layout constituent constraints

7.4.1 Macro definitions

```
DEFINE(DocLayRootGFS, "
      <construction-expr> ::=       REP  CHO({OBJECT_CLASS_ID_OF(PageSet)}...);
")

DEFINE(PageSetGFS, "<construction-expr> ::= <constraint-1>
                                          | <constraint-2>
                                          | <constraint-3>
                                          | <constraint-4>
                                          | <constraint-5>;

<constraint-1>      ::=     OBJECT_CLASS_ID_OF(Page);

<constraint-2>      ::=     REP  OBJECT_CLASS_ID_OF(Page);

<constraint-3>      ::=     REP  SEQ(OBJECT_CLASS_ID_OF(RectoPage)
                                     OPT  OBJECT_CLASS_ID_OF(VersoPage))
                            | REP  SEQ(OBJECT_CLASS_ID_OF(VersoPage)
                                     OPT  OBJECT_CLASS_ID_OF(RectoPage));

<constraint-4>      ::=     SEQ(OBJECT_CLASS_ID_OF(Page)
                                     OPT  REP  OBJECT_CLASS_ID_OF(Page));

<constraint-5>      ::=     SEQ(OBJECT_CLASS_ID_OF(Page)
                                     OPT REP  SEQ(OBJECT_CLASS_ID_OF(RectoPage)
                                          OPT OBJECT_CLASS_ID_OF(VersoPage)))
                            | SEQ(OBJECT_CLASS_ID_OF(Page)
                                     OPT REP  SEQ(OBJECT_CLASS_ID_OF(VersoPage)
                                          OPT OBJECT_CLASS_ID_OF(RectoPage)));
")

DEFINE(PageGFS, "
      <construction-expr> ::=       SEQ([OBJECT_CLASS_ID_OF (BasicHeader)]
                                        OBJECT_CLASS_ID_OF (BasicBody)
                                        [OBJECT_CLASS_ID_OF (BasicFooter)]);
")

DEFINE(INITIALISEPGNUM, "
            REQ     #binding-name           {""PGnum""},
            REQ     #binding-value          {>=–1}
")

DEFINE(PAGENUMBER, "
            REQ     #binding-name           {""PGnum""},
            REQ     #binding-value          {<string-expr> ::=
                                                INC(B_REF(PREC(CURR-OBJ)) (""PGnum""));}

")
```

```
DEFINE(DocumentLayoutRoot,    "REQ    #constraint-name        {""0""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(PageSet,               "REQ    #constraint-name        {""1""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(Page,                  "REQ    #constraint-name        {""2""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(RectoPage,             "REQ    #constraint-name        {""3""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(VersoPage,             "REQ    #constraint-name        {""4""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(BasicHeader,           "REQ    #constraint-name        {""27""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(BasicBody,             "REQ    #constraint-name        {""28""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(SpecificBlock,         "REQ    #constraint-name        {""30""},
                              PERM    #external-data          {ANY_VALUE}")

DEFINE(BasicFooter,           "REQ    #constraint-name        {""33""},
                              PERM    #external-data          {ANY_VALUE}")
```

7.4.2 Factor constraints

7.4.2.1 FACTOR ANY-COMPOSITE-LAYOUT

```
{
GENERIC:
    REQ     Object-type                     {VIRTUAL},
    REQ     Object-class-identifier         {ANY_VALUE}

SPECIFIC:
    PERM    Object-type                     {VIRTUAL},
    REQ     Object-identifier               {ANY_VALUE}

SPECIFIC_AND_GENERIC:
    PERM    User-readable-comments          {ANY_STRING},
    PERM    User-visible-name               {ANY_STRING}
}
```

7.4.2.2 FACTOR ANY-PAGE

```
: ANY-COMPOSITE-LAYOUT    {
GENERIC:
    REQ     Object-type                     {'page'},
    CASE $DAC OF {
        $PDA-FPDA:
            PERM    Bindings                        {$PAGENUMBER},
            REQ     Generator-for-subordinates      {$PageGFS}
        }

SPECIFIC:
    PERM    Object-type                     {'page'},
    REQ     Subordinates                    {SUB_ID_OF(BasicHeader),
                                             SUB_ID_OF(BasicBody),
                                             SUB_ID_OF(BasicFooter)}

SPECIFIC_AND_GENERIC:
        PERM    Dimensions                  {$PermissiblePageDimensions},
        PERM    Page-position               {ANY_VALUE}
}
```

7.4.2.3 FACTOR ANY-FRAME

```
: ANY-COMPOSITE-LAYOUT     {
GENERIC:
     REQ     Object-type                    {'frame'}

SPECIFIC:
     PERM    Object-type                    {'frame'},
     REQ     Subordinates                   {SUB_ID_OF(SpecificBlock)+}

SPECIFIC_AND_GENERIC:
     PERM    Position                       {REQ #fixed-position
                                                 {REQ #horizontal-position {ANY_VALUE},
                                                 REQ #vertical-position {ANY_VALUE}}},
     PERM    Dimensions                     {REQ #horizontal-dimension
                                                 {REQ #fixed-dimension {ANY_VALUE}},
                                             REQ #vertical-dimension
                                                 {REQ #fixed-dimension {ANY_VALUE}}}
}
```

7.4.3 Constituent constraints

7.4.3.1 DocumentLayoutRoot

```
: ANY-COMPOSITE-LAYOUT  {
GENERIC:
     REQ     Object-type                         {'document-layout-root'},
     CASE $DAC OF {
          $PDA-FPDA:
               PERM      Bindings                        {$INITIALISEPGNUM},
               REQ       Generator-for-subordinates      {$DocLayRootGFS}
     },
     REQ     Application-comments                {$DocumentLayoutRoot}

SPECIFIC:
     PERM    Object-type                         {'document-layout-root'},
     CASE $DAC OF {
          $FDA:    PERM   Object-class     {OBJECT_CLASS_ID_OF(DocumentLayoutRoot)}
          $FPDA:   REQ    Object-class     {OBJECT_CLASS_ID_OF(DocumentLayoutRoot)}
     },
     REQ     Subordinates                        {SUB_ID_OF (PageSet)+},
     PERM    Application-comments                {$DocumentLayoutRoot}
}
```

7.4.3.2 PageSet

```
: ANY-COMPOSITE-LAYOUT     {
GENERIC:
     REQ          Object-type                    {'page-set'},
     CASE $DAC OF {
          $PDA-FPDA:
               PERM     Bindings                        {$INITIALISEPGNUM},
               REQ      Generator-for-subordinates      {$PageSetGFS}
     },
     REQ     Application-comments                {$PageSet}

SPECIFIC:
     PERM    Object-type                         {'page-set'},
     CASE $DAC OF {
          $FDA:    PERM   Object-class     {OBJECT_CLASS_ID_OF (PageSet)}
          $FPDA:   REQ    Object-class     {OBJECT_CLASS_ID_OF (PageSet)}
     },
     REQ     Subordinates                        {SUB_ID_OF (Page)+,
                                                  SUB_ID_OF (RectoPage)+,
                                                  SUB_ID_OF (VersoPage)+ },
     PERM    Application-comments                {$PageSet}
}
```

7.4.3.3 Page

```
: ANY-PAGE  {
GENERIC:
      REQ      Application-comments                {$Page}

SPECIFIC:
      CASE $DAC OF {
           $FDA:    PERM   Object-class            {OBJECT_CLASS_ID_OF(Page)}
           $FPDA:   REQ    Object-class            {OBJECT_CLASS_ID_OF(Page)}
      },
      PERM     Application-comments                {$Page}

SPECIFIC_AND_GENERIC:
      PERM     Medium-type                         {PERM #nominal-page-size
                                                                  {$NominalPageSizes},
                                                   PERM #side-of-sheet        {ANY_VALUE}}
}
```

7.4.3.4 RectoPage

```
: ANY-PAGE  {
GENERIC:
      REQ      Application-comments                {$RectoPage},
      REQ      Medium-type                         {PERM #nominal-page-size
                                                                  {$NominalPageSizes},
                                                    REQ #side-of-sheet        {'recto' | 'unspecified'}}

SPECIFIC:
      CASE $DAC OF {
           $FDA:    PERM   Object-class            {OBJECT_CLASS_ID_OF (RectoPage)}
           $FPDA:   REQ    Object-class            {OBJECT_CLASS_ID_OF (RectoPage)}
      },

      PERM     Application-comments                {$RectoPage},
      PERM     Medium-type                         {PERM #nominal-page-size
                                                                  {$NominalPageSizes},
                                                   PERM #side-of-sheet        {'recto' | 'unspecified'}}
}
```

7.4.3.5 VersoPage

```
: ANY-PAGE  {
GENERIC:
      REQ      Application-comments                {$VersoPage},
      REQ      Medium-type                         {PERM #nominal-page-size
                                                                  {$NominalPageSizes},
                                                    REQ #side-of-sheet        {'verso' |'unspecified'}}

SPECIFIC:
      CASE $DAC OF {
           $FDA:    PERM   Object-class            {OBJECT_CLASS_ID_OF (VersoPage)}
           $FPDA:   REQ    Object-class            {OBJECT_CLASS_ID_OF (VersoPage)}
      },
      PERM     Application-comments                {$VersoPage},
      PERM     Medium-type                         {PERM #nominal-page-size
                                                                  {$NominalPageSizes},
                                                    PERM #side-of-sheet        {'verso' | 'unspecified'}}
}
```

7.4.3.6 BasicHeader

```
:  ANY-FRAME     {
GENERIC:
      CASE $DAC OF {
           $PDA-FPDA:
                REQ  Logical-source          {OBJECT_CLASS_ID_OF (CommonContent)}
      },
      REQ      Application-comments           {$BasicHeader}
```

```
SPECIFIC:
    CASE $DAC OF {
        $FDA:   PERM  Object-class     {OBJECT_CLASS_ID_OF (BasicHeader)}
        $FPDA:  REQ   Object-class     {OBJECT_CLASS_ID_OF (BasicHeader)}
    },
    PERM    Application-comments       {$BasicHeader}
}
```

7.4.3.7 BasicBody

```
: ANY-FRAME     {
GENERIC:
    REQ     Application-comments       {$BasicBody}

SPECIFIC:
    CASE $DAC OF {
        $FDA:   PERM  Object-class     {OBJECT_CLASS_ID_OF (BasicBody)}
        $FPDA:  REQ   Object-class     {OBJECT_CLASS_ID_OF (BasicBody)}
    },
    PERM    Application-comments       {$BasicBody}
}
```

7.4.3.8 BasicFooter

```
: ANY-FRAME     {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ  Logical-source        {OBJECT_CLASS_ID_OF (CommonContent)}
    },
    REQ     Application-comments       {$BasicFooter}

SPECIFIC:
    CASE $DAC OF {
        $FDA:   PERM  Object-class     {OBJECT_CLASS_ID_OF (BasicFooter)}
        $FPDA:  REQ   Object-class     {OBJECT_CLASS_ID_OF (BasicFooter)}
    },
    PERM    Application-comments       {$BasicFooter}
}
```

7.4.3.9 SpecificBlock

```
{
SPECIFIC:
    REQ     Object-type                {'block'},
    REQ     Object-identifier          {ANY_VALUE},
    REQ     Content-portions           {CONTENT_ID_OF(Character-content-portion)+},
    PERM    Presentation-style         {STYLE_ID_OF(P-Style1)
                                       | STYLE_ID_OF(P-Style4)},
    PERM    Content-architecture-class {$FC  | $FPC},

    PERM    Presentation-attributes    {
        PERM    #character-attributes  {
            PERM    #alignment                          {ANY_VALUE},
            PERM    #character-path                     {'0-degrees' | '180-degrees'},
            PERM    #character-spacing                  {80 | 100 | 120 | 160 | 200},
            PERM    #code-extension-announcers          {$CDEXTEN},
            PERM    #first-line-offset                  {ANY_VALUE},
            PERM    #graphic-character-sets             {$PERMIT-GRCHAR},
            PERM    #graphic-character-subrepertoire    {ANY_VALUE},
            PERM    #graphic-rendition                  {$GRAPHICRENDITIONS},
            PERM    #itemization                        {ANY_VALUE},
            PERM    #line-layout-table                  {ANY_VALUE},
            PERM    #line-progression                   {'90-degrees' | '270-degrees'},
            PERM    #line-spacing                       {100 | 150  | 200 | 300 | 400},
            PERM    #initial-offset                     {ANY_VALUE}
        }
    },
```

```
PERM    User-readable-comments      {ANY_STRING},
PERM    User-visible-name           {ANY_STRING},
PERM    Position                    {REQ #fixed-position
                                        {REQ #horizontal-position   {ANY_VALUE},
                                         REQ #vertical-position     {ANY_VALUE}}},
PERM    Dimensions                  {REQ #horizontal-dimension
                                        {REQ #fixed-dimension   {ANY_VALUE}},
                                    REQ #vertical-dimension
                                    {REQ #fixed-dimension   {ANY_VALUE}}},
PERM    Application-comments    {$SpecificBlock}                }
```

-- Each block in a BasicBody shall reference only one content portion. --
-- A block in a BasicHeader or BasicFooter may reference any number of content portions. --

7.5 Layout style constituent constraints

7.5.1 Macro definitions

-- No macro definitions are applicable in this clause. --

7.5.2 Factor constraints

7.5.2.1 FACTOR ANY-LAYOUT-STYLE

```
{
REQ     Layout-style-identifier     {ANY_VALUE},
PERM    User-readable-comments      {ANY_STRING},
PERM    User-visible-name           {ANY_STRING}
}
```

7.5.3 Constituent constraints

7.5.3.1 L-Style1

```
: ANY-LAYOUT-STYLE  {
```

-- This style is used for the constituent constraints Passage only. --

```
CASE $GLAS OF {
    $COMPLETE:
        PERM    Indivisibility          {'page' -- to layout object type --
                                        | 'null'},
        REQ     Layout-object-class     {OBJECT_CLASS_ID_OF (PageSet)}
    VOID:
        PERM    Indivisibility          {'page' -- to layout object type --
                                        | 'null'}
} }
```

7.5.3.2 L-Style2

```
: ANY-LAYOUT-STYLE  {
```

-- This style is used for the constituent constraints BodyText only. --

```
CASE $GLAS OF {
    $COMPLETE:
        PERM    Indivisibility          {'page' -- to layout object type --
                                        | 'null'},
        PERM    New-layout-object       {{OBJECT_CLASS_ID_OF (Page)
                                         | OBJECT_CLASS_ID_OF (RectoPage)
                                         | OBJECT_CLASS_ID_OF (VersoPage)
                                         | OBJECT_CLASS_ID_OF (BasicBody)
                                                -- to layout object class -- }
                                        | 'page'  -- to layout object type --
                                        | 'null'},
```

```
            PERM    Offset                  {ANY_VALUE},
            PERM    Same-layout-object      {REQ {REQ #logical-object
                                                  {<object-id-expr> ::= PREC-OBJ(CURR-OBJ);}
                                                  | REQ #logical-object  {'null'}},
                                            PERM #layout-object       {'page'}},
            PERM    Separation              {PERM  #leading-edge      {ANY_INTEGER},
                                             PERM  #trailing-edge     {ANY_INTEGER}}
        VOID:
            PERM    Indivisibility          {'page' -- to layout object type --
                                            | 'null'},
            PERM    New-layout-object       {'page' -- to layout object type--
                                            | 'null'},
            PERM    Offset                  {ANY_VALUE},
            PERM    Same-layout-object      {REQ {REQ #logical-object
                                                  {<object-id-expr> ::= PREC-OBJ(CURR-OBJ);}
                                                  | REQ #logical-object  {'null'}},
                                            PERM #layout-object       {'page'}},
            PERM    Separation              {PERM #leading-edge       {ANY_INTEGER},
                                             PERM #trailing-edge      {ANY_INTEGER}}
} }
```

7.5.3.3 L-Style3

```
: ANY-LAYOUT-STYLE  {

-- This style is used for the constituent constraints CommonText and PageNumber. --

PERM    Concatenation               {ANY_VALUE},
PERM    Offset                      {ANY_VALUE},
PERM    Separation                  {PERM #leading-edge       {ANY_INTEGER},
                                     PERM #trailing-edge      {ANY_INTEGER}}
}
```

7.6 Presentation style constituent constraints

7.6.1 Macro definitions

-- No macro definitions are applicable in this subclause. --

7.6.2 Factor constituent constraints

7.6.2.1 FACTOR ANY-PRESENTATION-STYLE

```
{
REQ     Presentation-style-identifier   {ANY_VALUE},
PERM    User-readable-comments          {ANY_STRING},
PERM    User-visible-name               {ANY_STRING}
}
```

7.6.3 Constituent constraints

7.6.3.1 P-Style1

```
: ANY-PRESENTATION-STYLE  {

-- This style is used for the constituent constraints BodyText and SpecificBlock.--

PERM    Presentation-attributes     {
    PERM    #character-attributes           {
        PERM    #alignment                      {ANY_VALUE},
        PERM    #character-path                 {'0-degrees' | '180-degrees'},
        PERM    #character-spacing              {80 | 100 | 120 | 160 | 200},
        PERM    #code-extension-announcers      {$CDEXTEN},
        PERM    #first-line-offset              {ANY_VALUE},
```

```
            PERM    #graphic-character-sets                 {$PERMIT-GRCHAR},
            PERM    #graphic-character-subrepertoire        {ANY_VALUE},
            PERM    #graphic-rendition                      {$GRAPHICRENDITIONS},
            PERM    #itemization                            {ANY_VALUE},
            PERM    #line-layout-table                      {ANY_VALUE},
            PERM    #line-progression                       {'90-degrees' | '270-degrees'},
            PERM    #line-spacing                           {100 | 150 | 200 | 300 | 400},
            PERM    #indentation                            {ANY_VALUE},
            PERM    #orphan-size                            {ANY_VALUE},
            PERM    #widow-size                             {ANY_VALUE}
    }
} }
```

7.6.3.2 P-Style2

-- This presentation style is not used.--

7.6.3.3 P-Style3

-- This presentation style is not used.--

7.6.3.4 P-Style4

```
: ANY-PRESENTATION-STYLE  {
```

-- This style is used for the constituent constraints CommonText, PageNumber and SpecificBlock.--

```
PERM    Presentation-attributes     {
    PERM    #character-attributes           {
            PERM    #alignment                              {ANY_VALUE},
            PERM    #character-path                         {'0-degrees' | '180-degrees'},
            PERM    #character-spacing                      {80 | 100 | 120 | 160 | 200},
            PERM    #code-extension-announcers              {$CDEXTEN},
            PERM    #first-line-offset                      {ANY_VALUE},
            PERM    #graphic-character-sets                 {$PERMIT-GRCHAR},
            PERM    #graphic-character-subrepertoire        {ANY_VALUE},
            PERM    #graphic-rendition                      {$GRAPHICRENDITIONS},
            PERM    #itemization                            {ANY_VALUE},
            PERM    #line-layout-table                      {ANY_VALUE},
            PERM    #line-progression                       {'90-degrees' | '270-degrees'},
            PERM    #line-spacing                           {100 | 150 | 200 | 300 | 400},
            PERM    #indentation                            {ANY_VALUE}
    }
} }
```

7.7 Content portion constituent constraints

7.7.1 Macro definitions

-- No macro definitions are applicable in this subclause.--

7.7.2 Factor constraints

-- No factor constraints are applicable in this subclause.--

7.7.3 Constituent constraints

7.7.3.1 Character-content-portion

```
{
CASE $DAC OF {
    $FDA :
        REQ     Content-identifier-layout           {ANY_VALUE}
```

$PDA:

REQ Content-identifier-logical {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic -- logical object or a basic logical object class. --

$FPDA:

REQ Content-identifier-layout {ANY_VALUE},

REQ Content-identifier-logical {ANY_VALUE}

-- Both attributes are specified, if the content portion is associated with a -- basic logical object and a basic layout object. --

| REQ Content-identifier-logical {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic -- logical object class.--

},

PERM Alternative-representation {ANY_STRING},

PERM Content-information {CHARACTER

{#STAB {ANY_VALUE}
| #SHS { 0 | 1 | 2 | 3 } *-- 120, 100, 80 or 200 BMU --*
| #SCS {80 | 100 | 120 | 160 | 200}
| #SGR {$GRAPHICRENDITIONS}
| #SVS { 0 | 1 | 2 | 3 | 4 } *-- 200, 300, 400, 100 or 150 BMU --*
| #SRS {ANY_VALUE}
| #JFY {0}
| #CR
| #LF
| #PLD
| #PLU
| #SP
| #SUB
| #BPH
| #NBH
| #SOS
| #ST
| #LS0
| #LS1R
| #LS2R
| #LS3R
| #SS2
| #SS3
| #ESC {$DEG-CORE-G0}
| #ESC {$DEG-646-G0}
| #ESC {$DEG-ANY-G1}
| #ESC {$DEG-ANY-G2}
| #ESC {$DEG-ANY-G3}
| #ESC {$DEG-EMPTY-G1}
} ... }

}

8 Interchange format

8.1 Interchange format

For conformance to this Profile, the ODIF interchange format class A shall be used. The value of the document profile attribute “interchange format” for this interchange format is 'if-a'. This form of ODIF is defined in CCITT Rec. T.415 | ISO 8613-5.

8.2 Document application profile object identifier

The value for the document profile attribute “document application profile” for this interchange format is represented by the following object identifier:

ASN.1 {2 8 4 0 11 0}

8.3 Encoding of application comments

The encoding of the attribute “application comments” is defined in this encoding as an octet string as specified in CCITT Rec. T.415 | ISO 8613-5. This document application profile requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the following module definition:

```
FOD-DAPSpecification
DEFINITIONS      ::= BEGIN
EXPORTS      Appl-Comm-Encoding;

Appl-Comm-Encoding              ::= SEQUENCE {
        constraint-name         [0] IMPLICIT PrintableString OPTIONAL,
        external-data           [1] IMPLICIT OCTET STRING OPTIONAL }

END
```

8.4 Data lengths

The maximum length of data values of the type OCTET STRING, as defined in CCITT Rec. X.208 | ISO/IEC 8824, in data streams which may be encoded in accordance with this DAP is 32 767 octets. If it is required to encode an octet string of greater length than this, constructed type encoding shall be used. That is, data values greater than 32 767 in length shall be split into a sequence of strings shorter than 32 767, each of which is encoded using a primitive type.

Annex A

Amendments and technical corrigenda

(This annex forms an integral part of this Recommendation)

A.1 Amendments

A.1.1 Amendments to the base standard

The amendments applicable to this Profile includes text to be included in CCITT Rec. T.411 | ISO 8613-1 as the following annexes:

- *Annex E* – Use of ISO/IEC 10021 (MOTIS) to interchange documents conforming to CCITT Rec. T.410-Series | ISO 8613 – Published as First extension to the CCITT Rec. T.410-Series (1988), Addendum II | ISO 8613-1 (1989): Amendment 1.
- *Annex F* – Document Application Profile proforma and notation – Published as CCITT Rec. T.411, Annex F (1991) | ISO 8613-1 (1989): Addendum 1.
- *Annex G* – Conformance testing methodology – Published as ISO 8613-1 (1989): Amendment 2.
- *Annex H* – Recording of documents conforming to CCITT Rec. T.410-Series | ISO 8613 on flexible disk cartridges conforming to ISO 9293 – Currently a Draft Amendment to ISO 8613-1 (1989), Annex H.

This Profile does not include the following features of the amendment:

- *Addendum on security* – Published as First extension to the CCITT Rec. T.410-Series (1988), Addendum V | ISO 8613 (1989): Addendum 4.
- *Addendum on styles* – Published as First extension to the CCITT Rec. T.410-Series (1988), Addendum IV | ISO 8613 (1989): Addendum 6.

– *Addendum on alternative representation* – Published as First extension to the CCITT Rec. T.410-Series (1988), Addendum III | ISO 8613 (1989): Addendum 3.

– *Addendum on colour* – Published as Revision of the CCITT Rec. T.410-Series contained in COM VIII – R30-E Addendum 2 | ISO 8613 (1989): Amendment 2.

– *Addendum on tiled raster graphics* – Published as First extension to the CCITT Rec. T.410-Series (1988), Addendum I | ISO 8613 (1989): Addendum 1.

A.1.2 Proposed changes to standards due to defects

This amendment addresses the inclusion of the CCITT Rec. T.410-Series (1988) | ISO 8613 (1989) Technical Corrigenda 1, 2 and 3.

A.2 Technical corrigenda

A.2.1 Technical corrigenda to this Recommendation

There is no corrigenda specific to this Recommendation.

A.3 Versions of the ODA International Standard

The version of the ODA International Standard defined in A.1 is known as version "1.1". The associated date is 1992-01-01. The version of the ODA International Standard including all features of the amendments, addenda and technical corrigenda mentioned in A.1 is known as "version 1.2". The associated date is 1992-07-01.

The entries for ODA version for this DAP is:

– Standard or Recommendation: "CCITT Rec. T.410-Series (1988) | ISO 8613 (1989); version 1.1".

– Publication date: "1992-01-01".

Annex B

Recommended practices

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

B.1 Transfer methods for ODA

B.1.1 Conveyance of ODA over CCITT X.400-1984

This describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
    document-application-profile
        [0] IMPLICIT OBJECT IDENTIFIER,
    document-architecture-class
        [1] IMPLICIT INTEGER {
            formatted (0),
            processable (1),
            formatted-processable (2) }
OdaData ::= SEQUENCE OF Interchange-Data-Element
```

NOTE – It is recommended to transfer an ODA document as a single body part with tag 12:

```
Oda [12] IMPLICIT OCTET STRING
```

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this Profile.

B.1.2 Conveyance of ODA over FTAM

This describes the FTAM Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type shall be specified.

However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This provides for information concerning the interchange of ODA based documents with DTAM protocols.

DTAM (Document Transfer and Manipulation) is defined in the T.430-Series CCITT Recommendations and is, like ODA, an integral part of the T.400-Series CCITT Recommendations named *Open Document Architecture, Transfer and Manipulation.*

The T.520-Series CCITT Recommendations contain *Communication Application Profiles (CAP).* CCITT Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. CCITT Recommendation T.522 is applicable for the Open Document Format Profile (FOD) published in this Recommendation.

NOTE – The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in CCITT Recommendation T.561, 7.1, and CCITT Recommendation T.562, 7.1.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges by the use of an annex to CCITT Rec. T.411 | ISO/IEC 8613-1 (to be published), *Recording of Documents Conforming to ISO 8613 on Flexible Disk Cartridges Conforming to ISO 9293.* This annex provides for recording each ODA document as a separate file as defined in ISO 9293.

Annex C

Bibliography

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

– CCITT Recommendation T.52 (1993), *Non-Latin coded character sets for telematic services*.

– CCITT Recommendation T.502 (1991), *Document Application Profile PM-11 for the Interchange of Character Content Documents in Processable and Formatted Forms.*

– CCITT Recommendation T.505 (1991), *Document Application Profile PM-26 for the Interchange of Mixed Content Documents in Processable and Formatted Forms.*

– ISO 8571:1988, *Information processing systems – Open Systems Interconnection – File Transfer, Access and Management.*

– ISO/IEC 9070:1991, *Information technology – SGML support facilities – Registration procedures for public text owner identifiers.*

– ISO/IEC TR 9573:1988, *Information processing – SGML support facilities – Techniques for using SGML.*

– ISO/IEC 10021:1990, *Information technology – Text Communication – Message-Oriented Text Interchange Systems (MOTIS).*