

INTERNATIONAL
STANDARD

ISO/IEC
8613-6

Second edition
1994-12-15

**Information technology — Open
Document Architecture (ODA) and
Interchange Format: Character content
architectures**

*Technologies de l'information — Architecture des documents ouverts
(ODA) et format d'échange: Architecture des contenus de caractères*



Reference number
ISO/IEC 8613-6:1994(E)
Documents
AVA Document Sharing Hub

Contents

	<i>Page</i>
1 Scope	1
2 Normative references	2
2.1 Identical Recommendations International Standards	2
2.2 Paired Recommendations International Standards equivalent in technical content	2
2.3 Additional references	2
3 Definitions	2
4 Abbreviations	3
5 Conventions	3
6 General principles	3
6.1 Content architecture classes	3
6.2 Content	4
6.3 Presentation attributes	4
6.4 Control functions	4
6.5 Graphic characters	5
6.6 Space characters	5
6.7 Coding of content information	5
6.8 Internal structure	5
6.8.1 Formatted content	5
6.8.2 Processable content	5
6.8.3 Formatted processable content	6
7 Character positioning	6
7.1 Basic concepts	6
7.1.1 Character fonts	6
7.1.2 Directions	7
7.1.3 Character image model	7
7.1.4 Character spacing	9
7.1.5 Active position	9
7.1.6 Positioning area	9
7.1.7 Line boxes	9
7.2 Positioning of character images within a line box	10
7.2.1 Spacing between characters	11
7.2.2 Alignment	12
7.2.3 Tabulation	12
7.2.4 Character ordering	12

© ISO/IEC 1994

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

7.2.5	Parallel annotation	14
7.2.6	Subscript/superscript.....	15
7.2.7	Pairwise kerning.....	15
7.2.8	First line offset	15
7.2.9	Itemization	15
7.3	Positioning of line boxes within a basic layout object.....	15
8	Character imaging	17
8.1	Emphasis.....	18
8.1.1	Weight.....	18
8.1.2	Posture	18
8.1.3	Underlining	18
8.1.4	Blinking	19
8.1.5	Image inversion.....	19
8.1.6	Crossing-out.....	19
8.1.7	Colour	19
8.2	Font selection	21
8.2.1	Font specification.....	21
8.2.2	Font designation.....	21
8.2.3	Font invocation	21
8.3	Subscript and superscript	21
8.4	Character combinations.....	22
9	Definition of character presentation attributes	22
9.1	Shared presentation attributes	23
9.1.1	Alignment	23
9.1.2	Character fonts	23
9.1.3	Character orientation.....	24
9.1.4	Character path	24
9.1.5	Character spacing.....	24
9.1.6	Code extension announcers.....	24
9.1.7	First line offset	25
9.1.8	Graphic character sets	25
9.1.9	Graphic character subrepertoire	25
9.1.10	Graphic rendition	26
9.1.11	Itemization	26
9.1.12	Kerning offset	27
9.1.13	Line layout table	27
9.1.14	Line progression.....	28
9.1.15	Line spacing.....	28
9.1.16	Pairwise kerning.....	28
9.2	Layout presentation attributes	28
9.2.1	Formatting indicator.....	28
9.2.2	Initial offset	29
9.3	Logical presentation attributes	29
9.3.1	Indentation	29
9.3.2	Orphan size	30
9.3.3	Proportional line spacing	30
9.3.4	Widow size.....	30
9.4	Content architecture class attributes.....	31
9.4.1	Content architecture class	31
9.5	Interactions between presentation attributes and layout directives	31

10	Character content portion attributes	31
10.1	Common coding attributes	31
10.2	Other coding attributes.....	31
10.3	Content information	31
11	Formal definitions of character content architecture dependent data types	32
11.1	Introduction.....	32
11.2	Representation of presentation attributes.....	32
11.3	Representation of coding attributes.....	35
11.4	Representation of non-basic features and non-standard defaults	35
12	Graphic Characters.....	36
13	Definition of control functions and the character SPACE	36
13.1	Shared control functions	37
13.1.1	Carriage Return (CR).....	37
13.1.2	Graphic Character Composition (GCC).....	38
13.1.3	Identify Graphic Subrepertoire (IGS)	38
13.1.4	Line Feed (LF)	38
13.1.5	Partial Line Down (PLD).....	39
13.1.6	Partial Line Up (PLU).....	39
13.1.7	Set Character Spacing (SCS)	39
13.1.8	Select Graphic Rendition (SGR).....	39
13.1.9	Select Character Spacing (SHS)	42
13.1.10	Set Line Spacing (SLS).....	42
13.1.11	Start Reverse String (SRS).....	42
13.1.12	Selective Tabulation (STAB).....	42
13.1.13	Substitute Character (SUB).....	42
13.1.14	Select Line Spacing (SVS).....	43
13.1.15	Line Position Backward (VPB).....	43
13.1.16	Line Position Relative (VPR)	43
13.1.17	Code extension control functions.....	43
13.2	Layout control functions	43
13.2.1	Backspace (BS).....	43
13.2.2	Character Position Backward (HPB)	44
13.2.3	Character Position Relative (HPR)	44
13.2.4	No Justify (JFY).....	44
13.2.5	Set Additional Character Separation (SACS)	44
13.2.6	Set Reduced Character Separation (SRCS).....	44
13.2.7	Set SPACE Width (SSW)	44
13.3	Logical control functions	45
13.3.1	Break Permitted Here (BPH)	45
13.3.2	No Break Here (NBH)	45
13.3.3	Parallel Texts (PTX)	45
13.4	Delimiters.....	45
13.4.1	Start Of Original String (SOOS).....	45
13.4.2	Start Of String (SOS)	45
13.4.3	String Terminator (ST)	45
13.5	SPACE (SP).....	45
14	Content layout process	46
14.1	Introduction.....	46
14.1.1	Purpose.....	46
14.1.2	Available area	46

14.1.3	Presentation attributes	46
14.1.4	Character content architecture classes	46
14.1.5	Use of delimiters	47
14.1.6	Layout of the content	47
14.1.7	Layout sequence	47
14.1.8	Character sets	47
14.2	Content layout process for processable content	48
14.2.1	Single basic logical object to single basic layout object	48
14.2.2	Single basic logical object to multiple basic layout objects	52
14.2.3	Multiple basic logical objects to single basic layout object	52
14.3	Content layout process for formatted processable content	53
14.4	Content layout process for formatted content	53
15	Content imaging process	53
15.1	Introduction	53
15.2	Content imaging process for formatted content	53
15.2.1	Determination of initial point	54
15.2.2	Choosing character images	54
15.2.3	Formatting indicator	54
15.3	Content imaging process for formatted processable content	54
16	Interactions between presentation attributes and control functions	54
17	Definition of character content architecture classes	55
17.1	Formatted character content architecture class	55
17.2	Processable character content architecture class	56
17.3	Formatted processable character content architecture class	56
Annex A	– SGML representation of character content-specific attributes for ODL	58
A.1	Introduction	58
A.2	Names and public identifiers	58
A.3	Representation of attribute values	58
A.3.1	Constructed parameters	58
A.3.2	Parentheses	59
A.3.3	Alternative representation	59
A.3.4	String parameters	59
A.3.5	Keyword parameters	59
A.3.6	Integer parameters	59
A.4	Presentation attributes	59
A.4.1	Shared presentation attributes (format attribute-directives)	59
A.4.2	Layout presentation attributes (format attributes)	60
A.4.3	Logical presentation attributes (format directives)	60
A.5	Coding attributes	60
Annex B	– Summary of content architecture classes	61
B.1	Formatted character content architecture class	61
B.2	Processable character content architecture class	63
B.3	Formatted processable character content architecture class	65
Annex C	– Coded representations of control functions	67
Annex D	– Summary of object identifiers	68

Foreword

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

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

International Standard ISO/IEC 8613-6 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 18, *Document processing and related communication*, in collaboration with ITU-T. The identical text is published as ITU-T Rec. T.416.

This second edition cancels and replaces the first edition (ISO 8613-6:1989), which has been technically revised.

ISO/IEC 8613 consists of the following parts, under the general title *Information technology — Open Document Architecture (ODA) and Interchange Format*:

- *Part 1: Introduction and general principles*
- *Part 2: Document structures*
- *Part 3: Abstract interface for the manipulation of ODA documents*
- *Part 4: Document profile*
- *Part 5: Open Document Interchange Format*
- *Part 6: Character content architectures*
- *Part 7: Raster graphics content architectures*
- *Part 8: Geometric graphics content architectures*
- *Part 9: Audio content architectures*
- *Part 10: Formal specifications*
- *Part 11: Tabular structures and tabular layout*
- *Part 12: Identification of document fragments*
- *Part 13: Spreadsheet*
- *Part 14: Temporal relationships and non-linear structures*

Annex A forms an integral part of this part of ISO/IEC 8613. Annexes B, C and D are for information only.

Introduction

This ITU-T Recommendation | International Standard was prepared as a joint publication by ITU-T Study Group VII and ISO/IEC Joint Technical Committee 1.

At present, the ITU-T Set of Recommendations in the T.410 series | International Standard ISO 8613 consists of:

- Introduction and general principles;
- Document structures;
- Document profile;
- Open document interchange formats;
- Character content architectures;
- Raster graphics content architectures;
- Geometric graphics content architectures;
- Formal specification of the Open Document Architecture (FODA)
(The formal specification is applicable to ISO/IEC 8613 only).

Further Recommendations | International Standards may be added to this series of ITU-T Recommendations | International Standards.

Development of this set of ITU-T Recommendations | International Standards was originally in parallel with ECMA-101 standard: *Open document architecture*.

This series of ITU-T Recommendations | International Standards is a new edition of the CCITT T.410-Series of Recommendations (1998) and ISO 8613 (1989).

Significant technical changes are the inclusion of the following amendments as agreed by ITU-T and ISO/IEC:

- Alternative Representation;
- Annex on use of MHS/MOTIS;
- Colour;
- Conformance Testing Annex;
- Document Application Profile, Proforma and Notation;
- Security;
- Streams;
- Styles;
- Tiled Raster Graphics.

In addition, a number of technical corrigenda have been applied to this ITU-T Recommendation | International Standard.

This ITU-T Recommendation | International Standard contains four annexes:

- Annex A (integral): SGML representation of character content-specific attributes for ODL;
- Annex B (non-integral): Summary of content architecture classes Summary of content architecture classes;
- Annex C (non-integral): Coded representations of control functions;
- Annex D (non-integral): Summary of object identifiers.

This page intentionally left blank

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN DOCUMENT ARCHITECTURE (ODA) AND INTERCHANGE FORMAT: CHARACTER CONTENT ARCHITECTURES

1 Scope

The purpose of the ITU-T Rec. T.410-Series | ISO/IEC 8613 is to facilitate the interchange of documents.

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

NOTE – These Recommendations | International Standards are designed to allow for extensions, including hypermedia features, spreadsheets and additional types of content such as audio and video.

In addition to the content types defined in these Recommendations | International Standards, ODA also provides for arbitrary content types to be included in documents.

These Recommendations | International Standards apply to the interchange of documents by means of data communications or the exchange of storage media.

These Recommendations | International Standards provide for the interchange of documents for either or both of the following purposes:

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

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

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

These Recommendations | International Standards also provide for the interchange of ODA information structures used for the processing of interchanged documents.

This ITU-T Recommendation | International Standard:

- defines a character content architecture that can be used in conjunction with the document architecture defined in ITU-T Rec. T.412 | ISO/IEC 8613-2;
- defines the internal structure of content conforming to this character content architecture;
- defines those aspects of rendition applicable to the presentation of character content;
- defines the presentation and content portion attributes applicable to this character content architecture;
- describes a character content layout process which, together with the document processing model described in ITU-T Rec. T.412 | ISO/IEC 8613-2, determines the layout of character content in basic layout objects.

2 Normative references

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

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation T.411 (1993) | ISO/IEC 8613-1:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Introduction and general principles*.
- ITU-T Recommendation T.412 (1993) | ISO/IEC 8613-2:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Document structures*.
- ITU-T Recommendation T.414 (1993) | ISO/IEC 8613-4:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Document profile*.
- ITU-T Recommendation T.415 (1993) | ISO/IEC 8613-5:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format*.
- ITU-T Recommendation T.417 (1993) | ISO/IEC 8613-7:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Raster graphics content architectures*.
- ITU-T Recommendation T.418 (1993) | ISO/IEC 8613-8:1994, *Information technology – Open Document Architecture (ODA) and Interchange Format: Geometric graphics content architectures*.

2.2 Paired Recommendations | International Standards equivalent in technical content

- 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)*.

2.3 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 6429:1992, *Information technology – Control functions for coded character sets*.
- ISO 6937-1:1983, *Information processing – Coded character sets for text communication – Part 1: General introduction*.
- ISO 6937-2:1983, *Information processing – Coded character sets for text communication – Part 2: Latin alphabetic and non-alphabetic graphic characters*.
- ISO/IEC 7350:1991, *Information technology – Registration of repertoires of graphic characters from ISO 10367*.
- ISO/IEC 8613-10:1991, *Information technology – Text and office systems – Open Document Architecture (ODA) and interchange format – Part 10: Formal specifications*.
- ISO 8879:1986, *Information processing – Text and office systems – Standard Generalized Markup Language (SGML)*.
- ISO/IEC 9541-1:1991, *Information technology – Font information interchange: Part 1 – Architecture*.

3 Definitions

For the purposes of this Recommendation | International Standard, the definitions given in ITU-T Rec. T.411 | ISO/IEC 8613-1 apply.

4 Abbreviations

BPH	Break Permitted Here
BS	Backspace
CR	Carriage Return
GCC	Graphic Character Composition
HPB	Character Position Backward
HPR	Character Position Relative
IGS	Identify Graphic Subrepertoire
JFY	No Justify
LF	Line Feed
NBH	No Break Here
PLD	Partial Line Up
PLU	Partial Line Down
PTX	Parallel Texts
SACS	Set Additional Character Separation
SCS	Set Character Spacing
SGR	Select Graphic Rendition
SHS	Select Character Spacing
SLS	Set Line Spacing
SOOS	Start Of Original String
SOS	Start Of String
SRCS	Set Reduced Character Separation
SRS	Start Reverse String
ST	String Terminator
SSW	Set Space Width
STAB	Selective Tabulation
SUB	Substitute Character
SVS	Select Line Spacing
VPB	Line Position Backward
VPR	Line Position Relative

5 Conventions

For the purpose of this Recommendation | International Standard, the conventions given in ITU-T Rec. T.411 | ISO/IEC 8613-1 apply.

6 General principles

6.1 Content architecture classes

This part of ITU-T Rec. T.410-Series | ISO/IEC 8613 defines three classes of character content architecture:

- A character content architecture for formatted content which allows for document content to be presented (e.g. printed or displayed) as intended by the originator. Formatted content can be used in any basic component.

- A character content architecture for processable content which allows for document content to be processed (e.g. edited or formatted). Processable content can be used in any basic logical component.
- A character content architecture for formatted processable content which allows for document content to be processed and also to be presented as intended by the originator. Formatted processable content can be used in any basic component.

6.2 Content

The content of a basic component that conforms to a character content architecture is a character string. This character string is formed by concatenating the character strings in the content portions of the basic component.

The content character string consists of a combination of graphic characters, control functions and space characters.

6.3 Presentation attributes

Presentation attributes are applicable to basic logical and layout components. They contain information that specifies the initial conditions relating to the layout, the imaging and the selection of graphic characters of the content of these basic components. Some of these conditions can be changed by control functions contained within the content.

Presentation attributes are classified as follows:

- Logical presentation attributes which can be associated with processable and formatted processable character content. These attributes take effect during the content layout process but are ignored during the content imaging process.
- Layout presentation attributes which can be associated with formatted and formatted processable character content. These attributes take effect during the content imaging process. They are generated either by a content layout process or by a process that creates or edits the formatted or formatted processable content.
- Shared presentation attributes which can be associated with all character content architecture classes.

These attributes take effect during either or both of the content layout and imaging processes.

All presentation attributes in this Specification are defaultable.

6.4 Control functions

Control functions with zero or more parameters may specify information relating to the layout or imaging of subsequent graphic characters. A control function can also be used to extend or replace the set of graphic characters being used. The scope of all control functions is limited to the basic component in which they occur and, in the case of basic logical components, any basic logical component concatenated to this component (see 14.2.3).

Classification of control functions is similar to that of presentation attributes:

- Logical control functions which can be used in processable and formatted processable character content.
These control functions take effect during the content layout process but are ignored during the content imaging process.
- Layout control functions which can be used in formatted and formatted processable character content.
These control functions take effect during the content imaging process. They are generated by the content layout process. Alternatively, they may be inserted by a process (not described in ITU-T Rec. T.410-Series I ISO/IEC 8613) that creates or edits the formatted or formatted processable content.
- Shared control functions which can be used in all character content architecture classes. These control functions take effect during either or both of the content layout and imaging processes.

In addition, formatted processable content may contain control functions known as delimiters. These delimiters are used to indicate a string of one or more graphic characters and/or control functions that have been inserted or deleted as the result of a content layout process (see clause 14). The inserted graphic characters and/or control functions take effect only during the content imaging process. The delimiters take effect during the content layout process by deleting them and the inserted character sequence.

6.5 Graphic characters

The set of graphic characters used in the content of a basic component, and their coded representation, are specified by presentation attributes and code extension control functions (see clause 12 and 13.1.17).

Any set or sets of graphic characters may be used in the content of basic components, subject to the restrictions associated with the particular content architecture in use and subject to proper designation and invocation in accordance with ISO 2022.

Any non-spacing characters included in a graphic character set are not to be used in isolation but only in combination with spacing characters.

6.6 Space characters

The character SPACE (SP) is considered both as a logical control function and as a graphic character. As a graphic character, it has a graphical representation consisting of the absence of a graphic symbol. As a control function, it indicates a potential line break point (see 14.2.1.3.2).

NOTE – NBSP (No Break SPace) and any fixed-width space characters, such as “digit space”, “em space” and “en space” are regarded as graphic characters i.e. are not regarded as line break points.

6.7 Coding of content information

The coded representation of the content information within a content portion is in accordance with the rules specified in ISO 2022. If ODIF is used as the interchange format, the coding shall be performed for an 8-bit environment.

NOTE 1 – This is equivalent to presuming a code extension announcer value of ESC 2/0 4/7 for the C1 set.

If ODL is used in the interchange format, the coding should where possible be performed for an 8-bit environment. (The use of ODL is applicable to ISO/IEC 8613 only.)

NOTE 2 – If an ODL application requires coding for a 7-bit environment, this should be specified by the document application profile, preferably also stating the appropriate code extension announcer(s).

Coded representations of control functions are defined in ISO 6429 and are summarized in Annex C.

6.8 Internal structure

6.8.1 Formatted content

Formatted content is content for which all the necessary information relating to the layout and imaging of that content has been specified. Content in this form is intended to be imaged as specified and is not intended to be revised by an editing process or to be reformatted.

The content of a basic component conforming to a formatted character content architecture consists of one or more lines of characters. Each pair of successive lines is separated by a hard line terminator. The end of the content of a basic component implicitly terminates the last line.

6.8.2 Processable content

Processable content is content which has not been laid out. Content in this form is suitable for revision by an editing process.

NOTE – The editing process is implementation dependent and is not described in ITU-T Rec. T.410-Series 1 ISO/IEC 8613.

In order to image content in this form, it is necessary to apply a content layout process (see clause 14) to the content which converts the processable content into formatted content (see 6.8.1) or into formatted processable content (see 6.8.3).

To assist the processing (i.e. editing or layout processes) of processable content, a number of logical presentation attributes and control functions have been defined (see clauses 9 and 13). In addition, the character SPACE is regarded both as a graphic character and as a control function that indicates where a line break may occur when the content is laid out.

The content of a basic component conforming to a processable character content architecture consists of one or more sequences of characters. Each pair of successive character sequences is separated by a hard line terminator control function.

At the end of the content of a basic logical component to which another basic logical component is concatenated (see ITU-T Rec. T.412 | ISO/IEC 8613-2), the last character sequence continues into the content of the next basic logical component. In all other cases, the end of the content of the basic logical component implicitly terminates the last character sequence.

The division into character sequences represents the internal structure of the processable content of a basic logical component. Each character sequence is anonymous, in that no name or identifier is associated with it, and no relationship exists among character sequences except that of sequence.

6.8.3 Formatted processable content

Formatted processable content is content that is structured such that it contains both the formatted content and the processable content as subsets. It is identical in structure to the processable content, except that it may contain additional control functions and graphic characters that have been added as a result of the content layout process. It is identical in structure to the formatted content, except that it may contain logical control functions and delimiters as well as deleted logical content.

Thus, formatted processable content can be converted to processable content by deleting (or ignoring) all layout control functions, all occurrences of the delimiters and all control functions and characters inserted within those delimiters.

Alternatively, formatted processable content can be converted to formatted content by deleting (or ignoring) all logical control functions and the delimiters as well as deleted content but retaining the control functions and characters inserted within the delimiters.

NOTE – The conversion of formatted processable content to processable content is a reversible process (providing the same layout constraints are applicable to the content layout process) but converting formatted processable content to formatted content is irreversible.

The formatted view of a basic component conforming to a formatted processable character content architecture consists of one or more lines of characters. Each pair of successive lines is separated by either a hard or soft line terminator. The end of the content of a basic layout component implicitly terminates the last line.

The processable view of a basic component conforming to a formatted processable character content architecture consists of one or more sequences of characters. Each pair of successive character sequences is separated by a hard line terminator.

At the end of the content of a basic logical component to which another basic logical component is concatenated (see ITU-T Rec. T.412 | ISO/IEC 8613-2), the last character sequence continues into the content of the next basic logical component. In all other cases, the end of the content of the basic logical component implicitly terminates the last character sequence.

Soft line terminators are used as separators between lines within a character sequence. The division into character sequences represents the internal structure of the content of a basic logical component. Each character sequence is anonymous, in that no name or identifier is associated with it, and no relationship exists among character sequences except that of sequence.

7 Character positioning

This clause specifies how characters are to be positioned within a basic layout object. The intention is to aid understanding of the presentation attributes and control functions that relate to character positioning.

This clause provides for the positioning of any font that is defined in accordance with ISO/IEC 9541-1. This clause also caters for the positioning of characters pertaining to different fonts within the same basic layout object.

7.1 Basic concepts

7.1.1 Character fonts

In the context of this part of ITU-T Rec. T.410-Series | ISO/IEC 8613, the term *graphic character* is used in its abstract sense; that is, this term refers to a member of a set of graphic symbols used for the representation of information. The term *character image* is then used to refer to the rendition of a graphic character on a presentation medium.

A *font* is a set of character images, normally with a common design and size. A set of font attributes is associated with the font as a whole and a set of character attributes is associated with each individual character. These attributes are defined in ISO/IEC 9541-1.

The main purpose of the font attributes is for the recipient to identify the font used by the originator and, in case the specified font is not available, the font and character attributes serve as guidance for the recipient to find an appropriate substitute font among those available.

Further information concerning the designation and invocation of different fonts within a basic object is given in clause 8.

7.1.2 Directions

In the context of this part of ITU-T Rec. T.410-Series I ISO/IEC 8613, all directions are expressed as counter clockwise angles of rotation (in degrees) relative to a specified reference direction (an example is given in Figure 1).

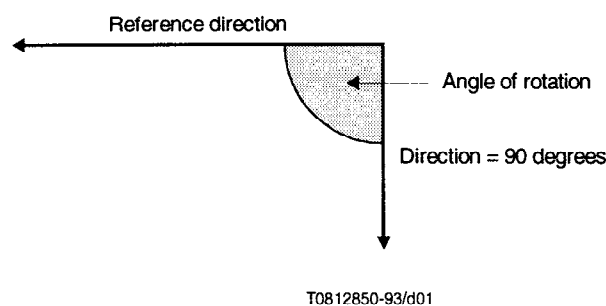


Figure 1 – Example of direction

The *character path* is the direction of progression of successive character images within a line box (defined in 7.1.7) and is expressed as a direction relative to the horizontal direction of the layout object (see Figure 4).

The *line progression* is the direction of progression of successive line boxes within the basic layout object and is expressed as a direction relative to the character path (see Figure 11).

The *character orientation* is the direction of the character baseline (defined in 7.1.3) relative to the character path.

Only one value for the character path, line progression and character orientation may be specified for a basic component.

7.1.3 Character image model

The *position point* is a reference point associated with a character image (see Figure 2). It is used for the positioning of the character image within a line box. The *escapement point* is a reference point associated with a character image (see Figure 2). It is used for the positioning of the next character image.

The *character baseline* is an imaginary line across a character image, for the purpose of defining the character orientation. The character baseline is a horizontal line when the character image is in its intended viewing orientation (see Figure 3).

A position point and escapement point shall be defined for each character orientation which is intended to be used (see Figure 3); i.e. “writing modes” corresponding to the required character orientations shall be defined in the font description, or fall-backs shall be defined in document application profiles.

A *kern* is that part of the character image that extends beyond its position and escapement points (see Figure 2).

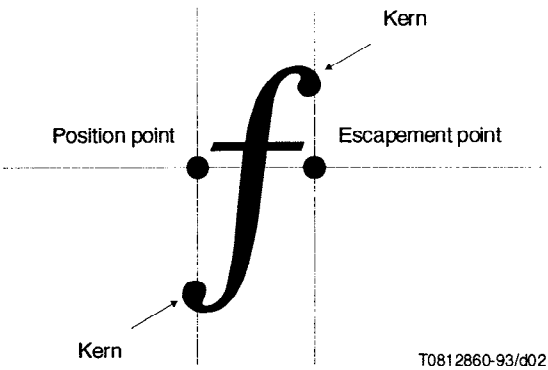
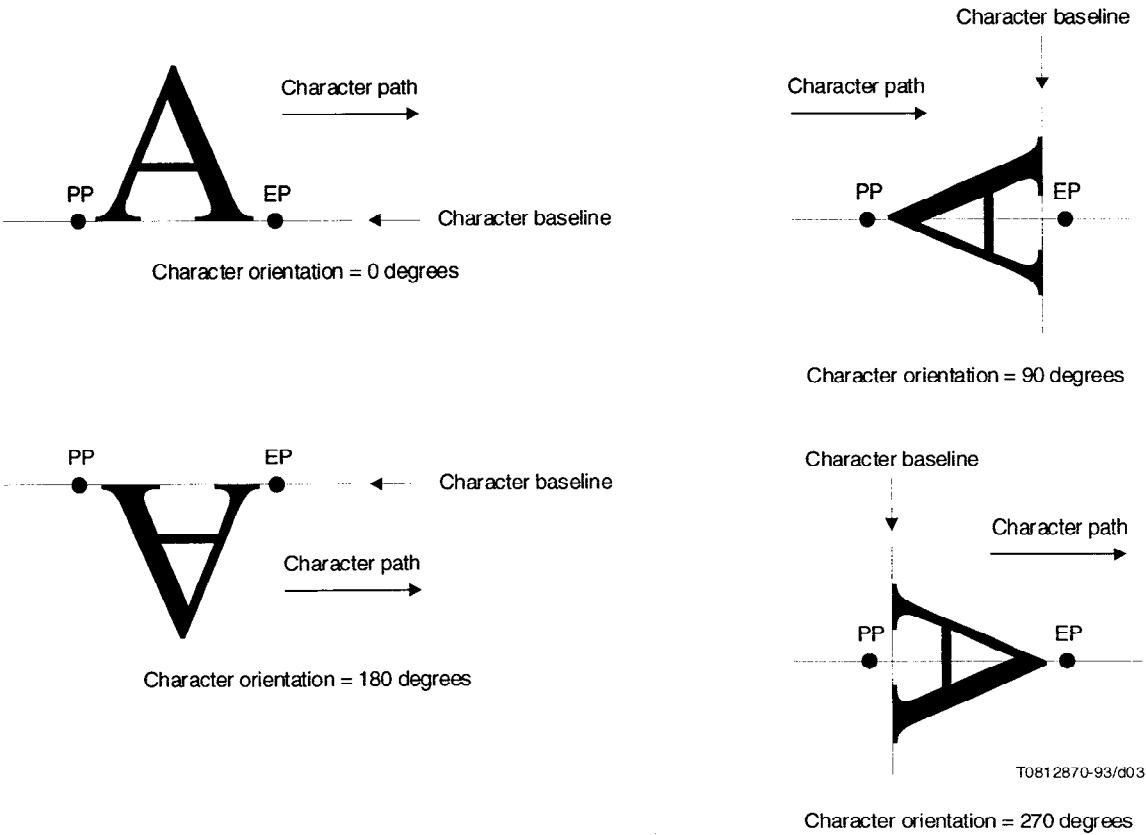


Figure 2 – Illustration of kerns



PP Position point
EP Escapement point

Figure 3 – Reference points for character image positioning

7.1.4 Character spacing

The concept of character spacing is only applicable when a constant spacing font is selected. It is used (in conjunction with the inter-character space) to determine the distance between character images within a line box as defined in 7.2.1.

The character spacing is equal to the distance between the position points of successive character images when the inter-character space is zero.

The character spacing is independent of the distance between the position point and escapement point of character images.

7.1.5 Active position

The *active position* is an abstraction of an imaging device concept such as a cursor. This concept is used in the definitions of control functions (see clause 13) where a sequential method of processing a character string is assumed.

The active position indicates the point, within the positioning area of a basic layout object, at which the action specified by the next character (graphic character or control function) is to be effected.

If the next character is a graphic character, its character image is positioned with the position point at the active position and the active position is advanced in the direction of the character path by the amount of spacing (defined in 7.2.1).

If the next character is a control function, this may cause the active position to move to another point within the positioning area.

7.1.6 Positioning area

A *positioning area* is a rectangular area, wholly contained within a basic layout object, within which position points and escapement points are to be positioned (see Figure 4). Kerns of character images are permitted to extend beyond the positioning area but are not permitted to extend beyond the edges of the basic layout object.

Two of the edges of the positioning area are referred to as the *start edge* and *end edge* (see Figure 4). The start edge and the end edge are defined such that the direction from the start edge to the end edge is in the direction of the character path.

The other two edges of the positioning area are referred to as the top edge and bottom edge. The top edge and the bottom edge are defined such that the direction from the top edge to the bottom edge is in the direction of line progression.

The start and end edges are indented from the corresponding edges of the basic layout object by a distance referred to as the *Kerning offset* (see Figure 4). The kerning offset specified depends upon the fonts of the characters to be positioned in the positioning area. Its value is chosen such that no part of any character image with kerns will extend beyond the boundary of the basic layout object when sequences of character images are positioned within it.

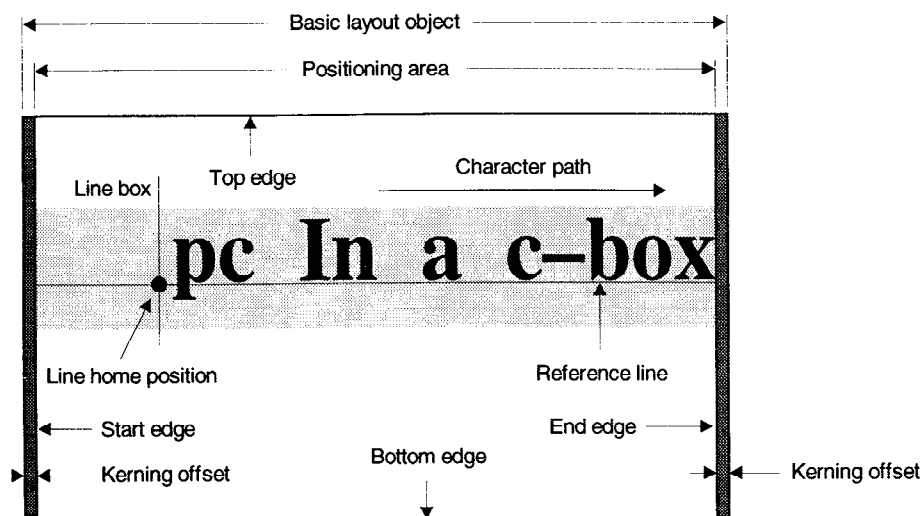
7.1.7 Line boxes

Within the positioning area, a sequence of character images is positioned within an area called a *line box* (see Figure 11). Each line box is a rectangular area that extends from the start edge to the end edge of the positioning area.

Each line box contains a reference point called the *line home position* (see Figure 4). This point is used for positioning the line box within the basic layout object. It also serves as the active position for the first graphic character or control function in each line.

Each line box contains an imaginary line called a *reference line* (see Figure 4). The reference line passes through the line home position in the direction of the character path. It extends from the start edge to the end edge within the line box and is used for the alignment of character images.

The length of the line box is equal to the distance between the start and end edges. The width (or height) of a line box is equal to the sum of the line box forward extent and the line box backward extent (see Figure 11). The line box backward extent is the distance between the reference line and the edge of the line box in the direction opposite to the direction of line progression. The line box forward extent is the distance between the reference line and the edge of the line box in the direction of line progression.



T0812880-93/d04

Figure 4 – Illustration of character positioning concepts

The values of the forward and backward extents depend on the maximum extents (measured perpendicular to the reference line) of the character fonts used in the line box. Determination of the forward and backward extents takes into account any displacements of character images perpendicular to the character path, e.g. for subscripts, superscripts and parallel annotation.

NOTE – As an example, for a Latin font with character path do, line progression d270 and when a single font is used in the line box, then the forward and backward extents are equal to the maximum right and maximum left extents for that font as defined in ISO/IEC 9541-1.

If no graphic characters are present in a line box, the font that is used to determine the backward and forward extents shall be the last font that is invoked in the line box, if any, or, otherwise, the font that was invoked or was assumed to be invoked (see 8.2.3) before the line box was started.

7.2 Positioning of character images within a line box

Successive character images are positioned within a line box in the direction of the character path.

The position points of the character images are aligned on the reference line unless the characters are imaged as subscripts, superscripts or parallel annotation.

There are several factors which affect the positioning of character images along the reference line:

- spacing between characters;
- alignment;
- tabulation;
- character ordering;
- parallel annotation;
- subscript/superscript;
- pairwise kerning;
- first line offset;
- itemization.

7.2.1 Spacing between characters

The *inter-character space* is an additional amount of spacing between the position points of successive character images, in the direction of the character path (see 7.1.4). A negative value indicates a reduction in the spacing between successive character images.

The distance between the position points of successive character images may be constant or variable depending upon the font as follows:

- For fonts with *constant spacing*, the distance between the position points of successive character images is independent of the characters and is the sum of the character spacing (as specified by presentation attributes and control functions) and the inter-character space (see Figure 5).
- For fonts with *variable spacing*, the distance between the position points of successive character images is dependent upon the character, i.e. normally the distance between the position point and the escapement point of a character, and is the sum of the net escapement of the character (as specified by the font) and the inter-character space (see Figure 6).



Figure 5 – Spacing for constant spacing font

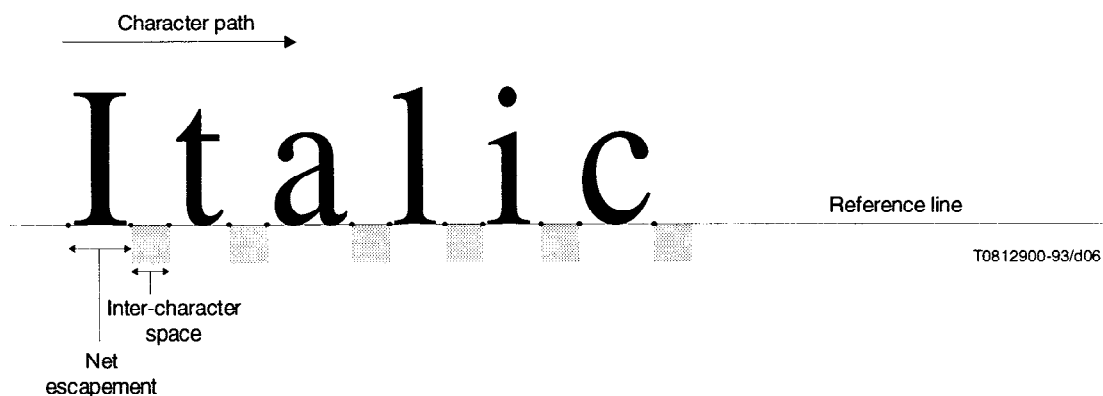


Figure 6 – Spacing for variable spacing font

The space width, i.e. the width of the SP (Space) character image, shall be determined as follows:

- for any SP that follows a soft line terminator and precedes the first graphic character of a line, or precedes a soft line terminator and follows the last graphic character of a line, the width is equal to zero;
- in a constant spacing font, the default width equals the character spacing;
- in a variable spacing font, the width is implicitly defined by the font;
- for all fonts, the width may be specified by a control function.

7.2.2 Alignment

The character images are positioned within a line box in accordance with the attribute “alignment” as follows:

- *Start-aligned* – The position point of the first character image is placed at the line home position.
- *End-aligned* – The escapement point of the last character is placed at the end edge of the positioning area.
- *Centred* – The distance from the line home position to the position point of the first character image is approximately equal to the distance from the escapement point of the last character image to the end edge of the positioning area.
- *Justified* – The position point of the first character image is placed at the line home position and the escapement point of the last character image is made coincident with the end edge of the positioning area by appropriately setting the space width and/or the inter-character space.

7.2.3 Tabulation

The position of character images along a reference line can be controlled by means of a set of *tabulation stops*. Each tabulation stop specifies a point along a reference line relative to the start edge of the positioning area.

A string of character images can be placed at a tabulation stop by means of a control function embedded in the text. The string may be start-aligned, end-aligned, centred or aligned around one or more specified characters within that string as follows (see Figure 7):

- *Start-aligned* – The position point of the first character image of the string is placed at the tabulation stop.
- *End-aligned* – The escapement point of the last character image of the string is placed at the tabulation stop.
- *Centred* – The string is placed such that the position point of the first character image and the escapement point of the last character image of the string are approximately equidistant from the tabulation stop.
- *Aligned-around* – The position point of the first character image of the first instance of the specified group of characters in that string is positioned at the tabulation stop. If the specified group of characters does not appear in the text associated with that tabulation stop, then the alignment defaults to end-aligned as defined above.

7.2.4 Character ordering

Within the content of a basic object, the interchange order is always the reading order of the language used.

In the case of certain languages, e.g. Arabic and Hebrew, where the alphanumeric text is read from right to left and the numeric text is read from left to right, the interchanged stream shall indicate the change in presentation direction at the appropriate point(s).

This is necessary since control functions in character content architectures are defined to operate sequentially according to their position in the character stream.

When a string of characters with reversed presentation direction is embedded in the text with normal presentation direction, the image of the last character of the string with reversed presentation direction is positioned adjacent to the image of the last character of the preceding string with normal presentation direction (see Figure 8).

NOTE – In Figure 8, the terms “first” and “last” are used in relation to the interchange order and the terms “normal” and “reversed” in relation to the direction of the character path.

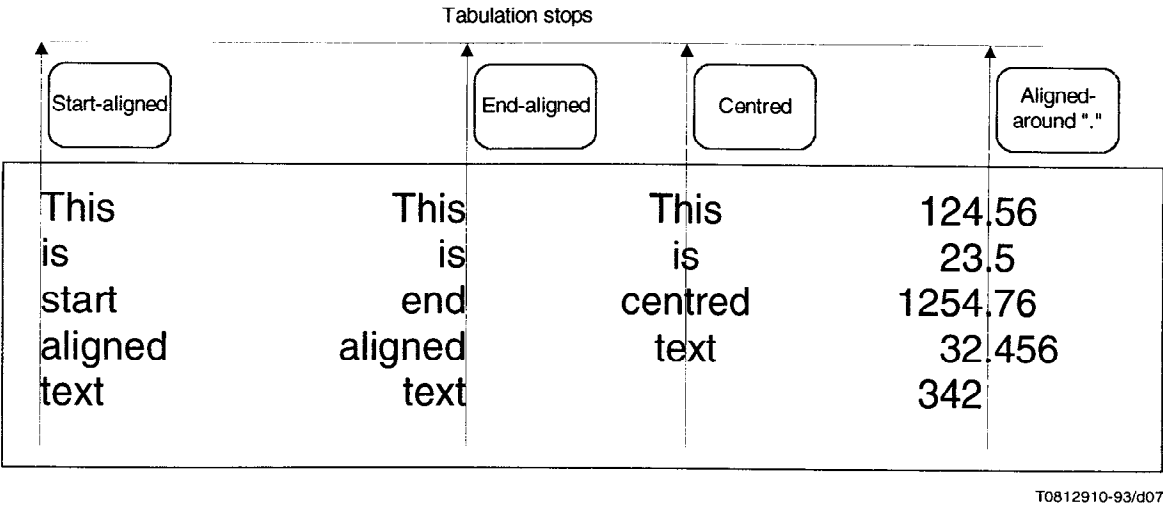


Figure 7 – Tabulation

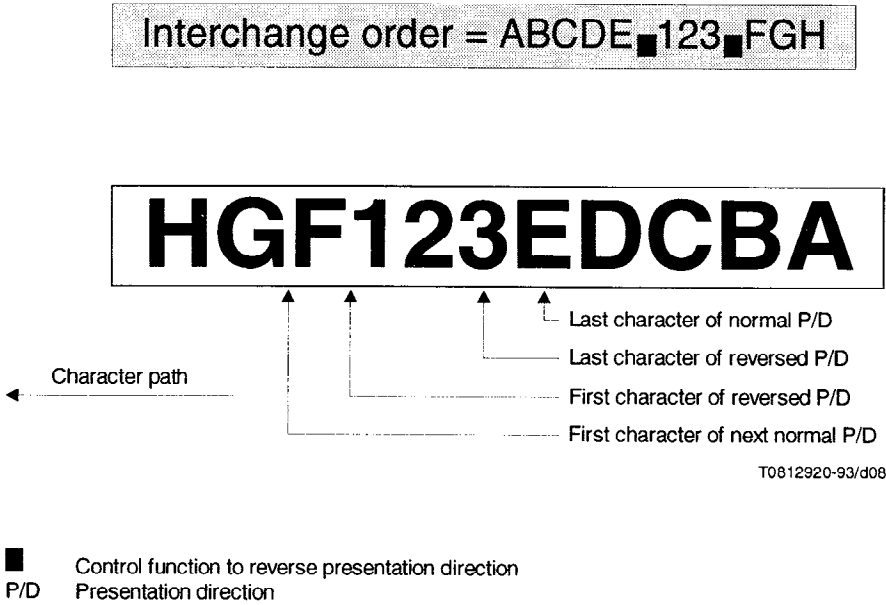
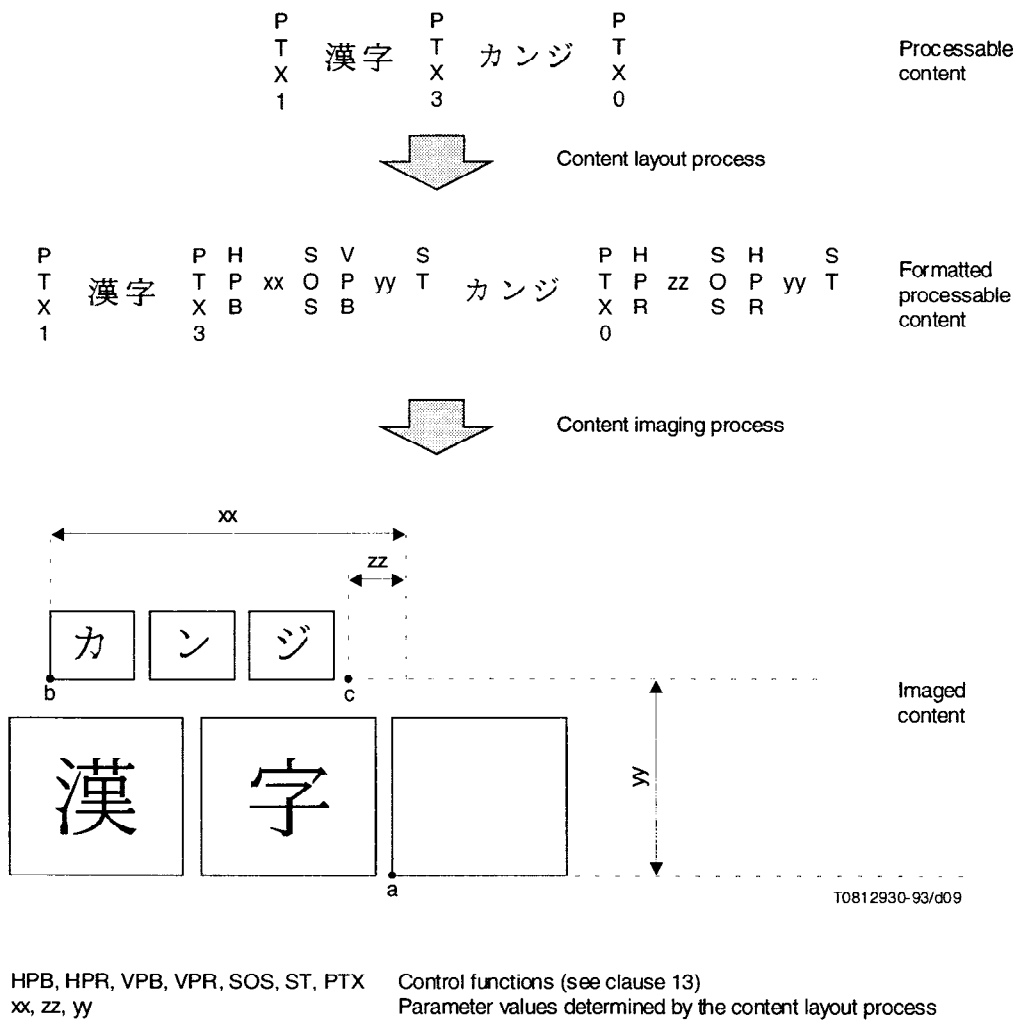


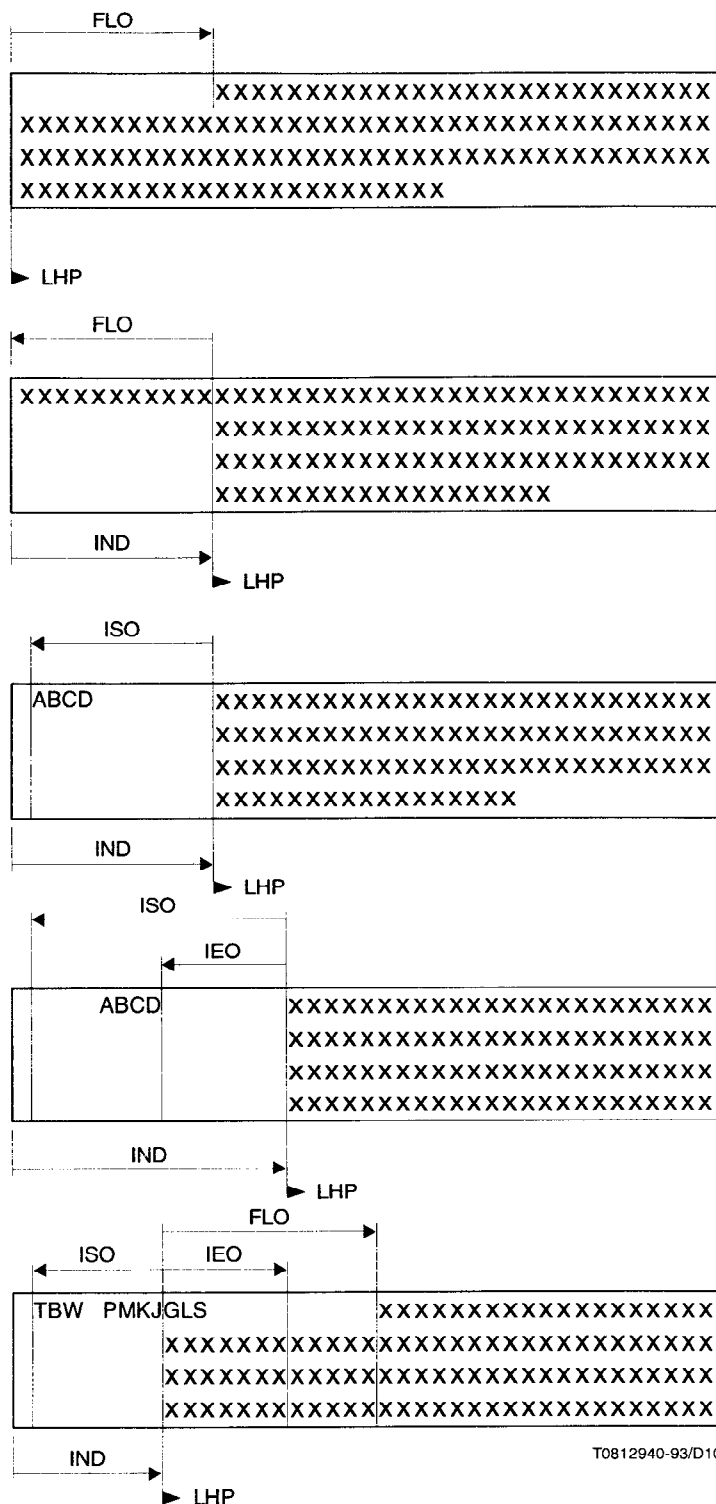
Figure 8 – Character ordering

7.2.5 Parallel annotation

Two character strings can be presented in parallel, by delimiting them such that the second string is used to indicate the pronunciation and interpretation of the first string. This feature is intended to be used for the Japanese language to provide pronunciation and interpretation information (Ruby) in the form of Kana character(s) for one or more Kanji characters. This is indicated in the formatted text by the Kana character(s) being centred either above or to the right of the Kanji character(s), for character image orientations orthogonal and parallel to the character path respectively (see Figure 9).

Where centering would result in Kana characters being positioned outside the positioning area, then the Kana character string is positioned such that it is start-aligned or end-aligned with the edge of the available area.





FLO First line offset
 ISO Identifier start offset
 IEO Identifier end offset
 LHP Line home position
 IND Indentation

Example 10.1

First line offset: positive
 Identifier alignment: no itemization
 Identifier start offset: zero
 Identifier end offset: zero
 Indentation: zero

Example 10.2

First line offset: negative
 Identifier alignment: no itemization
 Identifier start offset: zero
 Identifier end offset: zero
 Indentation: positive

Example 10.3

First line offset: zero
 Identifier alignment: start aligned
 Identifier start offset: negative
 Identifier end offset: zero
 Indentation: positive

Example 10.4

First line offset: zero
 Identifier alignment: end aligned
 Identifier start offset: negative
 Identifier end offset: negative
 Indentation: positive

Example 10.5

First line offset: positive
 Identifier alignment: start aligned
 Identifier start offset: negative
 Identifier end offset: positive
 Indentation: positive

Figure 10 – Illustration of itemization, first line offset and indentation

7.2.6 Subscript/superscript

Subscript rendition allows for the active position to be displaced from the reference line in the direction of line progression.

Superscript rendition allows for the active position to be displaced from the reference line in the direction opposite to that of line progression.

The combined effect of all subscript/superscript renditions within a line box shall be such that the active position is returned to the reference line before the occurrence of a hard or soft line terminator.

7.2.7 Pairwise kerning

Pairwise kerning allows for the moving of the active position from that defined by the preceding character. The distance and direction depend both on the character being imaged and the preceding character.

In the case of a constant spacing font, pairwise kerning is ignored.

In the case of a variable spacing font, the actual distance between the escapement point of one character image and the position point of the next character image is modified by the kerning information as defined in the character attributes of the font.

7.2.8 First line offset

First line offset allows for character imaging of the first line of a basic component to start at a position displaced from the line home position.

The offset is either in the direction of the character path (producing first line indentation) or in the direction opposite to the character path (producing overhang) as illustrated in Figure 10.

7.2.9 Itemization

Itemization allows for imaging of an item identifier on the first line of a basic component in positions which are not directly constrained by the line home position and the first line offset (see Figure 10).

An *item identifier* is a string of characters that precedes and is separated from the remainder of the first line of a basic component.

A start offset and an end offset are defined relative to the line home position. These offsets determine the location of the item identifier and the separation between the item identifier and the line home position.

The item identifier may be positioned on the first line in accordance with the item identifier alignment attribute as follows:

- *Start-aligned* – The position point of the first character image of the item identifier is placed at the start offset.
- *End-aligned* – The escapement point of the last character image of the item identifier is placed at the end offset.

7.3 Positioning of line boxes within a basic layout object

For positioning of line boxes in a basic layout object, the area of that object is independent of any adjoining areas. No part of the image shall be permitted to extend beyond the boundaries of the basic layout object.

The *initial point* is the point relative to which all line boxes are positioned within the basic layout object (see 9.2.2 and Figure 11).

The line home position of the first line box is at the initial point of the basic layout object. Subsequent line home positions are located on a line through the initial point in the direction of line progression.