INTERNATIONAL STANDARD

ISO/IEC 8613-10

First edition
1991-06-01

AMENDMENT 5
1993-05-15

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

Part 10:
Formal specifications

AMENDMENT 5 : Formal specification of the defaulting mechanism for defaultable attributes

Traitement de l'information – Bureautique – Architecture des documents de bureau (ODA) et format d'échange –

Partie 10: Spécifications formelles

AMENDEMENT 5 : Spécification formelle du mécanisme par défaut pour les attributs par défaut

Reference number
ISO/IEC 8613-10:1991/Amd.5:1993(E)

Documents
Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

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.

Amendment 5 to International Standard ISO/IEC 8613-10:1991 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology,* Sub-Committee SC 18, *Document processing and related communication.*

ISO/IEC 8613 consists of the following parts, under the general tile *Information processing – Text and office systems – Office Document Architecture (ODA) and interchange format*:

- *Part 1: Introduction and general principles*
- *Part 2: Document structures*
- *Part 4: Document profile*
- *Part 5: Office Document Interchange Format (ODIF)*
- *Part 6: Character content architectures*
- *Part 7: Raster graphics content architectures*
- *Part 8: Geometric graphics content architectures*
- *Part 10: Formal specifications*

Annex F forms an integral part of this part of ISO/IEC 8613.

© ISO/IEC 1993

All rights reserved. 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

Documents
Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

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

Part 10:
Formal specifications

AMENDMENT 5 : Formal specification of the defaulting mechanism for defaultable attributes

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Instructions for merging this Amendment into ISO/IEC 8613-10:

— In Contents add:

Annex F: Formal specification of the defaulting mechanism for defaultable attributes

— In the definition of the production rules (subclause 4.2), the following line is added to the definition of a "term":

"particularization-term |"

— The following production rule is added:

"particularization-term ::=
THAT var (formula)

NOTE – The semantics of the terminal symbol THAT in this production rule are specified in subclause 4.5".

— In subclause 4.5, the following text is added:

"THAT var (formula) The particularizator THAT makes a term from a formula. For a formula f, the term "THAT x (f)" denotes that entity which, if assigned to x, makes f true. In general, this makes sense only if f contains x as a free variable. If there is not exactly one entity (i.e., none or more than one) satisfying f, the term is undefined. Formally, particularization is characterized by

$$\forall y\ (y = \text{THAT } x\ (\text{f}) \Longleftrightarrow (\exists !x(\text{f}) \wedge \text{f}_y) \vee (\neg \exists !x(\text{f}) \wedge y = \mathsf{UNDEF}))$$

where f_y stands for $\exists\, x(\text{f} \wedge x = y)$, $\exists !x(\text{f})$ stands for $\exists\, y(\text{f}_y \wedge \forall\, x(\text{f} \Longrightarrow x = y))$, and f is assumed not to contain y as a free variable."

The Annex contained on the following pages is to be added to ISO/IEC 8613-10.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Annex F

(normative)

Formal specification of the defaulting mechanism for defaultable attributes

F.1 Introduction

This annex gives a formal specification of the defaulting mechanism for defaultable attributes.

Clause F.2 defines a set of general functions which are used in clause F.3.

Clause F.3 defines a set of functions which describe the derivation of all defaultable attributes in ISO 8613. It is substructured into clauses F.3.1 (defaultable attributes of ISO 8613-2), clauses F.3.2 (defaultable attributes of ISO 8613-6), clauses F.3.3 (defaultable attributes of ISO 8613-7) and clauses F.3.4 (defaultable attributes of ISO 8613-8).

Clause F.4 is an index for the predicate symbols, operator symbols and attribute names used in clauses F.2 and F.3.

F.2 General functions

This clause defines the general functions for determining the values of defaultable attributes according to clause 5.1.2.4 of ISO 8613-2.

Semiformal Description 9.1

Function "x ORELSE y"

The function x ORELSE y returns the value x if x is not undefined, otherwise the value y.

Definition 9.1

1 $\forall x, y$
2 $({}_{0}\, x \text{ ORELSE } y =$
3 IF $x \neq$ UNDEF THEN x ELSE $y_{0})$

Semiformal Description 9.2

Function "Attribute value by step BL"

If a layout style is referenced by a constituent *cst* and if the attribute *att* is specified for this layout style (3) then its value is the value determined by step BL (4). Otherwise the value is undefined by step BL.

Definition 9.2

1 $\forall cst, doby, att$
2 $({}_{0}\text{STEP_BL_VALUE}(cst, doby, att) =$
3 IF 'layout style' $\in \text{NAMS}^{1.18}(cst)$ *and* $att \in \text{NAMS}^{1.18}(\text{REF_LAY_STYLE}^{9.4}(cst, doby))$
4 THEN C ^$(\text{REF_LAY_STYLE}^{9.4}(cst, doby)) \bullet att$
5 ELSE UNDEF$_{0})$

Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.3

Function "Attribute value by step BP"

If a presentation style is referenced by a constituent *cst* and if the attribute *att* is specified for this presentation style (3) then its value is the value determined by step BP (4). Otherwise the value is undefined by step BP.

Definition 9.3

```
1   ∀ cst, doby, att
2   (₀STEP_BP_VALUE(cst, doby, att) =
3     IF 'presentation style' ∈ NAMS^1.18(cst) and att ∈ NAMS^1.18(REF_PRES_STYLE^9.5(cst, doby))
4     THEN C ^(REF_PRES_STYLE^9.5(cst, doby)) • att
5     ELSE UNDEF₀)
```

Semiformal Description 9.4

Function "Referenced layout style"

If a layout style exists within the document body *doby* for which the value of the attribute 'layout style identifier' is the same as the value of the attribute 'layout style' of a particular constituent *cst* this is called the referenced layout style (3) which is returned by this function. Otherwise, the function returns the empty nomination.

Definition 9.4

```
1   ∀ cst, doby
2   (₀REF_LAY_STYLE(cst, doby) =
3     C ^doby • <IsLayoutStyle^2.60(C xs) and C xs • 'layout style identifier' = C ^cst • 'layout style'>
4     ORELSE^9.1 [ : ]₀)
```

Semiformal Description 9.5

Function "Referenced presentation style"

If a presentation style exists within the document body *doby* for which the value of the attribute 'presentation style identifier' is the same as the value of the attribute 'presentation style' of a particular constituent *cst* this is called the referenced presentation style (3, 4) which is returned by this function. Otherwise, the function returns the empty nomination.

Definition 9.5

```
1   ∀ cst, doby
2   (₀REF_PRES_STYLE(cst, doby) =
3     C ^doby • <IsPresentationStyle^2.61(C xs) and
4               C xs • 'presentation style identifier' = C ^cst • 'presentation style'>
5     ORELSE^9.1 [ : ]₀)
```


Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.6

Function "Attribute value by step C"

If an object class is referenced by a constituent *cst* and if the attribute *att* is specified for this object class (3) then its value is the value determined by step C (4). Otherwise the value is undefined by step C.

Definition 9.6

1 $\forall\, cst, doby, att$
2 $({}_{0}\mathrm{STEP_C_VALUE}(cst, doby, att) =$
3 IF 'object class' $\in \mathrm{NAMS}^{1.18}(cst)$ *and* $att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))$
4 THEN C ^$(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)) \bullet att$
5 ELSE UNDEF_{0})

Semiformal Description 9.7

Function "Referenced object class"

If an object class exists within the document body *doby* for which the value of the attribute 'object class identifier' is the same as the value of the attribute 'object class' of a particular constituent *cst* this is called the referenced object class (3) which is returned by this function. Otherwise, the function returns the empty nomination.

Definition 9.7

1 $\forall\, cst, doby$
2 $({}_{0}\mathrm{REF_OBJECT_CLASS}(cst, doby) =$
3 C ^$doby \bullet$ <IsObjectClassDescription$^{2.37}$(C xs) *and* C $xs \bullet$ 'object class identifier' = C ^$cst \bullet$ 'object class'>
4 ORELSE$^{9.1}$ [:]$_{0}$)

Semiformal Description 9.8

Function "Attribute value by step DP"

If an object class is referenced by a constituent *cst*, and a presentation style is referenced by this object class and the attribute *att* is specified for this presentation style (3-5) then its value is the value determined by step DP (6). Otherwise the value is undefined by step DP.

Definition 9.8

1 $\forall\, cst, doby, att$
2 $({}_{0}\mathrm{STEP_DP_VALUE}(cst, doby, att) =$
3 IF 'object class' $\in \mathrm{NAMS}^{1.18}(cst)$ *and*
4 'presentation style' $\in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
5 $att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby), doby))$
6 THEN C ^$(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby), doby)) \bullet att$
7 ELSE UNDEF_{0})

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.9

Function "Attribute value by step DL"

If an object class is referenced by a constituent *cst*, and a layout style is referenced by this object class and the attribute *att* is specified for this layout style (3-5) then its value is the value determined by step DL (6). Otherwise the value is undefined by step DL.

Definition 9.9

```
1  ∀ cst, doby, att
2  (0 STEP_DL_VALUE(cst, doby, att) =
3    IF 'object class' ∈ NAMS^1.18(cst) and
4       'layout style' ∈ NAMS^1.18(REF_OBJECT_CLASS^9.7(cst, doby)) and
5       att ∈ NAMS^1.18(REF_LAY_STYLE^9.4(REF_OBJECT_CLASS^9.7(cst, doby), doby))
6    THEN C ^(REF_LAY_STYLE^9.4(REF_OBJECT_CLASS^9.7(cst, doby), doby)) • att
7    ELSE UNDEF0)
```

Semiformal Description 9.10

Function "Attribute value by step E"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class and the attribute *att* is specified for this (resource) object class (3–6) then this attribute value is the value determined by step E (7, 8). Otherwise the value is undefined by step E.

Definition 9.10

```
1  ∀ cst, doby, rdoby, prof, att
2  (0 STEP_E_VALUE(cst, doby, rdoby, prof, att) =
3    IF 'object class' ∈ NAMS^1.18(cst) and
4       'resource' ∈ NAMS^1.18(REF_OBJECT_CLASS^9.7(cst, doby)) and
5       att ∈ NAMS^1.18(RESOURCE_OBJECT_CLASS^9.11(rdoby,
6                       OBJECT_CLASSID_INRES^9.12(prof, REF_OBJECT_CLASS^9.7(cst, doby))))
7    THEN C ^(RESOURCE_OBJECT_CLASS^9.11(rdoby,
8             OBJECT_CLASSID_INRES^9.12(prof, REF_OBJECT_CLASS^9.7(cst, doby)))) • att
9    ELSE UNDEF0)
```

Semiformal Description 9.11

Function "Resource object class"

If an object class exists within the document body *rdoby* (of a resource document) for which the value of the attribute 'object class identifier' corresponds to *objcid*, this is called the resource object class (3) which is returned by this function. Otherwise, the function returns the empty nomination.

Definition 9.11

```
1  ∀ rdoby, objcid
2  (0 RESOURCE_OBJECT_CLASS(rdoby, objcid) =
3    C ^rdoby • <IsObjectClassDescription^2.37(C xs) and C xs • 'object class identifier' = objcid>
4    ORELSE^9.1 [ : ]0)
```


Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.12

Function "Object class identifier in a resource document"

The object class identifier of an object class in a resource document is determined by the attribute 'resources' in the document profile by taking the second component of that construct in the value of this attribute for which the first component corresponds to the value of the attribute 'resource' for a particular constituent *cst*. If such a construct cannot be found the value of the function is undefined.

Definition 9.12

$$
\begin{array}{ll}
1 & \forall\, prof, cst \\
2 & ({}_{0}\mathrm{OBJECT_CLASSID_INRES}(prof, cst) = \\
3 & \quad \mathrm{LASTC}^{1.14}\ (\mathrm{C}\ \hat{}\ prof \bullet \text{'resources'} \bullet \mathrm{<HEAD}^{1.13}(\mathrm{C}\ xs) = \mathrm{C}\ \hat{}\ cst \bullet \text{'resource'} \mathrm{>}) \bullet {}_{0})
\end{array}
$$

Semiformal Description 9.13

Function "Attribute value by step FP"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class, and a presentation style is referenced by the (resource) object class and the attribute *att* is specified for this presentation style (3–8) then this attribute value is the value determined by step FP (9, 10). Otherwise the value is undefined by step FP.

Definition 9.13

$$
\begin{array}{ll}
1 & \forall\, cst, doby, rdoby, prof, att \\
2 & ({}_{0}\mathrm{STEP_FP_VALUE}(cst, doby, rdoby, prof, att) = \\
3 & \quad \mathrm{IF}\ \text{'object class'} \in \mathrm{NAMS}^{1.18}(cst)\ \underline{and} \\
4 & \qquad \text{'resource'} \in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))\ \underline{and} \\
5 & \qquad \text{'presentation style'} \in \mathrm{NAMS}^{1.18}(\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
6 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))\ \underline{and} \\
7 & \qquad att \in \mathrm{NAMS}^{1.18}(\mathrm{RESOURCE_PRES_STYLE}^{9.14}(rdoby, \\
8 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \\
9 & \quad \mathrm{THEN}\ \mathrm{C}\ \hat{}\ (\mathrm{RESOURCE_PRES_STYLE}^{9.14}(rdoby, \\
10 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \bullet att \\
11 & \quad \mathrm{ELSE\ UNDEF}_{0})
\end{array}
$$

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.14

Function "Presentation style in a resource document"

The presentation style in a resource document belonging to an object class with the object class identifier *objcid* is that constituent *rpres* in a resource document *rdoby* which is a presentation style (3, 4) and for which there exists a constituent *rcst* in the resource document (5) such that the value of the attribute 'presentation style identifier' of the presentation style is equal to the value of the attribute 'presentation style' of the constituent *rcst* and the value of the attribute 'object class identifier' of the constituent *rcst* equals *objcid* (6, 7). If such a presentation style cannot be found the resource presentation style is undefined.

Definition 9.14

1 $\forall\, rdoby, objcid$
2 $({}_0\,\mathrm{RESOURCE_PRES_STYLE}(rdoby, objcid) =$
3 THAT $rpres$
4 $({}_1\, rpres \in rdoby$ *and* $\mathrm{IsPresentationStyle}^{2.61}(rpres)$ *and*
5 $\exists\, rcst \in rdoby$
6 $({}_2\,\mathrm{C}\ \hat{}\ rpres \bullet$ 'presentation style identifier' $= \mathrm{C}\ \hat{}\ rcst \bullet$ 'presentation style' *and*
7 $\mathrm{C}\ \hat{}\ rcst \bullet$ 'object class identifier' $= objcid_2)_1)_0)$

Semiformal Description 9.15

Function "Attribute value by step FL"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class, and a layout style is referenced by the (resource) object class and the attribute *att* is specified for this layout style (3–8) then this attribute value is the value determined by step FL (9, 10). Otherwise the value is undefined by step FL.

Definition 9.15

1 $\forall\, cst, doby, rdoby, prof, att$
2 $({}_0\,\mathrm{STEP_FL_VALUE}(cst, doby, rdoby, prof, att) =$
3 IF 'object class' $\in \mathrm{NAMS}^{1.18}(cst)$ *and*
4 'resource' $\in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
5 'layout style' $\in \mathrm{NAMS}^{1.18}(\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby,$
6 $\mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))$ *and*
7 $att \in \mathrm{NAMS}^{1.18}(\mathrm{RESOURCE_LAY_STYLE}^{9.16}(rdoby,$
8 $\mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))$
9 THEN $\mathrm{C}\ \hat{}\ (\mathrm{RESOURCE_LAY_STYLE}^{9.16}(rdoby,$
10 $\mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \bullet att$
11 ELSE $\mathrm{UNDEF}_0)$

Documents Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.16

Function "Layout style in a resource document"

The layout style in a resource document belonging to an object class with the object class identifier *objcid* is that constituent *rlay* in a resource document *rdoby* which is a layout style (3, 4) and for which there exists a constituent *rcst* in the resource document (5) such that the value of the attribute 'layout style identifier' of the layout style is equal to the value of the attribute 'layout style' of the constituent *rcst* and the value of the attribute 'object class identifier' of the constituent *rcst* equals *objcid* (6, 7). If such a layout style cannot be found the resource layout style is undefined.

Definition 9.16

1 $\forall\, rdoby, objcid$
2 $({}_{0}\text{RESOURCE_LAY_STYLE}(rdoby, objcid) =$
3 THAT $rlay$
4 $({}_{1}\, rlay \in rdoby$ *and* $\text{IsLayoutStyle}^{2.60}(rlay)$ *and*
5 $\exists\, rcst \in rdoby$
6 $({}_{2}\, \text{C}\ \hat{}\, rlay \bullet \text{'layout style identifier'} = \text{C}\ \hat{}\, rcst \bullet \text{'layout style'}$ *and*
7 $\text{C}\ \hat{}\, rcst \bullet \text{'object class identifier'} = objcid_{2})_{1})_{0})$

Semiformal Description 9.17

Function "Attribute value by step G"

If there exists a constituent *sup* in the document body to which the constituent *cst* is immediately subordinate (3) and if the attribute value can be found (in a default value list) on the superior constituent *sup* (3, 4) then this attribute value is the value determined by step G. Otherwise, if a superior constituent exists, the search for the attribute value according to step G is continued (recursively) on the superior constituent *cst* (6). Otherwise, if no superior constituent exists or if the attribute value cannot be found there, the value is undefined by step G.

Definition 9.17

1 $\forall\, cst, doby, rdoby, prof, att, ctype$
2 $({}_{0}\text{STEP_G_VALUE}(cst, doby, rdoby, prof, att, ctype) =$
3 $\text{SUPERIOR_VALUE}^{9.18}({}_{1}\text{THAT}\ sup\ ({}_{2}\, sup \in doby$ *and* $(cst)\text{DescribesImSubOf}^{2.152}(sup)_{2}),$
4 $doby, rdoby, prof, att, ctype_{1})$
5 $\text{ORELSE}^{9.1}\ \text{STEP_G_VALUE}^{9.17}({}_{3}\text{THAT}\ sup\ ({}_{4}\, sup \in doby$ *and* $(cst)\text{DescribesImSubOf}^{2.152}(sup)_{4}),$
6 $doby, rdoby, prof, att, ctype_{3})_{0})$

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.18

Function "Superior value"

If the attribute value can be found in the default value list of the constituent *cst*, then this value is returned as the superior value (3). Otherwise, if the constituent refers to an object class (4) and the attribute value can be found in the default value list of the referenced object class, then this value is returned as the superior value (5). Otherwise, if the referenced object class refers to a resource document (6) and the attribute value can be found in the resource default value list of the referenced object class, then this value is returned as the superior value (7). Otherwise the superior value is undefined.

Definition 9.18

```
1   ∀ cst, doby, rdoby, prof, att, ctype
2   (0 SUPERIOR_VALUE(cst, doby, rdoby, prof, att, ctype) =
3     DEFAULT_VALUE^9.19(cst, att, ctype, doby) ORELSE^9.1
4     (1 IF 'object class' ∈ NAMS^1.18(cst)
5       THEN (2 DEFAULT_VALUE^9.19(REF_OBJECT_CLASS^9.7(cst, doby), att, ctype, doby) ORELSE^9.1
6       (3 IF 'resource' ∈ NAMS^1.18(REF_OBJECT_CLASS^9.7(cst, doby))
7         THEN (4 RES_DEFAULT_VALUE^9.21(REF_OBJECT_CLASS^9.7(cst, doby), rdoby, prof, att, ctype)
8               ORELSE^9.1 UNDEF4)
9         ELSE UNDEF3)2)
10      ELSE UNDEF1)0)
```

Semiformal Description 9.19

Function "Default value"

If a default value list is specified for the constituent *cst* (4), and default values for the particular type of constituent are specified (5, 6) and a default value for the particular attribute is specified (7, 8) then this value is the default value. Otherwise it is tested whether a style is referenced from the default value list which provides a value for the attribute (9). This function reflects the two cases that one (7) or more (8) sets of default values are specified, i.e., (a) one set is specified for composite object types (7); (b) one or more sets may be specified for basic object types (8).

Definition 9.19

```
1   ∀ cst, att, ctype, doby
2   (0 DEFAULT_VALUE(cst, att, ctype, doby) =
3     C (1 THAT a
4         (2 'default value lists' ∈ NAMS^1.18(cst) and
5           ∃ v
6           (3 v ∈̂ cst • 'default value lists' • and v = ctype and
7             (4 (5 IsNom(C v) and a = v • att5) or
8               (6 IsCol(C v) and a = v • • att6)4)3)2)
9     ORELSE^9.1 DEFAULT_VALUE_BY_STYLE^9.20(cst, doby, att, ctype)1)0)
```


Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.20

Function "Default value by style"

If the attribute 'default value lists' is specified (3), contains an entry for the particular type of constituent (4, 5), there exists a reference to a layout style (7) and this layout style contains the attribute (8), then the value of this attribute is the default value supplied by a style. Similarly, the value may also be supplied by a reference to a presentation style from within a default value list (10–15). This function reflects the two cases that one (10–12) or more (13–15) sets of default values are specified, i.e., (a) one set is specified for composite object types; (b) one or more sets may be specified for basic object types.

Definition 9.20

```
1   ∀ cst, doby, att, ctype
2   (0 DEFAULT_VALUE_BY_STYLE(cst, doby, att, ctype) =
3     IF 'default value lists' ∈ NAMS^1.18(cst) and
4     ∃ v
5     (1 v ∈̂ ^cst • 'default value lists' • and v = ctype and
6       ∃ a
7       (2 (3 IsNom(C v) and a = v • 'layout style' and
8           att ∈ NAMS^1.18(REF_LAY_STYLE^9.4(C v, doby))
9           THEN C ^(REF_LAY_STYLE^9.4(C v, doby)) • att3) or
10        (4 IsNom(C v) and a = v • 'presentation style' and
11          att ∈ NAMS^1.18(REF_PRES_STYLE^9.5(C v, doby))
12          THEN C ^(REF_PRES_STYLE^9.5(C v, doby)) • att4) or
13        (5 IsCol(C v) and a = v • • 'presentation style' and
14          att ∈ NAMS^1.18(REF_PRES_STYLE^9.5(C v •, doby))
15          THEN C ^(REF_PRES_STYLE^9.5(C v •, doby)) • att5)2)1)0)
```

Semiformal Description 9.21

Function "Resource default value"

If a default value is specified in a resource object class referenced by the object class *cst*, then this value is the resource default value. Otherwise the resource default value is undefined.

Definition 9.21

```
1   ∀ cst, rdoby, prof, att, ctype
2   (0 RES_DEFAULT_VALUE(cst, rdoby, prof, att, ctype) =
3     DEFAULT_VALUE^9.19(RESOURCE_OBJECT_CLASS^9.11(rdoby,
4                         OBJECT_CLASSID_INRES^9.12(prof, cst)), att, ctype, rdoby)0)
```

Semiformal Description 9.22

Function "Attribute value by step H"

If the attribute 'document application profile defaults' is specified in the document profile and this attribute specifies a value for the attribute *att* (3) then this is the value determined by step H. Otherwise the value is undefined by step H.

Definition 9.22

```
1   ∀ prof, att
2   (0 STEP_H_VALUE(prof, att) =
3     C (1 THAT a (2 a ∈̂ ^prof • 'document application profile defaults' and N a = att2)1)0)
```

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.23

Function "Parameter value by step BL"

If a layout style is referenced by a constituent *cst* and if the attribute *att* is specified for this layout style (3) and if the parameter *par* is specified for this attribute (4), then its value is the value determined by step BL (5). Otherwise the value is undefined by step BL.

Definition 9.23

1 $\forall\, cst, doby, att, par$
2 $({}_{0}\mathrm{STEP_BL_PAR_VALUE}(cst, doby, att, par) =$
3 IF 'layout style' $\in \mathrm{NAMS}^{1.18}(cst)$ *and* $att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_LAY_STYLE}^{9.4}(cst, doby))$ *and*
4 $par \in \mathrm{NAMS}^{1.18}(\mathrm{C}\ \hat{}\mathrm{REF_LAY_STYLE}^{9.4}(cst, doby) \bullet att)$
5 THEN C $\hat{}(\mathrm{REF_LAY_STYLE}^{9.4}(cst, doby)) \bullet att \bullet par$
6 ELSE UNDEF_{0})

Semiformal Description 9.24

Function "Parameter value by step C"

If an object class is referenced by a constituent *cst* and if the attribute *att* is specified for this object class (3) and if the parameter *par* is specified for the attribute (4), then its value is the value determined by step C (5). Otherwise the value is undefined by step C.

Definition 9.24

1 $\forall\, cst, doby, att, par$
2 $({}_{0}\mathrm{STEP_C_PAR_VALUE}(cst, doby, att, par) =$
3 IF 'object class' $\in \mathrm{NAMS}^{1.18}(cst)$ *and* $att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
4 $par \in \mathrm{NAMS}^{1.18}(\mathrm{C}\ \hat{}\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby) \bullet att)$
5 THEN C $\hat{}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)) \bullet att \bullet par$
6 ELSE UNDEF_{0})

Semiformal Description 9.25

Function "Parameter value by step DL"

If an object class is referenced by a constituent *cst*, and a layout style is referenced by this object class and the attribute *att* is specified for this layout style (3-5) and the parameter *par* is specified for this attribute (6), then its value is the value determined by step DL (7). Otherwise the value is undefined by step DL.

Definition 9.25

1 $\forall\, cst, doby, att, par$
2 $({}_{0}\mathrm{STEP_DL_PAR_VALUE}(cst, doby, att, par) =$
3 IF 'object class' $\in \mathrm{NAMS}^{1.18}(cst)$ *and*
4 'layout style' $\in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
5 $att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby), doby))$ *and*
6 $par \in \mathrm{NAMS}^{1.18}(\mathrm{C}\ \hat{}\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby), doby) \bullet att)$
7 THEN C $\hat{}(\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby), doby) \bullet att \bullet par$
8 ELSE UNDEF_{0})

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.26

Function "Parameter value by step E"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class and the attribute *att* is specified for this (resource) object class (3–6) and the parameter *par* is specified for this attribute (7, 8) then this attribute value is the value determined by step E (9, 10). Otherwise the value is undefined by step E.

Definition 9.26

$$
\begin{array}{ll}
1 & \forall\, cst, doby, rdoby, prof, att, par \\
2 & ({}_{0}\mathrm{STEP_E_PAR_VALUE}(cst, doby, rdoby, prof, att, par) = \\
3 & \quad \mathrm{IF}\ \text{'object class'} \in \mathrm{NAMS}^{1.18}(cst)\ \underline{and} \\
4 & \qquad \text{'resource'} \in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))\ \underline{and} \\
5 & \qquad att \in \mathrm{NAMS}^{1.18}(\ \mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
6 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))\ \underline{and} \\
7 & \qquad par \in \mathrm{NAMS}^{1.18}(\mathrm{C}\ \hat{}\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
8 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))) \cdot att) \\
9 & \quad \mathrm{THEN\ C}\ \hat{}(\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
10 & \qquad\quad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \cdot att \cdot par \\
11 & \quad \mathrm{ELSE\ UNDEF}_{0})
\end{array}
$$

Semiformal Description 9.27

Function "Parameter value by step FL"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class, and a layout style is referenced by the (resource) object class and the attribute *att* is specified for this layout style (3–8) and the parameter *par* is specified for this attribute (9, 10), then this attribute value is the value determined by step FL (11, 12). Otherwise the value is undefined by step FL.

Definition 9.27

$$
\begin{array}{ll}
1 & \forall\, cst, doby, rdoby, prof, att, par \\
2 & ({}_{0}\mathrm{STEP_FL_PAR_VALUE}(cst, doby, rdoby, prof, att, par) = \\
3 & \quad \mathrm{IF}\ \text{'object class'} \in \mathrm{NAMS}^{1.18}(cst)\ \underline{and} \\
4 & \qquad \text{'resource'} \in \mathrm{NAMS}^{1.18}(\mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))\ \underline{and} \\
5 & \qquad \text{'layout style'} \in \mathrm{NAMS}^{1.18}(\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
6 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))\ \underline{and} \\
7 & \qquad att \in \mathrm{NAMS}^{1.18}(\ \mathrm{RESOURCE_LAY_STYLE}^{9.16}(rdoby, \\
8 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))))\ \underline{and} \\
9 & \qquad par \in \mathrm{NAMS}^{1.18}(\mathrm{C}\ \hat{}\mathrm{RESOURCE_LAY_STYLE}^{9.16}(rdoby, \\
10 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby))) \cdot att) \\
11 & \quad \mathrm{THEN\ C}\ \hat{}(\mathrm{RESOURCE_LAY_STYLE}^{9.16}(rdoby, \\
12 & \qquad\quad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, \mathrm{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \cdot att \cdot par \\
13 & \quad \mathrm{ELSE\ UNDEF}_{0})
\end{array}
$$

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.28

Function "Parameter value by step G"

If there exists a constituent *sup* in the document body to which the constituent *cst* is immediately subordinate (3) and if the parameter value can be found (in a default value list) on the superior constituent *sup* (3, 4) then this parameter value is the value determined by step G. Otherwise, if a superior constituent exists, the search for the parameter value according to step G is continued (recursively) on the superior constituent *cst* (6). Otherwise, if no superior constituent exists or if the parameter value cannot be found there, the value is undefined by step G.

Definition 9.28

```
1  ∀ cst, doby, rdoby, prof, att, par, ctype
2  (0 STEP_G_PAR_VALUE(cst, doby, rdoby, prof, att, par, ctype) =
3    SUPERIOR_PAR_VALUE^9.30(1 THAT sup (2 sup ∈ doby and (cst)DescribesImSubOf^2.152(sup)2),
4                                  doby, rdoby, prof, att, par, ctype1)
5    ORELSE^9.1 STEP_G_PAR_VALUE^9.28(3 THAT sup (4 sup ∈ doby and (cst)DescribesImSubOf^2.152(sup)4),
6                                  doby, rdoby, prof, att, par, ctype3)0)
```

Semiformal Description 9.29

Function "Parameter value by step H"

If the attribute 'document application profile defaults' is specified in the document profile and this attribute specifies a value for the attribute *att* (4) and if this attribute specifies a value for the parameter *par* (5), then this is the value determined by step H. Otherwise the value is undefined by step H.

Definition 9.29

```
1  ∀ prof, att, par
2  (0 STEP_H_PAR_VALUE(prof, att, par) =
3    C (1 THAT a
4        (2 a Ê ^prof • 'document application profile defaults' and N a = att and
5          ∃ p (3 p Ê ^prof • 'document application profile defaults' • att and N p = par3)2)1)0)
```


Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.30

Function "Superior parameter value"

If the parameter value can be found in the default value list of the constituent *cst*, then this value is returned as the superior value (3). Otherwise, if the constituent refers to an object class (4) and the parameter value can be found in the default value list of the referenced object class, then this value is returned as the superior value (5). Otherwise, if the referenced object class refers to a resource document (6) and the parameter value can be found in the resource default value list of the referenced object class, then this value is returned as the superior value (7, 8). Otherwise the superior value is undefined.

Definition 9.30

```
1   ∀ cst, doby, rdoby, prof, att, ctype, par
2   (0 SUPERIOR_PAR_VALUE(cst, doby, rdoby, prof, att, ctype, par) =
3     DEFAULT_PAR_VALUE^9.31(cst, att, ctype, par, doby) ORELSE^9.1
4     (1 IF 'object class' ∈ NAMS^1.18(cst)
5       THEN (2 DEFAULT_PAR_VALUE^9.31(REF_OBJECT_CLASS^9.7(cst, doby), att, ctype, par, doby)
6       ORELSE^9.1 (3 IF 'resource' ∈ NAMS^1.18(REF_OBJECT_CLASS^9.7(cst, doby))
7                     THEN (4 RES_DEFAULT_PAR_VALUE^9.33(REF_OBJECT_CLASS^9.7(cst, doby),
8                                                        rdoby, prof, att, ctype, par)
9                           ORELSE^9.1 UNDEF4)
10                    ELSE UNDEF3)2)
11      ELSE UNDEF1)0)
```

Semiformal Description 9.31

Function "Default value of a parameter"

If a default value list is specified for the constituent *cst* (4), and default values for the particular type of constituent are specified (5, 6) and a default value for the particular parameter is specified (7, 8) then this value is the default value. Otherwise it is tested whether a style is referenced from the default value list which provides a value for the parameter (9). This function reflects the two cases that one (7) or more (8) sets of default values are specified, i.e., (a) one set is specified for composite object types (7); (b) one or more sets may be specified for basic object types (8).

Definition 9.31

```
1   ∀ cst, att, ctype, par, doby
2   (0 DEFAULT_PAR_VALUE(cst, att, ctype, par, doby) =
3     C (1 THAT a
4         (2 'default value lists' ∈ NAMS^1.18(cst) and
5           ∃ v
6           (3 v ∈̂ ¯cst • 'default value lists' • and v = ctype and
7             (4 (5 IsNom(C v) and a = v • att • par5) or
8               (6 IsCol(C v) and a = v • • att • par6)4)3)2)
9     ORELSE^9.1 DEFAULT_PAR_VALUE_BY_STYLE^9.32(cst, doby, att, ctype, par)1)0)
```

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.32

Function "Default parameter value by style"

If the attribute 'default value lists' is specified (3), contains an entry for the particular type of constituent (4, 5), there exists a reference to a layout style (7) and this layout style contains the attribute (8) and the parameter (9), then the value of this parameter is the default parameter value supplied by a style (10). Similarly, the value may also be supplied by a reference to a presentation style from within a default value list (11–18). This function reflects the two cases that one (11–14) or more (15–18) sets of default values are specified, i.e., (a) one set is specified for composite object types; (b) one or more sets may be specified for basic object types.

Definition 9.32

$$
\begin{array}{ll}
1 & \forall\, cst, doby, att, ctype, par \\
2 & ({}_{0}\,\mathrm{DEFAULT_PAR_VALUE_BY_STYLE}(cst, doby, att, ctype, par) = \\
3 & \quad \mathrm{IF}\ \text{'default value lists'} \in \mathrm{NAMS}^{1.18}(cst)\ \underline{and} \\
4 & \quad \exists\, v \\
5 & \quad ({}_{1}\, v \hat{\in} \hat{}\, cst \bullet \text{'default value lists'} \bullet\ \underline{and}\ v = ctype\ \underline{and} \\
6 & \qquad \exists\, a \\
7 & \qquad ({}_{2}({}_{3}\,\mathrm{IsNom(C}\ v)\ \underline{and}\ a = v \bullet \text{'layout style'}\ \underline{and} \\
8 & \qquad\quad att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{C}\ v, doby))\ \underline{and} \\
9 & \qquad\quad par \in \mathrm{NAMS}^{1.18}(\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{C}\ v, doby) \bullet par) \\
10 & \qquad\quad \mathrm{THEN\ C}\ \hat{}\,(\mathrm{REF_LAY_STYLE}^{9.4}(\mathrm{C}\ v, doby)) \bullet att \bullet par_{3})\ \underline{or} \\
11 & \qquad ({}_{4}\,\mathrm{IsNom(C}\ v)\ \underline{and}\ a = v \bullet \text{'presentation style'}\ \underline{and} \\
12 & \qquad\quad att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v, doby))\ \underline{and} \\
13 & \qquad\quad par \in \mathrm{NAMS}^{1.18}(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v, doby) \bullet att) \\
14 & \qquad\quad \mathrm{THEN\ C}\ \hat{}\,(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v, doby)) \bullet att \bullet par_{4})\ \underline{or} \\
15 & \qquad ({}_{5}\,\mathrm{IsCol(C}\ v)\ \underline{and}\ a = v \bullet\bullet \text{'presentation style'}\ \underline{and} \\
16 & \qquad\quad att \in \mathrm{NAMS}^{1.18}(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v\bullet, doby))\ \underline{and} \\
17 & \qquad\quad par \in \mathrm{NAMS}^{1.18}(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v\bullet, doby) \bullet att) \\
18 & \qquad\quad \mathrm{THEN\ C}\ \hat{}\,(\mathrm{REF_PRES_STYLE}^{9.5}(\mathrm{C}\ v\bullet, doby)) \bullet att \bullet par_{5})_{2})_{1})_{0})
\end{array}
$$

Semiformal Description 9.33

Function "Resource default value of a parameter"

If a default value is specified in a resource object class referenced by the object class *cst*, then this value is the resource default value. Otherwise the resource default value is undefined.

Definition 9.33

$$
\begin{array}{ll}
1 & \forall\, cst, rdoby, prof, att, ctype, par \\
2 & ({}_{0}\,\mathrm{RES_DEFAULT_PAR_VALUE}(cst, rdoby, prof, att, ctype, par) = \\
3 & \quad \mathrm{DEFAULT_PAR_VALUE}^{9.31}(\mathrm{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby, \\
4 & \qquad\qquad \mathrm{OBJECT_CLASSID_INRES}^{9.12}(prof, cst)), att, ctype, par, rdoby)_{0})
\end{array}
$$

Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.34

Function "Sub-parameter value by step C"

If an object class is referenced by a constituent *cst* and if the attribute *att* is specified for this object class (3) and if the parameter *par* and sub-parameter *subpar* are specified for the attribute (4, 5), then the value of the sub-parameter is the value determined by step C (6). Otherwise the value is undefined by step C.

Definition 9.34

1 $\forall\, cst, doby, att, par, subpar$
2 $({}_0\text{STEP_C_SUBPAR_VALUE}(cst, doby, att, par, subpar) =$
3 IF 'object class' $\in \text{NAMS}^{1.18}(cst)$ *and* $att \in \text{NAMS}^{1.18}(\text{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
4 $par \in \text{NAMS}^{1.18}(\text{C}\ \hat{}\text{REF_OBJECT_CLASS}^{9.7}(cst, doby) \bullet att)$ *and*
5 $subpar \in \text{NAMS}^{1.18}(\text{C}\ \hat{}\text{REF_OBJECT_CLASS}^{9.7}(cst, doby) \bullet att \bullet par)$
6 THEN C $\hat{}(\text{REF_OBJECT_CLASS}^{9.7}(cst, doby)) \bullet att \bullet par \bullet subpar$
7 ELSE UNDEF_0)

Semiformal Description 9.35

Function "Sub-parameter value by step E"

If an object class is referenced by a constituent *cst*, and a (resource) object class in a resource document is referenced by this object class and the attribute *att* is specified for this (resource) object class (3–6) and the parameter *par* and sub-parameter *subpar* are specified for this attribute (7–10) then the value of the sub-parameter is the value determined by step E (11, 12). Otherwise the value is undefined by step E.

Definition 9.35

1 $\forall\, cst, doby, rdoby, prof, att, par, subpar$
2 $({}_0\text{STEP_E_SUBPAR_VALUE}(cst, doby, rdoby, prof, att, par, subpar) =$
3 IF 'object class' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 'resource' $\in \text{NAMS}^{1.18}(\text{REF_OBJECT_CLASS}^{9.7}(cst, doby))$ *and*
5 $att \in \text{NAMS}^{1.18}(\text{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby,$
6 $\text{OBJECT_CLASSID_INRES}^{9.12}(prof, \text{REF_OBJECT_CLASS}^{9.7}(cst, doby)))))$ *and*
7 $par \in \text{NAMS}^{1.18}(\text{C}\ \hat{}\text{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby,$
8 $\text{OBJECT_CLASSID_INRES}^{9.12}(prof, \text{REF_OBJECT_CLASS}^{9.7}(cst, doby))) \bullet att)$ *and*
9 $subpar \in \text{NAMS}^{1.18}(\text{C}\ \hat{}\text{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby,$
10 $\text{OBJECT_CLASSID_INRES}^{9.12}(prof, \text{REF_OBJECT_CLASS}^{9.7}(cst, doby))) \bullet att \bullet par)$
11 THEN C $\hat{}(\text{RESOURCE_OBJECT_CLASS}^{9.11}(rdoby,$
12 $\text{OBJECT_CLASSID_INRES}^{9.12}(prof, \text{REF_OBJECT_CLASS}^{9.7}(cst, doby)))) \bullet att \bullet par \bullet subpar$
13 ELSE UNDEF_0)

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.36

Function "Sub-parameter value by step G"

If there exists a constituent *sup* in the document body to which the constituent *cst* is immediately subordinate (3) and if the sub-parameter value can be found (in a default value list) on the superior constituent *sup* (3, 4) then this sub-parameter value is the value determined by step G. Otherwise, if a superior constituent exists, the search for the sub-parameter value according to step G is continued (recursively) on the superior constituent *cst* (6-8). Otherwise, if no superior constituent exists or if the sub-parameter value cannot be found there, the value is undefined by step G.

Definition 9.36

1 $\forall\, cst, doby, rdoby, prof, att, par, subpar, ctype$
2 $(_0$STEP_G_SUBPAR_VALUE$(cst, doby, rdoby, prof, att, par, ctype) =$
3 SUPERIOR_PAR_VALUE$^{9.30}(_1$ THAT sup $(_2 sup \in doby$ *and*
4 (cst)DescribesImSubOf$^{2.152}(sup)_2)$,
5 $doby, rdoby, prof, att, par \bullet subpar, ctype_1)$
6 ORELSE$^{9.1}$ STEP_G_SUBPAR_VALUE$^{9.36}(_3$ THAT sup $(_4 sup \in doby$ *and*
7 (cst)DescribesImSubOf$^{2.152}(sup)_4)$,
8 $doby, rdoby, prof, att, par, subpar, ctype_3)_0)$

F.3 Functions for the values of defaultable attributes

This clause defines for each defaultable attribute the rules to determine its value.

NOTE — Default values for attributes are only derived for objects or content portions, not for object classes. Therefore, the argument *cst* in the functions defined below denotes always an object or content portion, never an object class.

F.3.1 Defaultable attributes of ISO 8613-2

Semiformal Description 9.37

Function "Value of the attribute 'application comments'"

The value of the attribute 'application comments' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step C (6). If this is not the case, test if the attribute value can be found by step E (7). If this is not the case, the value of the attribute 'application comments' is the empty catenation.

Definition 9.37

1 $\forall\, cst, doby, prof, rdoby$
2 $(_0$APPLICATION_COMMENTS_VALUE$(cst, doby, prof, rdoby) =$
3 IF IsAttributeSet$^{2.62}(cst)$ *and* 'application comments' $\in$ NAMS$^{1.18}(cst)$ *and*
4 *not* IsPlaceholder$^{1.19}$(C ^cst • 'application comments')
5 THEN C ^cst • 'application comments'
6 ELSE STEP_C_VALUE$^{9.6}(cst, doby,$ 'application comments') ORELSE$^{9.1}$
7 $(_1$STEP_E_VALUE$^{9.10}(cst, doby, rdoby, prof,$ 'application comments')
8 ORELSE$^{9.1}$ [$\rightarrow$] $_1)_0)$

Documents Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.38

Function "Value of the attribute 'balance'"

The value of the attribute 'balance' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step C (6). If this is not the case, test if the attribute value can be found by step E (7). If this is not the case, the value of the attribute 'balance' is 'null'.

Definition 9.38

```
1  ∀ cst, doby, prof, rdoby
2  (0 BALANCE_VALUE(cst, doby, prof, rdoby) =
3    IF IsAttributeSet^2.62(cst) and 'balance' ∈ NAMS^1.18(cst) and
4       not IsPlaceholder^1.19(C ^cst • 'balance')
5    THEN C ^cst • 'balance'
6    ELSE STEP_C_VALUE^9.6(cst, doby, 'balance') ORELSE^9.1
7    (1 STEP_E_VALUE^9.10(cst, doby, rdoby, prof, 'balance')
8      ORELSE^9.1 'null'1)0)
```

Semiformal Description 9.39

Function "Value of the attribute 'bindings'"

The value of the attribute 'bindings' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step C (6). If this is not the case, test if the attribute value can be found by step E (7). If this is not the case, the value of the attribute 'bindings' is the empty set.

Definition 9.39

```
1  ∀ cst, doby, prof, rdoby
2  (0 BINDINGS_VALUE(cst, doby, prof, rdoby) =
3    IF IsAttributeSet^2.62(cst) and 'bindings' ∈ NAMS^1.18(cst) and
4       not IsPlaccholdcr^1.19(C ^cst • 'bindings')
5    THEN C ^cst • 'bindings'
6    ELSE STEP_C_VALUE^9.6(cst, doby, 'bindings') ORELSE^9.1
7    (1 STEP_E_VALUE^9.10(cst, doby, rdoby, prof, 'bindings')
8      ORELSE^9.1 []1)0)
```

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.40

Function "Value of the attribute 'block alignment'"

The value of the attribute 'block alignment' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9). If this is not the case, test if the attribute value can be found by step H (10). If this is not the case, the value of the attribute 'block alignment' is 'right-hand aligned'.

Definition 9.40

$\forall\, cst, doby, prof, rdoby$
$(_0\text{BLOCK_ALIGNMENT_VALUE}(cst, doby, prof, rdoby) =$
IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'block alignment' $\in \text{NAMS}^{1.18}(cst)$ *and*
not $\text{IsPlaceholder}^{1.19}(\text{C } \hat{}cst \bullet \text{'block alignment'})$
THEN C $\hat{}cst \bullet$ 'block alignment'
ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'block alignment') $\text{ORELSE}^{9.1}$
$(_1\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'block alignment') $\text{ORELSE}^{9.1}$
$(_2\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'block alignment') $\text{ORELSE}^{9.1}$
$(_3\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'block alignment', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
$\text{ORELSE}^{9.1}$ $(_4\text{STEP_H_VALUE}^{9.22}(prof,$ 'block alignment')
$\text{ORELSE}^{9.1}$ 'right-hand aligned' $_4)_3)_2)_1)_0)$

Semiformal Description 9.41

Function "Value of the attribute 'border'"

The value of the attribute 'border' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BP (6). If this is not the case, test if the attribute value can be found by step C (7). If this is not the case, test if the attribute value can be found by step DP (8). If this is not the case, test if the attribute value can be found by step E (9). If this is not the case, test if the attribute value can be found by step FP (10). If this is not the case, test if the attribute value can be found by step G (11). If this is not the case, test if the attribute value can be found by step H (12). If this is not the case, the value comprises three parameters where the value of the parameters 'border line width' and 'border freespace width' is 0 and the value of the parameter 'border line type' is 'solid'.

Definition 9.41

$\forall\, cst, doby, prof, rdoby$
$(_0\text{BORDER_VALUE}(cst, doby, prof, rdoby) =$
IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'border' $\in \text{NAMS}^{1.18}(cst)$ *and*
not $\text{IsPlaceholder}^{1.19}(\text{C } \hat{}cst \bullet \text{'border'})$
THEN C $\hat{}cst \bullet$ 'border'
ELSE $\text{STEP_BP_VALUE}^{9.3}(cst, doby,$ 'border') $\text{ORELSE}^{9.1}$
$(_1\text{STEP_C_VALUE}^{9.6}(cst, doby,$ 'border') $\text{ORELSE}^{9.1}$
$(_2\text{STEP_DP_VALUE}^{9.8}(cst, doby,$ 'border') $\text{ORELSE}^{9.1}$
$(_3\text{STEP_E_VALUE}^{9.10}(cst, doby, rdoby, prof,$ 'border') $\text{ORELSE}^{9.1}$
$(_4\text{STEP_FP_VALUE}^{9.13}(cst, doby, rdoby, prof,$ 'border') $\text{ORELSE}^{9.1}$
$(_5\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'border', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
$\text{ORELSE}^{9.1}$ $(_6\text{STEP_H_VALUE}^{9.22}(prof,$ 'border')
$\text{ORELSE}^{9.1}$ ['border line width' : 0; 'border line type' : 'solid';
'border freespace width' : 0]$_6)_5)_4)_3)_2)_1)_0)$

Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.42

Function "Value of the attribute 'colour'"

The value of the attribute 'colour' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BP (6). If this is not the case, test if the attribute value can be found by step C (7). If this is not the case, test if the attribute value can be found by step DP (8). If this is not the case, test if the attribute value can be found by step E (9). If this is not the case, test if the attribute value can be found by step FP (10). If this is not the case, test if the attribute value can be found by step G (11). If this is not the case, test if the attribute value can be found by step H (12). If this is not the case, the value of the attribute 'colour' is 'colourless'.

Definition 9.42

```
1   ∀ cst, doby, prof, rdoby
2   (0 COLOUR_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'colour' ∈ NAMS^1.18(cst) and
4        not IsPlaceholder^1.19(C ^cst • 'colour')
5     THEN C ^cst • 'colour'
6     ELSE STEP_BP_VALUE^9.3(cst, doby, 'colour') ORELSE^9.1
7     (1 STEP_C_VALUE^9.6(cst, doby, 'colour') ORELSE^9.1
8       (2 STEP_DP_VALUE^9.8(cst, doby, 'colour') ORELSE^9.1
9         (3 STEP_E_VALUE^9.10(cst, doby, rdoby, prof, 'colour') ORELSE^9.1
10          (4 STEP_FP_VALUE^9.13(cst, doby, rdoby, prof, 'colour') ORELSE^9.1
11            (5 STEP_G_VALUE^9.17(cst, doby, rdoby, prof, 'colour', OBJECT_TYPE_VALUE^9.61(cst, doby))
12              ORELSE^9.1 (6 STEP_H_VALUE^9.22(prof, 'colour')
13                ORELSE^9.1 'colourless' 6) 5) 4) 3) 2) 1) 0)
```

Semiformal Description 9.43

Function "Value of the attribute 'concatenation'"

The value of the attribute 'concatenation' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9). If this is not the case, the value of the attribute 'concatenation' is 'non-concatenated'.

Definition 9.43

```
1   ∀ cst, doby, prof, rdoby
2   (0 CONCATENATION_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'concatenation' ∈ NAMS^1.18(cst) and
4        not IsPlaceholder^1.19(C ^cst • 'concatenation')
5     THEN C ^cst • 'concatenation'
6     ELSE STEP_BL_VALUE^9.2(cst, doby, 'concatenation') ORELSE^9.1
7     (1 STEP_DL_VALUE^9.9(cst, doby, 'concatenation') ORELSE^9.1
8       (2 STEP_FL_VALUE^9.15(cst, doby, rdoby, prof, 'concatenation') ORELSE^9.1
9         (3 STEP_G_VALUE^9.17(cst, doby, rdoby, prof, 'concatenation', OBJECT_TYPE_VALUE^9.61(cst, doby))
10          ORELSE^9.1 'non-concatenated' 3) 2) 1) 0)
```

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.44

Function "Value of the attribute 'content architecture class'"

The value of the attribute 'content architecture class' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step C (6). If this is not the case, test if the attribute value can be found by step E (7). If this is not the case, test if the attribute value can be found by step G (8, 9). If this is not the case, test if the attribute value can be found by step H (10). If this is not the case, the value of the attribute 'content architecture class' is '2 8 2 6 0'.

Definition 9.44

```
1   ∀ cst, doby, prof, rdoby
2   (0 CONTENT_ARCHITECTURE_CLASS_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'content architecture class' ∈ NAMS^1.18(cst) and
4       not IsPlaceholder^1.19(C ^cst • 'content architecture class')
5     THEN C ^cst • 'content architecture class'
6     ELSE STEP_C_VALUE^9.6(cst, doby, 'content architecture class') ORELSE^9.1
7     (1 STEP_E_VALUE^9.10(cst, doby, rdoby, prof, 'content architecture class') ORELSE^9.1
8       (2 STEP_G_VALUE^9.17(cst, doby, rdoby, prof, 'content architecture class',
9                            OBJECT_TYPE_VALUE^9.61(cst, doby)) ORELSE^9.1
10        (3 STEP_H_VALUE^9.22(prof, 'content architecture class')
11          ORELSE^9.1 '2 8 2 6 0'3)2)1)0)
```

Semiformal Description 9.45

Function "Value of the attribute 'dimensions'"

The value of the attribute 'dimensions' comprises the parameter 'horizontal dimension' whose value is a horizontal dimension value, and the parameter 'vertical dimension' whose value is a vertical dimension value.

Definition 9.45

```
1   ∀ cst, doby, prof, rdoby
2   (0 DIMENSIONS_VALUE(cst, doby, prof, rdoby) =
3     ['horizontal dimension' : HORIZONTAL_DIMENSION_VALUE^9.46(cst, doby, prof, rdoby);
4       'vertical dimension' : VERTICAL_DIMENSION_VALUE^9.47(cst, doby, prof, rdoby)]0)
```


AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.46

Function "Value of the parameter 'horizontal dimension'"

The value of the parameter 'horizontal dimension' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step C (7). If this is not the case, test if the parameter value can be found by step E (8). If this is not the case, test if the parameter value can be found by step G (9). If this is not the case and the constituent is a page, test if the parameter value can be found by step H (11, 12). If step H provides no value or if the constituent is not a page, the value is the parameter 'horizontal dimension' whose value is the sub-parameter 'fixed dimension' with a fixed dimension value (14).

Definition 9.46

```
1   ∀ cst, doby, prof, rdoby
2   (0 HORIZONTAL_DIMENSION_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'dimensions' ∈ NAMS^1.18(cst) and
4        'horizontal dimension' ∈ NAMS^1.18(C ^cst • 'dimensions') and
5        not IsPlaceholder^1.19(C ^cst • 'dimensions' • 'horizontal dimension')
6     THEN C ^cst • 'dimensions' • 'horizontal dimension'
7     ELSE STEP_C_PAR_VALUE^9.24(cst, doby, 'dimensions', 'horizontal dimension') ORELSE^9.1
8     (1 STEP_E_PAR_VALUE^9.26(cst, doby, rdoby, prof, 'dimensions', 'horizontal dimension') ORELSE^9.1
9       (2 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'dimensions', 'horizontal dimension',
10                                OBJECT_TYPE_VALUE^9.61(cst, doby)) ORELSE^9.1
11        (3 (4 IF OBJECT_TYPE_VALUE^9.61(cst, doby) = 'composite or basic page'
12            THEN STEP_H_PAR_VALUE^9.29(prof, 'dimensions', 'horizontal dimension')
13                 ORELSE^9.1 UNDEF4)
14          ORELSE^9.1 ['fixed dimension' : HORIZONTAL_FIXED_DIMENSION_VALUE^9.48(cst)] 3)2)1)0)
```

Semiformal Description 9.47

Function "Value of the parameter 'vertical dimension'"

The value of the parameter 'vertical dimension' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step C (7). If this is not the case, test if the parameter value can be found by step E (8). If this is not the case, test if the parameter value can be found by step G (9). If this is not the case and the constituent is a page, test if the parameter value can be found by step H (11, 12). If step H provides no value or if the constituent is not a page, the value is the parameter 'vertical dimension' whose value is the sub-parameter 'fixed dimension' with a fixed dimension value (14).

Definition 9.47

```
1   ∀ cst, doby, prof, rdoby
2   (0 VERTICAL_DIMENSION_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'dimensions' ∈ NAMS^1.18(cst) and
4        'vertical dimension' ∈ NAMS^1.18(C ^cst • 'dimensions') and
5        not IsPlaceholder^1.19(C ^cst • 'dimensions' • 'vertical dimension')
6     THEN C ^cst • 'dimensions' • 'vertical dimension'
7     ELSE STEP_C_PAR_VALUE^9.24(cst, doby, 'dimensions', 'vertical dimension') ORELSE^9.1
8     (1 STEP_E_PAR_VALUE^9.26(cst, doby, rdoby, prof, 'dimensions', 'vertical dimension') ORELSE^9.1
9       (2 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'dimensions', 'vertical dimension',
10                                OBJECT_TYPE_VALUE^9.61(cst, doby)) ORELSE^9.1
11        (3 (4 IF OBJECT_TYPE_VALUE^9.61(cst, doby) = 'composite or basic page'
12            THEN STEP_H_PAR_VALUE^9.29(prof, 'dimensions', 'vertical dimension')
13                 ORELSE^9.1 UNDEF4)
14          ORELSE^9.1 ['fixed dimension' : VERTICAL_FIXED_DIMENSION_VALUE^9.50(cst)] 3)2)1)0)
```

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.48

Function "horizontal fixed dimension value"

The horizontal fixed dimension value is 9240 (the horizontal dimension of the assured reproduction area of an ISO A4 page) for a page and an available horizontal dimension value otherwise.

Definition 9.48

1 $\forall\, cst$
2 $({}_0\,\text{HORIZONTAL_FIXED_DIMENSION_VALUE}(cst) =$
3 IF $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby) =$ 'composite or basic page'
4 THEN 9240
5 ELSE $\text{AVAILABLE_HORIZONTAL_DIMENSION_VALUE}^{9.49}(cst)_0)$

Semiformal Description 9.49

Function "available horizontal dimension value"

The available horizontal dimension is determined during the layout process. It is considered an atom-valued function here.

Definition 9.49

1 $\forall\, cst$
2 $({}_0\,\text{IsAtom}(\text{AVAILABLE_HORIZONTAL_DIMENSION_VALUE}(cst))_0)$

Semiformal Description 9.50

Function "vertical fixed dimension value"

The vertical fixed dimension value is 13200 (the vertical dimension of the reassured reproduction area of an ISO A4 page) for a page and an available vertical dimension value otherwise.

Definition 9.50

1 $\forall\, cst$
2 $({}_0\,\text{VERTICAL_FIXED_DIMENSION_VALUE}(cst) =$
3 IF $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby) =$ 'composite or basic page'
4 THEN 13200
5 ELSE $\text{AVAILABLE_VERTICAL_DIMENSION_VALUE}^{9.51}(cst)_0)$

Semiformal Description 9.51

Function "available vertical dimension value"

The available vertical dimension is determined during the layout process. It is considered an atom-valued function here.

Definition 9.51

1 $\forall\, cst$
2 $({}_0\,\text{IsAtom}(\text{AVAILABLE_VERTICAL_DIMENSION_VALUE}(cst))_0)$

Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.52

Function "Value of the attribute 'fill order'"

The value of the attribute 'fill order' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9). If this is not the case, the value of the attribute 'fill order' is 'normal order'.

Definition 9.52

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_0\text{FILL_ORDER_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'fill order' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}(\text{C } \hat{}cst \bullet \text{'fill order'})$
5 THEN C $\hat{}cst \bullet$ 'fill order'
6 ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'fill order') ORELSE$^{9.1}$
7 $({}_1\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'fill order') ORELSE$^{9.1}$
8 $({}_2\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'fill order') ORELSE$^{9.1}$
9 $({}_3\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'fill order', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
10 ORELSE$^{9.1}$ 'normal order' ${}_3)_2)_1)_0)$

Semiformal Description 9.53

Function "Value of the attribute 'indivisibility'"

The value of the attribute 'indivisibility' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9). If this is not the case, the value of the attribute 'indivisibility' is 'null'.

Definition 9.53

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_0\text{INDIVISIBILITY_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'indivisibility' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}(\text{C } \hat{}cst \bullet \text{'indivisibility'})$
5 THEN C $\hat{}cst \bullet$ 'indivisibility'
6 ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'indivisibility') ORELSE$^{9.1}$
7 $({}_1\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'indivisibility') ORELSE$^{9.1}$
8 $({}_2\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'indivisibility') ORELSE$^{9.1}$
9 $({}_3\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'indivisibility', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
10 ORELSE$^{9.1}$ 'null' ${}_3)_2)_1)_0)$

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.54

Function "Value of the attribute 'layout category'"

The value of the attribute 'layout category' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9). If this is not the case, the value of the attribute 'layout category' is 'null'.

Definition 9.54

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_0\text{LAYOUT_CATEGORY_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'layout category' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}$(C ^$cst$. 'layout category')
5 THEN C ^cst . 'layout category'
6 ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'layout category') $\text{ORELSE}^{9.1}$
7 $({}_1\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'layout category') $\text{ORELSE}^{9.1}$
8 $({}_2\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'layout category') $\text{ORELSE}^{9.1}$
9 $({}_3\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'layout category', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
10 $\text{ORELSE}^{9.1}$ 'null' $_3)_2)_1)_0)$

Semiformal Description 9.55

Function "Value of the attribute 'layout object class'"

The value of the attribute 'layout object class' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9, 10). If this is not the case, the value of the attribute 'layout object class' is 'null'.

Definition 9.55

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_0\text{LAYOUT_OBJECT_CLASS_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'layout object class' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}$(C ^$cst$. 'layout object class')
5 THEN C ^cst . 'layout object class'
6 ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'layout object class') $\text{ORELSE}^{9.1}$
7 $({}_1\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'layout object class') $\text{ORELSE}^{9.1}$
8 $({}_2\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'layout object class') $\text{ORELSE}^{9.1}$
9 $({}_3\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'layout object class',
10 $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
11 $\text{ORELSE}^{9.1}$ 'null' $_3)_2)_1)_0)$

Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.56

Function "Value of the attribute 'layout path'"

The value of the attribute 'layout path' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step C (6). If this is not the case, test if the attribute value can be found by step E (7). If this is not the case, test if the attribute value can be found by step G (8, 9). If this is not the case, test if the attribute value can be found by step H (10). If this is not the case, the value of the attribute 'layout path' is '270°'.

Definition 9.56

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_{0}\text{LAYOUT_PATH_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'layout path' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaccholder}^{1.19}$(C ^$cst$ • 'layout path')
5 THEN C ^cst • 'layout path'
6 ELSE $\text{STEP_C_VALUE}^{9.6}(cst, doby,$ 'layout path') $\text{ORELSE}^{9.1}$
7 $({}_{1}\text{STEP_E_VALUE}^{9.10}(cst, doby, rdoby, prof,$ 'layout path') $\text{ORELSE}^{9.1}$
8 $({}_{2}\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'layout path', $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
9 $\text{ORELSE}^{9.1}$ $({}_{3}\text{STEP_H_VALUE}^{9.22}(prof,$ 'layout path')
10 $\text{ORELSE}^{9.1}$ '270°' $)_{3})_{2})_{1})_{0})$

Semiformal Description 9.57

Function "Value of the attribute 'medium type'"

The value of the attribute 'medium type' comprises the parameter 'nominal page size' whose value is a nominal page size value, and the parameter 'side of sheet' whose value is a side of sheet value.

Definition 9.57

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_{0}\text{MEDIUM_TYPE_VALUE}(cst, doby, prof, rdoby) =$
3 ['nominal page size' : $\text{NOMINAL_PAGE_SIZE_VALUE}^{9.58}(cst, doby, prof, rdoby)$;
4 'side of sheet' : $\text{SIDE_OF_SHEET_VALUE}^{9.59}(cst, doby, prof, rdoby)$] $_{0})$

Documents Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.58

Function "Value of the parameter 'nominal page size'"

The value of the parameter 'nominal page size' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step C (7). If this is not the case, test if the parameter value can be found by step E (8). If this is not the case, test if the parameter value can be found by step G (9). If this is not the case, test if the parameter value can be found by step H (11). If this is not the case, the value of the parameter 'nominal page size' is the pair 9920 and 14030.

Definition 9.58

```
1   ∀ cst, doby, prof, rdoby
2   (0 NOMINAL_PAGE_SIZE_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'medium type' ∈ NAMS^1.18(cst) and
4         'nominal page size' ∈ NAMS^1.18(C ^cst • 'medium type') and
5         not IsPlaceholder^1.19(C ^cst • 'medium type' • 'nominal page size')
6     THEN C ^cst • 'medium type' • 'nominal page size'
7     ELSE STEP_C_PAR_VALUE^9.24(cst, doby, 'medium type', 'nominal page size') ORELSE^9.1
8     (1 STEP_E_PAR_VALUE^9.26(cst, doby, rdoby, prof, 'medium type', 'nominal page size') ORELSE^9.1
9       (2 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'medium type', 'nominal page size',
10                                 OBJECT_TYPE_VALUE^9.61(cst, doby)) ORELSE^9.1
11        (3 STEP_H_PAR_VALUE^9.29(prof, 'medium type', 'nominal page size')
12          ORELSE^9.1 [ → 9920 → 14030 → ]3)2)1)0)
```

Semiformal Description 9.59

Function "Value of the parameter 'side of sheet'"

The value of the parameter 'side of sheet' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step C (7). If this is not the case, test if the parameter value can be found by step E (8). If this is not the case, test if the parameter value can be found by step G (9). If this is not the case, test if the parameter value can be found by step H (11). If this is not the case, the value of the parameter 'side of sheet' is 'unspecified'.

Definition 9.59

```
1   ∀ cst, doby, prof, rdoby
2   (0 SIDE_OF_SHEET_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'medium type' ∈ NAMS^1.18(cst) and
4         'side of sheet' ∈ NAMS^1.18(C ^cst • 'medium type') and
5         not IsPlaceholder^1.19(C ^cst • 'medium type' • 'side of sheet')
6     THEN C ^cst • 'medium type' • 'side of sheet'
7     ELSE STEP_C_PAR_VALUE^9.24(cst, doby, 'medium type', 'side of sheet') ORELSE^9.1
8     (1 STEP_E_PAR_VALUE^9.26(cst, doby, rdoby, prof, 'medium type', 'side of sheet') ORELSE^9.1
9       (2 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'medium type', 'side of sheet',
10                                 OBJECT_TYPE_VALUE^9.61(cst, doby)) ORELSE^9.1
11        (3 STEP_H_PAR_VALUE^9.29(prof, 'medium type', 'side of sheet')
12          ORELSE^9.1 'unspecified'3)2)1)0)
```


Documents
Document Sharing Hub
Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.60

Function "Value of the attribute 'new layout object'"

The value of the attribute 'new layout object' may be specified directly on a particular constituent (3-5). If this is not the case, test if the attribute value can be found by step BL (6). If this is not the case, test if the attribute value can be found by step DL (7). If this is not the case, test if the attribute value can be found by step FL (8). If this is not the case, test if the attribute value can be found by step G (9, 10). If this is not the case, the value of the attribute 'new layout object' is 'null'.

Definition 9.60

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_{0}\text{NEW_LAYOUT_OBJECT_VALUE}(cst, doby, prof, rdoby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'new layout object' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}$(C ^$cst$ • 'new layout object')
5 THEN C ^cst • 'new layout object'
6 ELSE $\text{STEP_BL_VALUE}^{9.2}(cst, doby,$ 'new layout object') $\text{ORELSE}^{9.1}$
7 $({}_{1}\text{STEP_DL_VALUE}^{9.9}(cst, doby,$ 'new layout object') $\text{ORELSE}^{9.1}$
8 $({}_{2}\text{STEP_FL_VALUE}^{9.15}(cst, doby, rdoby, prof,$ 'new layout object') $\text{ORELSE}^{9.1}$
9 $({}_{3}\text{STEP_G_VALUE}^{9.17}(cst, doby, rdoby, prof,$ 'new layout object',
10 $\text{OBJECT_TYPE_VALUE}^{9.61}(cst, doby))$
11 $\text{ORELSE}^{9.1}$ 'null'$_{3})_{2})_{1})_{0})$

Semiformal Description 9.61

Function "Value of the attribute 'object type'"

The value of the attribute 'object type' may be specified directly on a particular constituent (3-5). If this is not the case, the attribute value is determined by step C (6).

NOTE — An undefined result of step C is not possible for a valid ODA document.

Definition 9.61

1 $\forall\, cst, doby$
2 $({}_{0}\text{OBJECT_TYPE_VALUE}(cst, doby) =$
3 IF $\text{IsAttributeSet}^{2.62}(cst)$ *and* 'object type' $\in \text{NAMS}^{1.18}(cst)$ *and*
4 *not* $\text{IsPlaceholder}^{1.19}$(C ^$cst$ • 'object type')
5 THEN C ^cst • 'object type'
6 ELSE $\text{STEP_C_VALUE}^{9.6}(cst, doby,$ 'object type'$)_{0})$

Semiformal Description 9.62

Function "Value of the attribute 'offset'"

The value of the attribute 'offset' comprises the parameter 'leading offset' whose value is a leading offset value, the parameter 'trailing offset' whose value is a trailing offset value, the parameter 'left-hand offset' whose value is a left-hand offset value, and the parameter 'right-hand offset' whose value is a right-hand offset value.

Definition 9.62

1 $\forall\, cst, doby, prof, rdoby$
2 $({}_{0}\text{OFFSET_VALUE}(cst, doby, prof, rdoby) =$
3 ['leading offset' : $\text{LEADING_OFFSET_VALUE}^{9.63}(cst, doby, prof, rdoby)$;
4 'trailing offset' : $\text{TRAILING_OFFSET_VALUE}^{9.64}(cst, doby, prof, rdoby)$;
5 'left-hand offset' : $\text{LEFTHAND_OFFSET_VALUE}^{9.65}(cst, doby, prof, rdoby)$;
6 'right-hand offset' : $\text{RIGHTHAND_OFFSET_VALUE}^{9.66}(cst, doby, prof, rdoby)]_{0})$

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993

Semiformal Description 9.63

Function "Value of the parameter 'leading offset'"

The value of the parameter 'leading offset' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step BL (7). If this is not the case, test if the parameter value can be found by step DL (8). If this is not the case, test if the parameter value can be found by step FL (9). If this is not the case, test if the parameter value can be found by step G (10, 11). If this is not the case, the value of the parameter 'leading offset' is 0.

Definition 9.63

```
1   ∀ cst, doby, prof, rdoby
2   (0 LEADING_OFFSET_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'offset' ∈ NAMS^1.18(cst) and
4        'leading offset' ∈ NAMS^1.18(C ^cst • 'offset') and
5        not IsPlaceholder^1.19(C ^cst • 'offset' • 'leading offset')
6     THEN C ^cst • 'offset' • 'leading offset'
7     ELSE STEP_BL_PAR_VALUE^9.23(cst, doby, 'offset', 'leading offset') ORELSE^9.1
8     (1 STEP_DL_PAR_VALUE^9.25(cst, doby, 'offset', 'leading offset') ORELSE^9.1
9       (2 STEP_FL_PAR_VALUE^9.27(cst, doby, rdoby, prof, 'offset', 'leading offset') ORELSE^9.1
10        (3 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'offset', 'leading offset',
11                                 OBJECT_TYPE_VALUE^9.61(cst, doby))
12          ORELSE^9.1 0 3) 2) 1) 0)
```

Semiformal Description 9.64

Function "Value of the parameter 'trailing offset'"

The value of the parameter 'trailing offset' may be specified directly on a particular constituent (3-6). If this is not the case, test if the parameter value can be found by step BL (7). If this is not the case, test if the parameter value can be found by step DL (8). If this is not the case, test if the parameter value can be found by step FL (9). If this is not the case, test if the parameter value can be found by step G (10, 11). If this is not the case, the value of the parameter 'trailing offset' is 0.

Definition 9.64

```
1   ∀ cst, doby, prof, rdoby
2   (0 TRAILING_OFFSET_VALUE(cst, doby, prof, rdoby) =
3     IF IsAttributeSet^2.62(cst) and 'offset' ∈ NAMS^1.18(cst) and
4        'trailing offset' ∈ NAMS^1.18(C ^cst • 'offset') and
5        not IsPlaceholder^1.19(C ^cst • 'offset' • 'trailing offset')
6     THEN C ^cst • 'offset' • 'trailing offset'
7     ELSE STEP_BL_PAR_VALUE^9.23(cst, doby, 'offset', 'trailing offset') ORELSE^9.1
8     (1 STEP_DL_PAR_VALUE^9.25(cst, doby, 'offset', 'trailing offset') ORELSE^9.1
9       (2 STEP_FL_PAR_VALUE^9.27(cst, doby, rdoby, prof, 'offset', 'trailing offset') ORELSE^9.1
10        (3 STEP_G_PAR_VALUE^9.28(cst, doby, rdoby, prof, 'offset', 'trailing offset',
11                                 OBJECT_TYPE_VALUE^9.61(cst, doby))
12          ORELSE^9.1 0 3) 2) 1) 0)
```


AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8613-10_1991_Amd 5_1993