INTERNATIONAL STANDARD ISO/IEC 13240:2001
TECHNICAL CORRIGENDUM 1

Published 2003-03-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Document description and processing languages — Interchange Standard for Multimedia Interactive Documents (ISMID)

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Description des documents et langages de traitement — Norme d'échange pour documents interactifs multimédia (ISMID)

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 13240:2001 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

The following modifications correct the ISMID architecture defined by ISO/IEC 13240:2001.

Subclause 9.4 Property Values, Example in Note 8

Replace (attribute-string "LEFT" with (attribute-string "Left"
Replace (element-with-id ButtonY"))) 100)) with (element-with-id "ButtonY"))) 100))
Replace ID="buttonX" with ID="ButtonX"

ICS 35.240.30 **Ref. No. ISO/IEC 13240:2001/Cor.1:2003(E)**

© ISO/IEC 2003 – All rights reserved

Published in Switzerland

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

Maquette

INTERNATIONAL STANDARD ISO/IEC 13240:2001(E)

Information technology
Document description and processing languages
Interchange Standard for Multimedia Interactive Documents (ISMID)

Technologies de l'information Description des documents et langages de traitement
Norme d'échange pour documents interactifs multimédia (ISMID)

ISO/IEC 13240:2001(E)

First edition 2001-12-15

© ISO/IEC 2001

ICS 35.240.30

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 either ISO at the address below or ISO's member body in the country of the requester.

Price based on 67 pages

ISO copyright office
Case postale 56 · CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Published in Switzerland

--`,,`-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Information technology Document description and processing languages Interchange Standard for Multimedia Interactive Documents (ISMID)

Contents

- Foreword
- Introduction
- 1 Scope
  - 1.1 Definition of scope
    - 1.1.1 The ISMID Architecture
    - 1.1.2 The Example ISMID application
  - 1.2 Field of application
- 2 Conformance
- 3 Normative references
- 4 Terms and definitions
- 5 Symbols and abbreviations
- 6 Notations
  - 6.1 RCS name, full name, description, and clause
  - 6.2 Constraints
  - 6.3 Note
- 7 The ISMID HyTime support declarations
- 8 The ISMID expression language
- 9 The ISMID architecture
  - 9.1 Parameter entity declarations
    - 9.1.1 Create object
    - 9.1.2 Control flow
    - 9.1.3 Variable Types
    - 9.1.4 Modify an object
    - 9.1.5 Responses
  - 9.2 Multimedia Interactive Document
  - 9.3 Content Reference
  - 9.4 Property Values
  - 9.5 Interface Objects
    - 9.5.1 Common object attributes
    - 9.5.2 Create a Container Object
    - 9.5.3 Modify a Container Object
    - 9.5.4 Create a Content Object
    - 9.5.5 Modify a Content Object
    - 9.5.6 Create a Control Object
    - 9.5.7 Modify a Control Object
    - 9.5.8 Create an External Object

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

- 9.5.9 Modify an External Object
- 9.5.10 Destroy an Object
- 9.6 Label
- 9.7 Stimulus
- 9.8 Response chains
- 9.9 Argument Declaration
- 9.10 Execute a response chain
- 9.11 Control Flow
  - 9.11.1 If statement
  - 9.11.2 Elseif statement
  - 9.11.3 Else statement
  - 9.11.4 While statement
  - 9.11.5 Switch, Case and Default Statements
  - 9.11.6 Break statement
  - 9.11.7 Expression
  - 9.11.8 Argument
  - 9.11.9 Variable declaration
  - 9.11.10 Assignment statement
  - 9.11.11 Variable name
  - 9.11.12 Execute an external process
  - 9.11.13 MID end
- 10 ISMID Application Definition Document (IADD)
  - 10.1 Architectural Form Usage
  - 10.2 Requirements
    - 10.2.1 Encompassing Architecture (Document Type Definition)
    - 10.2.2 Data Content Notations
    - 10.2.3 Entities
    - 10.2.4 Stimuli
    - 10.2.5 Properties
    - 10.2.6 Objects
- Annex A (normative) ISMID Meta−DTD
- Annex B (informative) Sample ISMID Application Definition Document

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

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

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.

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 13240 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

Annex A forms a normative part of this International Standard. Annex B is for information only.

Introduction

Interactive documents combine data content with instructions that define the behavior of the interactive document; that is, how and when the content is represented on a computer display or how and when information is passed to an external system. Because a reader's interactions with a document might be important to its presentation, a standard way to encode behavior instructions associated with specific content is essential if the document is to be transportable. This standard focuses on the representation of the behavioral aspects of interactive documents and hyperdocuments. The language defined in this standard facilitates the interchange of interactive documents among heterogeneous development and delivery systems by defining the behavioral aspects of interactive documents independent of proprietary scripting or processing languages.

Interactive Documents

In the context of interactive documents, interactive means that a software application is responsible for dynamically rendering content in response to stimuli from users or other sources. The response is often to tailor the content and presentation of subsequent information. Therefore, to create an interactive document, developers must consider a philosophy different from that used to create documents and hyperdocuments that will be presented in a static manner; they must program behavior into the document. The inclusion of the behavior component is one facet of interactive documents that can cause incompatibility between interactive document development and delivery systems.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

It is important to recognize the distinction between "document" in the SGML sense (a static source of data) and "document" in the rendition sense (a particular view or presentation of the source data). It is also important to recognize that any static source of data (i.e., any document in the SGML sense) can be made interactive through the application of behavioral style. Throughout this standard the term interactive document is used when referring to the presentation of the source data (i.e., when referring to a document that has been dynamically rendered).

Document Standards

Standards currently exist for describing the structure, content, and static presentation of documents and hyperdocuments. The Interchange Standard for Multimedia Interactive Documents (ISMID) adds the missing component − a standard for representing the behaviorial component of documents in an unambiguous way for the purpose of interchange among heterogeneous interactive document development and delivery systems.

While there are established standards for implementing Graphical User Interface (GUI) objects in a programming context, there is a need for a standard that will allow developers of interactive documents to express in a device independent manner how they intended to make use of those interface objects. ISMID provides an architecture for defining how the interface objects communicate with the structured content covered by existing standards.

Because the presentation of interface objects in interactive documents is often integral to the message being communicated, ISMID allows the developer to specify presentation properties of the interface objects. Types of presentation properties could include things such as font, color, and location of the object. It is up to the interactive document developer to decide whether ISMID is used to apply these properties or whether a style specification language like DSSSL is used.

The obvious use of the architecture described in this standard is to provide a way to interchange interactive behavior with respect to window−based GUI objects. However, the ISMID architecture is general and is not limited to any specific GUI environment − in fact, it need not define a GUI at all. The architecture defines a language for defining any stimuli that a system could receive, including stimuli that are not received through a GUI. The responses also may be actions other than rendering information to the user through a GUI. An object derived from the *CreateControl* element form need not be a GUI widget; it could be a means of communicating with an external system.

Note 1. For example, the object derived from the *CreateControl* element form may be an "EnvironmentMonitor" control that monitors the temperature in a climate sensitive environment. In this case the stimulus received by the MID may be that the temperature has exceeded 40 degrees. The response to this stimulus may be that the MID presents a warning to the user as well as sends, via a control that communicates with the equipment, "ShutDown" instructions to the equipment that should not be run when the temperature exceeds 40 degrees.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

1 Scope

1.1 Definition of scope

This International Standard, known as the Interchange Standard for Multimedia Interactive Documents or ISMID, facilitates the interchange of Multimedia Interactive Documents (MIDs) among heterogeneous interactive document development and delivery systems by providing the architecture from which common interchange languages can be created. ISMID is a client architecture of International Standard ISO/IEC 10744:1997, Information technology –– Hypermedia/Time–based Structuring Language (HyTime) and is an SGML application conforming to International Standard ISO 8879 –– Standard Generalized Markup Language.

ISMID also specifies use of the DSSSL expression language described in clause 8 of ISO/IEC 10179:1996, Information technology –– Processing languages –– Document Style Semantics and Specification Language (DSSSL).

In general, the rendering of a Multimedia Interactive Document (MID) requires three major components:

- Content: the information that the document author intended to communicate to the user of the MID;
- Behavior: the instructions that govern what, when, where, and how content is presented, and how the MID responds to stimuli from various sources; and
- Software: the application that interprets behavior, retrieves the content, and presents the information to the user.

Various vendors, standards bodies, and other groups define the notations (such as SGML for text, CGM for graphics, MPEG for video, and WAV for audio) with which the information in the content component is represented. It is up to software vendors to determine the most effective way to implement the software component. Therefore, the ISMID standard addresses only the behavior component.

1.1.1 The ISMID Architecture

The ISMID standard defines an architecture that enables authors and developers to describe the behavior component of MIDs in a device–independent manner. This architecture is known as the ISMID architecture (Clause 9 The ISMID architecture). The ISMID architecture defines architectural forms for the behavior component.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

The ISMID architecture includes:

- Interface objects;
    - Container objects that are used to group other interface objects (e.g., Windows, Frames, Panels);
    - Content objects that are used to render specific information content (e.g., Video, Image, Audio, Animation, Text);
    - Control objects through which interaction with the MID takes place (e.g., Button, Scroll Bar, List Box, Check Box). Control objects enable users to make decisions and facilitate navigation through a MID instance;
    - External objects are used to allow the MID to communicate with external systems such as the computer operating environment or a diagnostic system.
- Stimuli, which are actions received via the interface objects; and
- Responses, which are reactions performed by the MID in response to a given stimulus.

Although interface objects are listed as a major part of the ISMID architecture, ISMID does not define architectural forms that represent these objects directly. Instead these objects are specified using architectural forms to describe the creation, modification, and destruction of the objects. The architectural forms for creation, modification, and destruction of objects are classified as responses because a MID would create, modify, or destroy an object only in response to a stimulus from a user or some other source.

The stimulus and response element forms in the architecture provide the templates for specific ISMID applications. These architectural forms within the architecture are general enough to allow for any new stimulus and response elements that may be required in the future.

It is important to note that content objects are not the actual information being conveyed to the user, but are used to render and manipulate the information. The information referenced by content objects can be reused, and can reside in various external notations that are independent of the content object.

1.1.2 The Example ISMID application

An example ISMID application is included in this standard (Annex B: Sample ISMID Application Definition Document (Informative)). The example defines a set of elements, derived from the architecture, for stimuli and responses. The semantics of the forms in the example reflect the objects, stimuli, and responses used in state−of−the−art interactive document development and delivery systems. For example, one of the element types defined in the example ISMID application is for displaying images. Because the image object is used to present information content to the user, the element is derived from the content object forms in the architecture.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The example elements implement all of the attributes of the architecture forms and add attributes that define the presentation properties of these objects as they exist in current use. For example, a button object in the example ISMID application has attributes that define properties for caption, visible, enabled, and foreground color.

It is envisioned that various industries and organizations will define ISMID applications in which they remove from and/or add to the specific objects, stimuli, and responses defined in the example application in order to meet the industry's requirements. These industry−defined applications shall also define the semantics of the objects, stimuli, and responses. Conforming ISMID applications must be defined according to the ISMID Application Definition Document (IADD) (Clause 10 ISMID Application Definition Document (IADD)). The ISMID application in non−normative Annex B: Sample ISMID Application Definition Document (Informative) is an example of an IADD.

The example application DTD in Annex B: Sample ISMID Application Definition Document (Informative) includes notation forms for the most common information types in current use. These notation declarations do not imply that a conforming ISMID system must support these notations. The notations supported by an ISMID system shall be specified within the ISMID application. For example, the Department of Defense may specify that SGML must be supported for text while a commercial entity with existing software investments may prefer that RTF be supported for text. Which specific objects, stimuli, responses and notations must be supported for a particular industry or organization can be specified formally in the IADD.

1.2 Field of application

The field of application of ISMID is Electronic Performance Support Systems (EPSS), computer−based interactive document systems that provide users access to just−in−time information and on−the−job training. Two common types of EPSS are Interactive Electronic Technical Manuals (IETMs) and Interactive Courseware (ICW).

The use of EPSSs is increasing within many industries as the quantity of information that workers must have access to increases. Because organizations are increasing their investments in these systems, it is important that they have a way to protect their investments by representing all aspects of the EPSS in a device independent manner.

2 Conformance

Because ISMID is primarily a language that facilitates the interchange of interactive documents, it is not a conformance requirement that ISMID systems operate directly on the ISMID document in its native notation. Developers will probably chose to translate the ISMID representation to their own proprietary internal representation optimized for their application programs to execute.

A conforming ISMID system shall:

--`,,`-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

- be able to import any ISMID document that conforms to any application that the system supports.
- be able to export ISMID documents that conform to any applications that the system supports.
- as a minimum, support the HyTime modules and options specified in the ISMID HyTime support declaration (Clause 7 The ISMID HyTime support declarations). A conforming system need not be limited to these modules and options. If additional modules and/or options are supported, they shall be specified in the system's HyTime support declaration.
- support the HyFunk expression language defined in clause 6.8.6 of ISO/IEC 10744:1997 Information technology −− Hypermedia/Time−based Structuring Language (HyTime)
- support all of the interface objects, stimuli, responses, and notations specified in a valid ISMID application which the system supports.

A conforming ISMID application shall:

- be defined by an IADD as described in Clause 10 ISMID Application Definition Document (IADD).
- if the application makes use of HyTime modules and/or options in addition to those specified in the ISMID HyTime support declaration (Clause 7 The ISMID HyTime support declarations), specify the additional modules and/or options in the application's HyTime support declaration.

A conforming ISMID document shall:

- be a conforming SGML document that also conforms to the encompassing architecture described in the IADD.
- contain all features that are required by the application.
- not exceed the feature set described in the IADD for the application to which the document conforms.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this International Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

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

ISO/IEC 10179:1996, *Information technology −− Processing languages −− Document Style Semantics and Specification Language (DSSSL)*

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4 Terms and definitions

For the purposes of this International Standard, the following terms and definitions apply.

4.1

application

A DTD conforming to the ISMID architecture, and serving as the basis for a class of interactive documents e.g., for an industry.

4.2

behavior component

The instructions that govern how an interactive document responds to stimuli from various sources.

4.3

container object

A type of interface object used to group other interface objects.

4.4

control object

A type of interface object through which interaction with an interactive document takes place.

4.5

document

As defined in ISO 8879:1986, a document is a collection of information that is identified as a unit and that is intended for human perception.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.6

external objects

A type of interface object that is used to enable an interactive document to communicate with external systems.

4.7

hyperdocument

As defined in ISO/IEC 10744:1997, a hyperdocument is two or more documents or other information objects that are connected to one another by one or more hyperlinks.

4.8

hyperlink

As defined in ISO/IEC 10744:1997, a hyperlink is an information structure that represents a relationship among two or more objects.

4.9

interface objects

Components of an interactive document that are the vehicle through which stimuli and responses occur.

4.10

interactive document

A particular presentation of static SGML source data, dynamically rendered for viewing by a particular user. It typically contains instructions that define the rules of its own presentation.

4.11

interactive document delivery system

The run−time software used to render an interactive document.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.12

interactive document development system

A software system used to author interactive documents.

4.13

interactive presentation

A form of presentation of a document in which the order of navigation through the document content is modified during presentation in response to conditions.

Note 2. The modifications are typically in response to external stimuli, such as user interactions, which is why these presentations are termed "interactive". However, the dynamic modification can also be in response to presentation algorithms contained in the document or within the system controlling the presentation.

Note 3. A document need not be an interactive document in order for it to be the subject of an interactive presentation.

4.14

content object

A type of interface object used to render information content.

4.15

response

A reaction performed by the interactive document, triggered by a stimulus.

4.16

response chain

A predetermined sequence of responses that may be initiated by a single stimulus.

4.17

static document

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

A document that is intended for static presentation.

4.18

static presentation

A form of presentation of a document in which the order of navigation through the document content is essentially the same as the order in which the content appears in the computer representation of the document.

Note 4. In a static presentation, any dynamic modification is typically limited to the navigational functionality offered by a printed volume, such as page–turning and cross–reference following.

4.19

stimulus

An action, received by an interface object, that triggers reactions performed by the interactive document.

4.20

stimulus–driven

A system in which action is initiated when an interface object receives a stimulus.

5 Symbols and abbreviations

AFDR	Architectural Form Definition Requirements
DSSSL	Document Style Semantics and Specification Language
EPSS	Electronic Performance Support System
HyTime	Hypermedia/Time–based Structuring Language
ICW	Interactive Courseware
IADD	ISMID Application Definition Document
IETM	Interactive Electronic Technical Manual
ISMID	Interchange Standard for Multimedia Interactive Documents
MID	Multimedia Interactive Document
RCS	Reference Concrete Syntax

6 Notations

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

ISMID is an enabling document architecture whose definition (this International Standard) conforms to the Architectural Form Definition Requirements in Annex A.3 of ISO/IEC 10744:1997 the *SGML Architectural Form Definition Requirements* (AFDR). The formal definition of the ISMID notation is expressed as a meta−DTD. The specification of the ISMID architecture is accomplished by a combination of narrative text and formal definitions.

Any references in this document to industry and proprietary standards, products, user groups, and publications are not normative, and do not imply endorsement by ISO, IEC, or their national member bodies or affiliates. Any brand names or trademarks mentioned are the property of their respective owners.

The formal definitions are expressed in SGML.

The formal SGML definitions are part of the text of this International Standard and are protected by copyright. In order to facilitate conformance to the ISMID architecture, the formal SGML definitions may be copied as specified in the following copyright notice: *Copyright (C) 2001 International Organization for Standardization. Permission to copy in any form is granted for use with conforming ISMID systems and applications as defined in ISO/IEC 13240, provided this notice is included in all copies.* The permission to copy does not apply to any other material in this International Standard.

Note 5. This document uses editorial conventions mandated by the ISO with which the reader should be familiar in order to understand the implications of certain words.

The text describing each construct emphasizes semantics, while the formal SGML definition provides the rigorous syntactic definitions underlying the text descriptions.

Note 6. For this reason, it is recommended that the reader refer to the SGML definitions while reading the textual descriptions. Although the SGML definition always follows the related text, the user may find it helpful to read the SGML first in some cases.

When a construct is first introduced, it is described in the text. If the construct occurs in the formal SGML specification, both the formal SGML name and a full name in English are presented, as follows:

- The element form **full construct name** (*SGMLname*) ...
- The attribute **full construct name** (*SGMLname*) ...

The declarations include comments, called "conventional comments", that follow conventions established in the HyTime standard to specify syntactic and semantic constraints and other information that is known to an architecture engine such as an ISMID engine. The conventional comments do not extend SGML in any way. They are used in the architecture definitions only, as a notation for the documentation of the architecture. They need not be included in application DTDs and, if they are included, the SGML parser will treat them as it would any other comment.

6.1 RCS name, full name, description, and clause

Every form name is followed by comments giving the full name, a description of the form, and the number of the clause in this International Standard in which the form is defined.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Individual attributes have a full name and description comment.

6.2 Constraints

Comments labeled "Constraints" define additional semantic or syntactic constraints on the constructs they follow. Constraint comments that follow the name of a form define constraints on the use of that form in general. Constraint comments that follow a component of a declaration (for example, the default value prescription of an attribute declaration), define the specific constraints on that component.

6.3 Note

Note comments provide additional information not provided by the other comment types and are informational rather than constraining.

7 The ISMID HyTime support declarations

As a HyTime client architecture, ISMID uses the HyTime support declarations to specify the HyTime modules and options that must be supported by conforming ISMID systems.

```
<?IS10744 ArcBase HyTime>



<!NOTATION

  HyTimePUBLIC

   "ISO/IEC 10744:1997//NOTATION AFDR ARCBASEHypermedia/Time-based

    Structuring Language (HyTime)//EN"

>



<!ATTLIST #NOTATION HyTime

   ArcFormA  NAME              HyTime

   ArcNamrA  NAME              HyNames

   ArcSuprA  NAME              sHyTime

   ArcIgnDA  NAME              HyIgnD

   ArcDocF   NAME              #FIXED HyDoc

   ArcDTD    CDATA             "HyTime"

   ArcQuant  CDATA             #FIXED "NAMELEN 9"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ArcDataF  NAME               #FIXED HyBridN

   ArcBridF  NAME               #FIXED HyBrid

   ArcAuto  (ArcAuto|nArcAuto)  nArcAuto

   ArcOptSA  NAMES              "GenArc base links"

   GenArc    CDATA              "ireftype"

   base      CDATA              "HyFunk"

   locs      CDATA              "referatt refloc reftype mixedloc

                                 nameloc nmsploc treeloc"

   links     CDATA              ""

   sched     CDATA              ""

   rend      CDATA              ""

   anysgml   (anysgml|nanysgml) anysgml

   exrefs    (exrefs|nexrefs)   exrefs

   refmodel  (SGMLmdl|nSGMLmdl) nSGMLmdl

   hyqcnt    NUMBER             32

   manyanch  NUMBER             #IMPLIED

   manyaxes  NUMBER             "4"

>



<!NOTATION

  AFDRMeta

    PUBLIC "ISO/IEC 10744//NOTATION AFDR Meta-DTD Notation//EN">



<!ENTITY

   HyTime

    PUBLIC

    "ISO/IEC 10744:1997//DTD AFDR Meta-DTD Hypermedia/Time-based Structuring

    Language (HyTime)//EN"

    CDATA AFDRMeta>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

8 The ISMID expression language

A conforming ISMID system shall, at a minimum, support the **HyTime Marker Function Language** (*HyFunk*) expression language. This expression language is adequate for documents that require only integer arithmetic operations. *HyFunk* is a subset of the DSSSL expression Language. The following notation shall be used to specify use of HyFunk as the ISMID expression language.

```
<!notation

   HyFunk

    PUBLIC

     "ISO/IEC 10744:1997//NOTATION HyTime Marker Function Language//EN"

>
```

If a richer set of expressions are required, the ISMID system shall support the expression language described in clause 8 of the DSSSL standard. The following notation shall be used to specify support of the DSSSL expression language within ISMID documents.

```
<!NOTATION

   DSSSLExp

    PUBLIC

     "ISO/IEC 10179:1996//NOTATION DSSSL Expression Language Notation//EN"

>
```

ISMID also allows the use of other expression languages within a MID. Support for the expression language is declared by including a notation declaration for the language in the IADD. Which expression language is used for a particular expression is indicated by the *Language* attribute on the *Expression* element.

9 The ISMID architecture

A MID is defined as a stimulus−driven application. This means that interface objects within the MID wait for a stimulus initiated by a user, an external system, or another interface object before responding.

For each stimulus, the author/developer of a MID instance can define a series of responses that, at runtime, determine how the interface object will respond to the stimulus. Specific stimulus types are defined in the IADD

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

for each ISMID application. Each stimulus in the ISMID application implements the attributes of the corresponding stimulus element form in the ISMID architecture, from which the element is derived. For each stimulus, the author may specify a series of specific actions (e.g., creating, destroying, or modifying objects and variables) that will occur in the MID. For example, when the user clicks a "Show Video" button (i.e., when the interface object labeled "Show Video" receives a click stimulus from the user), the developer could program the "Show Video" button to respond by:

- Creating a Video Content Object;
- Creating a Stop Playing control object (button);
- Opening a video source in the Video Content Object;
- Playing the video; and
- Creating a series of responses within the click stimulus of the Stop Playing button that will terminate the playing of the video, or associating the click stimulus of the Stop Playing button with a previously defined response chain that will terminate the playing of the video.

Interface objects are the targets of user−, object− or externally−generated stimuli presented to the MID. The element type declaration for each element that creates or modifies an interface object can contain within its content model a series of stimulus elements. For example, a button control object could contain an "OnClick" stimulus and an "OnMouseOver" stimulus. Each stimulus element can contain a chain of one or more responses to be executed when the object receives the stimulus; this is referred to as "hard−coding" the response chain. Alternatively, the stimulus could reference a previously defined response chain; this is referred to as "soft−coding" the response chain.

Within a response chain, objects are created, modified, and destroyed. Control flow statements can also be included within response chains so that responses can be based on conditions or states within the MID.

9.1 Parameter entity declarations

9.1.1 Create object

Create object elements shall be used to create Container objects, Content objects, Control objects, and External objects.

```
<!entity %

   CreateObject  -- Create an object --

    "CreateContainer  | CreateContentObject  | CreateControl | CreateExternalObject">
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

9.1.2 Control flow

Control flow shall allow construction of If, While, and Switch constructs.

```
<!entity %

   ControlFlow -- Control flow --

    "If | While | Switch">
```

9.1.3 Variable Types

```
<!entity %

   VariableTypes -- Variable types --

    "Boolean | i1 | i2 | i4 | i8 | ui1 | ui2 | ui4 | ui8 | r4 | r8 | String | Char | Variant">
```

9.1.4 Modify an object

Modify object elements shall be used to modify any properties of existing Container objects, Content objects, Control objects, and External objects.

```
<!entity %

   ModifyObject -- Modify an object --

    "ModifyContainer | ModifyContentObject | ModifyControl | ModifyExternalObject">
```

9.1.5 Responses

Responses shall include *CreateObject* , *ModifyObject* , and control *ControlFlow* entities defined above. Responses shall also allow assignment of values to variables, destruction of objects, or execution of any predefined response chain or external process.

```
<!entity %

   Responses  -- Responses --

    "%CreateObject; | %ControlFlow; | %ModifyObject; | DestroyObject |

     ExecuteResponseChain | Execute">
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

9.2 Multimedia Interactive Document

The **Multimedia Interactive Document** (*MID*) element form is the document element form of an ISMID document. When the *MID* is initiated, it receives an *OnSystemStart* stimulus. In response to this stimulus, the MID executes the Responses in the *SystemChain* . Zero or more other response chains can be defined within the *MID* element but they are not executed until they are referenced through a stimulus within the MID. These response chains exist as a pool of functions that can be called as required throughout the MID.

The *ID* attribute is a unique identifier for each element of this type. The outermost container object's parent attribute must exhibit the value of this attribute.

```
<!element
  MID              -- Multimedia Interactive Document document element --
                   -- Clause 9.2 --
  - O
  (ContentReference*, SystemChain, ResponseChain*)
>

<!attlist
  MID
   HyTime           -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid" -- HyTime bridge element form --
   ID               -- unique identifier --
    ID
    #REQUIRED
>
```

9.3 Content Reference

The **content reference** (*ContentReference*) element form is used to address the content to be rendered in a *Content Object* .

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The **referenced content does not exist** (*ContentNotExist*) attribute determines how the system shall handle the case where content is not found at runtime. The document author determines whether the "not found" condition should result in an error or should be ignored by the processing system. The default is to ignore the condition and continue processing. If the *ContentNotExist* attribute exhibits an *error* value, a conforming ISMID system shall report an error if the referenced content is not found.

```
<!element
  ContentReference  -- Content reference --
                    -- Clause 9.2 --
  - O
  (nmlist)
>


<!attlist
  ContentReference
   HyTime             -- HyTime architectural form name --
    CDATA
    #FIXED "nameloc"  -- HyTime nameloc element form --
   ContentNotExist    -- Referenced content does not exist --
                      -- specifies whether or not the author wants the system to
                         report content reference errors or to ignore them --
    (error | ignore)
    "ignore"
>


<!element
  nmlist      -- Name list --
              -- Clause 9.2 --
  - O
  (#PCDATA)
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!attlist

  nmlist

   HyTime           -- HyTime architectural form name --

    CDATA

    #FIXED "nmlist" -- HyTime nmlist element form --

   nametype         -- HyTime nametype attribute form --

    (element | entity)

    "entity"

   docorsub         -- HyTime docorsub attribute form --

    #ENTITY

    #IMPLIED

>
```

9.4 Property Values

Interactive documents often set the value of an object property relative to another object's property values. A HyTime **value reference** (*valueref*) attribute shall be used to obtain the effective value of an attribute from the properties of another object.

Note 7. The "effective value" of an attribute is the value that an application or architectural processor needs, which is not necessarily limited to the types of data or objects that can be expressed through SGML's attribute value syntax.

The value exhibited by the **value reference** attribute shall be fixed in the ISMID encompassing architecture (Document Type Definition).

The referential attribute must refer to the ID of an *Expression* element that determines the effective value of the property attribute.

Note 8. For example, the position of a control element may be determined relative to the position of other control elements. The property, defined by the value of an attribute, would be resolved in the expression, and the results of the expression would be used as the effective value of the attribute. In this example, the value of the Left attriubute on the element with ID « ButtonX » is equal to the vlaue of the Left attribute on the element with ID « ButtonY » plus 100 pixels.

```
<CreateButton
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    ID="ButtonY"
    Left="200"
>
<Expression ExpressionLanguage="DSSSL" id="RelPosition">
 (number->string
  (+ (string->number
   (attribute-string "LEFT"
    (element-with-id  ButtonY"))) 100))
</Expression>
<CreateButton
  ID="buttonX"
  valueref = "Left LeftRef"
  Left = "500"
  LeftRef="RelPosition"
>
```

9.5 Interface Objects

There are four classes of interface objects: containers, content objects, controls, and external objects. Within a MID, these objects exist in a hierarchy. This means that each object is contained within some larger parent object. The only exception to this rule is the outermost container object, the parent of which must be the *MID* element. Each object specifies its parent explicitly rather than relying on the SGML hierarchy to indicate parent–child relationships. This approach gives authors direct control of the order in which objects are created and destroyed.

The following sections describe the architecture with respect to these objects.

9.5.1 Common object attributes

Certain attributes are common to all create and modify object elements. They are defined and described here so that the definitions and descriptions need not be repeated in each section.

A HyTime *valueref* attribute shall be used to obtain the effective value of an attribute from the properties of another object (see Clause 9.4 Property Values). The value exhibited by the *valueref* attribute shall be fixed in the ISMID encompassing architecture (Document Type Definition).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The **referenced object does not exist** (*ObjectNotExist)* attribute determines how the system shall handle the case where a referenced object is not found at runtime. The *ObjectNotExist* attribute applies when either a Parent container is referenced or when the property of an object is referenced to provide the effective value for an attribute. The document author determines whether the "not found" condition should result in an *error* or should be ignored by the processing system. The default is to ignore the condition and continue processing. If the *ObjectNotExist* attribute exhibits an *error* value, a conforming ISMID system shall report an error if the referenced object does not exist.

Note 9. This attribute is provided to handle situations where interactive traversal leads to unexpected paths through the document, rather than to report logic errors. The error condition is most likely to occur when an object is either referenced before it is created or after it has been destroyed.

When an object expected to provide the effective value for an attribute does not exist and *ObjectNotExist* is set to *ignore* , the system shall use the fixed attribute value of required attributes. When a parent container object does not exist, the content object shall not be created and *ObjectNotExist* shall determine whether the condition is reported or ignored.

```
<!entity %

   CommonObjectAttributes "

    HyTime

     NAME

     #FIXED `HyBrid'

    valueref

     CDATA

     #IMPLIED

    ObjectNotExist

    (error | ignore)

    `ignore'

">
```

9.5.2 Create a Container Object

The **create a container object** (*CreateContainer*) element form shall be used to create objects whose primary purpose is grouping other interface objects within the interactive document. Each container object may define stimuli to which the system will respond. If no stimuli are included for the container, the container cannot receive stimuli from users or other sources.

Note 10. Examples of container objects within a windows operating environment are windows, panels, and frames.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Variable declarations (*VariableDeclaration*) and (*Stimulus*) are associated with the *CreateContainer* element through their occurrence within the *CreateContainer* element type's content model.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute shall contain the name of the *Parent* attribute followed by a model group (constrained to be an "or group") that expresses the MID element and which *CreateContainer* element types are valid as parents of this object. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The *Parent* attribute must reference element types derived from either the *MID* or the *CreateContainer* forms. The value of the *Parent* attribute for the outermost container object must be the value of the ID attribute of the element derived from the *MID* architectural form.

The *ID* attribute is a unique identifier for each element of this type. Each object for which this object is the parent must exhibit the value of this attribute in its parent attribute.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  CreateContainer  -- Create a container object --
                   -- Clause 9.5.2 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>


<!attlist
  CreateContainer
   %CommonObjectAttributes; -- Common object attributes --
   ireftype                 -- Used to indicate which elements are valid as parents of
                               this container object --
    CDATA
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    #REQUIRED

   Parent                    -- Used to identify the parent of this container object.

                                Constraint:  must reference element types derived from either

                                the MID or the CreateContainer element forms --

    IDREF

    #REQUIRED

   ID                        -- Unique identifier --

    ID

    #REQUIRED

>
```

9.5.3 Modify a Container Object

The **modify a container object** (*ModifyContainer*) element form shall be used to redefine properties of an existing container object, to modify values of variables that were declared for the container object, and/or to specify new or modify existing responses to stimuli for this container object.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The HyTime *ireftype* attribute contains the name of the *Object* attribute followed by a model group (constrained to be an "or group") of GIs that express which *CreateContainer* element types that are valid objects to be modified. It may also contain the name of the *Parent* attribute followed by a model group (contrained to be an "or group") that expresses which *CreateContainer* element types are valid parent objects.

The *Object* attribute identifies the existing container object to be modified, by referring to its unique identifier. The *Object* attribute must reference element types derived from the *CreateContainer* form. The **unique identifier** (ID) shall not be modified. All other attributes of the object may be modified.

The *Parent* attribute may be modified, but must continue to reference element types derived from either the *MID* or the *CreateContainer* forms.

Note 11. Modifying the *Parent* of a container object effectively moves the container to the new *Parent* container, positioning the container relative to its new *Parent* .

New variable assignments, and stimuli can be associated with the *ModifyContainer* element through their occurrence within the *ModifyContainer* 's content model. All variable declarations must occur when the object is created; new variables shall not be declared when modifying a container object.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

All modifications of an object that use an expression to refer to a property attribute value, refer to the current state of the object, not to the original creation state.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  ModifyContainer -- Modify a container object --
                  -- Clause 9.5.3 --
  - O
  (Assign*, Expression*, Stimulus*)
>

<!attlist
  ModifyContainer
   %CommonObjectAttributes;
   ireftype           -- used to indicate which object types can be modified
                         by this element.  It may also indicate which object types
                         are valid as parents for this object. --
    CDATA
    #REQUIRED
   Object             -- Identifies the container object to be modified
                         Constraint: the value of this attribute must be the unique
                         identifier of an existing container object. --
    IDREF
    #REQUIRED
   Parent             -- Used to identify the parent of this content object.
                         Constraint:  must reference element types derived from the MID or
                         CreateContainer element forms. --
    IDREF
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    #IMPLIED

>
```

9.5.4 Create a Content Object

The **create a content object** (*CreateContentObject*) element form shall be used to create a host object for information content. If no stimuli are included for the content object, the content object cannot receive stimuli from users or other sources.

Note 12. Examples of content objects are text, image, video, audio, and animation objects.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. This architectural declaration indicates that the *SemanticContent* attribute must reference a *ContentReference* element. In addition, the *ireftype* shall contain the name of the *Parent* attribute followed by a model group (constrained to be an "or group") that expresses which *CreateContainer* element types are valid as parents of this object. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The *Parent* attribute must reference element types derived from the *CreateContainer* form. The document author is responsible for ensuring that the parent container is created before any content object that references it. The system developer is responsible for checking for existence of parent objects, and responding as specified in the *ObjectNotExist* attribute.

The *SemanticContent* attribute is used to locate information that will be rendered in the content object. The content is located by reference through the HyTime *nameloc* element called *ContentReference* .

The *ID* attribute is a unique identifier for each element of this type. Each object for which this object is the parent must exhibit the value of this attribute in its parent attribute.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element

  CreateContentObject  -- Create a content object --

                       -- Clause 9.5.4 --

  - O

  (Label*, VariableDeclaration*, Expression*, Stimulus*)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
>


<!attlist
  CreateContentObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which element can be referenced by the
                               SemanticContent attribute and which elements are valid as
                               parents of this content object --
    CDATA
    "SemanticContent ContentReference"
   Parent                   -- Used to identify the parent of this content object.
                               Constraint:  must reference element types derived from the
                               CreateContainer element form. --
    IDREF
    #REQUIRED
   SemanticContent           -- used to reference the information to be rendered in
                                this content object --
    IDREF
    #IMPLIED
   ID                        -- unique identifier --
    ID
    #REQUIRED
>
```

9.5.5 Modify a Content Object

The **modify a content object** (*ModifyContentObject*) element form shall be used to redefine properties of an existing content object, to modify values of variables that were declared for the content object, and/or to specify new or modify existing responses to stimuli for this content object.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. This architectural declaration indicates that the *SemanticContent* attribute must reference a *ContentReference* element. The *ireftype* attribute value shall also contain the name of the *Object*

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

attribute followed by a model group (contrained to be an "or group") that expresses which *CreateContentObject* element types are valid objects to be modified. It may also contain the name of the *Parent* attribute followed by a model group (contrained to be an "or group") that expresses which *CreateContainer* element types are valid parent objects.

The *Object* attribute identifies the existing content object to be modified, by referring to its unique identifier. The *Object* attribute must reference element types derived from the *CreateContentObject* form. The **unique identifier** (*ID*) shall not be modified. All other attributes of the object may be modified.

The *Parent* attribute may be optionally modified, but must continue to reference element types derived from the *CreateContainer* form.

Note 13. Modifying the *Parent* of a container object effectively moves the container to the new *Parent* container, positioning the container relative to its new *Parent* .

The *SemanticContent* attribute uses HyTime location addressing to reference the source data content that will populate the content object.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  ModifyContentObject  -- Modify a content object --
                       -- Clause 9.5.5 --
  - O
  (Assign*, Expression*, Stimulus*)
>

<!attlist
  ModifyContentObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which element can be referenced by the
                               SemanticContent attribute; which object types can be modified
                               by this element; and which elements are valid as parents of
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                              this content object. --
    CDATA
    "SemanticContent ContentReference"
   Object                     -- Identifies the content object to be modified
                              Constraint: the value of this attribute must be the unique
                              identifier of an existing content object. --
    IDREF
    #REQUIRED
   Parent                     -- Used to identify the parent of this content object.
                              Constraint:  must reference element types derived from the
                              CreateContainer element form. --
    IDREF
    #IMPLIED
   SemanticContent            -- used to reference the information to be rendered in
                              this content object --
    IDREF
    #IMPLIED
>
```

9.5.6 Create a Control Object

The **create a control object** (*CreateControl*) element form shall be used to define selection, navigation, and input/output controls for interaction with users. If no stimuli are included for the control object, the control object cannot receive stimuli from users or other sources.

Note 14. Examples of control objects in a windows environment are buttons, check boxes, and scroll bars.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *Parent* attribute followed by a list of GIs that express which *CreateContainer* or *CreateContentObject* element types are valid as parents of this object. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The *Parent* attribute must reference element types derived from the *CreateContainer* or *CreateContentObject* element forms.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The *ID* attribute is a unique identifier for each element of this type.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  CreateControl              -- Create a control object --
                             -- Clause 9.5.6 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>

<!attlist
  CreateControl
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                  -- Used to indicate which elements are valid as parents of
                                this control object --
    CDATA
    #REQUIRED
   Parent                    -- Used to identify the parent of this control object.
                                Constraint:  must reference element types derived from the
                                CreateContainer or CreateContentObject element forms. --
    IDREF
    #REQUIRED
   ID                        -- unique identifier --
    ID
    #REQUIRED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

9.5.7 Modify a Control Object

The **modify a control object** (*ModifyControl*) element form shall be used to redefine properties of an existing control object; to modify values of variables that were declared for the control object; and/or to specify new or modify existing responses to stimuli for this control object.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *Object* attribute followed by a model group (contrained to be an "or group") that express which *CreateControl* element types are valid objects to be modified. It may also contain the name of the *Parent* attribute followed by a list of GIs that express which *CreateContainer* or *CreateContentObject* element types are valid parent objects.

The *Object* attribute identifies the existing container object to be modified, by referring to its unique identifier. The *Object* attribute must reference element types derived from the *CreateControl* form. The unique identifier (ID) of the *CreateControl* object shall not be modified. All other attributes of the object may be modified.

The *Parent* attribute may be optionally modified, but must continue to reference element types derived from the *CreateContainer* or *CreateContentObject* forms.

Note 15. Modifying the *Parent* of a container object effectively moves the container to the new *Parent* container, positioning the container relative to its new *Parent* .

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  ModifyControl             -- Modify a control object --
                            -- Clause 9.5.7 --
  - O
  (Assign*, Expression*, Stimulus*)
>

<!attlist
  ModifyControl
   %CommonObjectAttributes; -- see clause 9.5.1 --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ireftype                -- used to indicate which object types can be modified
                              by this element.  It may also indicate which object types are valid a
                              parents for this object. --
    CDATA
    #IMPLIED
   Object                  -- Identifies the control object to be modified
                              Constraint: the value of this attribute must be the unique
                              identifier of an existing control object. --
    IDREF
    #REQUIRED
   Parent                  -- Used to identify the parent of this control object.
                              Constraint:  must reference element types derived from the
                              CreateContainer or CreateContentObject element forms. --
    IDREF
    #IMPLIED
>
```

9.5.8 Create an External Object

The **create an external object** (*CreateExternalObject*) element form shall be used to create objects representing things outside of the ISMID application. If no stimuli are included for the container, the container cannot receive stimuli from users or other sources.

Note 16. Examples of external objects are the computer operating environment, an external database, or a diagnostic system connected to the ISMID system.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *Parent* attribute followed by a model group (contrained to be an "or group") that express which *CreateContainer* element types are valid as parents of this object. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The *Parent* attribute must reference element types derived from the *CreateContainer* form.

The *ID* attribute is a unique identifier for each element of this type.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  CreateExternalObject      -- Create an external object --
                            -- Clause 9.5.8 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>

<!attlist
  CreateExternalObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which elements are valid as parents of
                               this external object --
    CDATA
    #IMPLIED
   Parent                   -- Used to identify the parent of this external object.
                               Constraint:  must reference element types derived from the
                               CreateContainer element form. --
    IDREF
    #REQUIRED
   ID                       -- unique identifier --
    ID
    #REQUIRED
>
```

9.5.9 Modify an External Object

The **modify an external object** (*ModifyExternalObject*) element form shall be used to modify objects created using the *CreateExternalObject* architectural form.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *Object* attribute followed by a model group (contrained to be an "or group") that express which *CreateExternalObject* element types are valid objects to be modified.It may also contain the name of the *Parent* attribute followed by a list of GIs that express which *CreateContainer* element types are valid parent objects.

The *Object* attribute identifies the existing container object to be modified, by referring to its unique identifier. The *Object* attribute must reference element types derived from the *CreateExternalObject* form. The unique identifier (ID) of *CreateExternalObject* shall not be modified. All other attributes of the object may be modified.

The *Parent* attribute may be optionally modified, but must continue to reference element types derived from the *CreateContainer* form.

Note 17. Modifying the *Parent* of a container object effectively moves the container to the new *Parent* container, positioning the container relative to its new *Parent* .

The *Object* attribute must reference element types derived from the CreateContainer form.

See Clause 9.5.1 Common object attributes for information on the **common object attributes** (*CommonObjectAttributes*).

```
<!element
  ModifyExternalObject      -- Modify an external object --
                            -- Clause 9.5.9 --
  - O
  (Assign*, Expression*, Stimulus*)
>

<!attlist
  ModifyExternalObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- used to indicate which object types can be modified
                               by this element.  It may also indicate which object types
                               are valid as parents for this object. --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    CDATA
    #IMPLIED
   Object                    -- Identifies the external object to be modified
                                Constraint: the value of this attribute must be the unique
                                identifier of an existing external object. --
    IDREF
    #REQUIRED
   Parent                    -- Used to identify the parent of this external object.
                                Constraint:  must reference element types derived from the
                                CreateContainer element form. --
    IDREF
    #IMPLIED
>
```

9.5.10 Destroy an Object

The **destroy an object** (*Destroy*) element form shall be used to delete any *Object* and all of its children. Any variables that are scoped at the level of the *Object* will also be deleted.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *Object* attribute followed by a model group (contrained to be an "or group") that express which create object element types are valid objects to be destroyed.

The *Object* attribute must reference element types derived from the *CreateContainer* , *CreateContentObject* , *CreateControl* , or *CreateExternalObject* forms.

Note 18. If the object doesn't exist, the system shall ignore the destroy response.

```
<!element
  DestroyObject      -- Destroy an object --
                     -- Clause 9.5.10 --
  - O
  EMPTY
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
>


<!attlist
  DestroyObject
   HyTime            -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"  -- HyTime bridge element form --
   ireftype          -- used to indicate which create object element types are valid
                        objects to be destroyed.  --
    CDATA
    #IMPLIED
   Object            -- Identifies the object to be destroyed
                        Constraint: the value of this attribute must be the unique
                        identifier of an existing object. --
    IDREF
    #REQUIRED
>
```

9.6 Label

The **object label** (*Label*) element form shall be used to provide a descriptive name for the object in which the element appears. This label may be used to identify the object to the user in the documentation or in the application.

The *xml:lang* attribute is used to specify the language in which the *label* is written.

```
<!element
  Label             -- Object label --
                    -- Clause 9.6 --
  - O
  (#PCDATA)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
>



<!attlist

  Label

   HyTime           -- HyTime architectural form name --

    NAME

    #FIXED "HyBrid" -- HyTime bridge element form --

   xml:lang         -- used to specify the language used in the content of this element --

    CDATA

    #REQUIRED

>
```

9.7 Stimulus

The **stimulus** (*Stimulus*) element form shall contain a chain of responses and/or a reference to a previously defined response chain. If the *Stimulus* element both contains and references response chains, the ISMID system shall first execute the contained response chain and then execute the referenced response chain.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of the *ResponseChainRef* attribute followed by the name of the *ResponseChain* element type. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The **response chain reference** (*ResponseChainRef*) attribute form is used to reference an existing response chain to be executed when this stimulus is received by the system.

```
<!element

  Stimulus           -- Stimulus --

                     -- Clause 9.7 --

  - O

  ((%Responses;)*, End?)

>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!attlist
  Stimulus
   HyTime            -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"  -- HyTime bridge element form --
   ireftype          -- used to indicate which element type is the response chain
                        element type  --
    CDATA
    "ResponseChainRef ResponseChain"
   ResponseChainRef  -- used to reference an existing response chain --
    IDREF
    #IMPLIED
>
```

9.8 Response chains

Response chains contain a series of response elements that are executed when the response chain is called.

The **response chain** (*ResponseChain*) element form is called with an *ExecuteResponseChain* element form. The call identifies which chain to execute by using the *ResponseChain* 's unique identifier. Parameters are passed to the response chain via the *ArgumentDeclaration* element form.

The **system response chain** (*SystemChain*) element form must exist and is the chain that is initiated by the implied *OnSystemStart* stimulus resulting from system startup.

```
<!element
  (SystemChain | ResponseChain) -- System response chain --
                                -- Response chain --
                                -- Clause 9.8 --
  - O
  (Label*, ArgumentDeclaration*, (%Responses;)*, End?)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
>


<!attlist
  ResponseChain
   HyTime                       -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"             -- HyTime bridge element form --
   ID                           -- unique identifier --
    ID
    #IMPLIED
>
```

9.9 Argument Declaration

The **argument declaration** (*ArgumentDeclaration*) element form shall be used to declare arguments that can be passed to response chains using the *ExecuteResponseChain* element form.

The *VariableType* attribute specifies the type of the variable being declared.

```
<!element
  ArgumentDeclaration   -- Argument declaration --
                        -- Clause 9.9 --
  - O
  (Name)
>


<!attlist
  ArgumentDeclaration
   HyTime               -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"     -- HyTime bridge element form --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   VariableType          -- used to identify the type of the argument being declared --
    (%VariableTypes;)
    #REQUIRED
>
```

9.10 Execute a response chain

The **execute a response chain** (*ExecuteResponseChain*) element form shall be used to execute the responses contained in an existing response chain.

The HyTime attribute *ireftype* associates names of referential attributes with element types to which their immediate targets must conform. The *ireftype* attribute contains the name of a *ResponseChainRef* attribute followed by the GI of the *ResponseChain* element. The value of this attribute shall be fixed in the encompassing architecture (DTD).

The **response chain reference** (*ResponseChainRef*) attribute form is used to reference an existing response chain.

```
<!element
  ExecuteResponseChain -- Execute a response chain --
                       -- Clause 9.10 --
  - O
  (Argument*)
>

<!attlist
  ExecuteResponseChain
   HyTime              -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"    -- HyTime bridge element form --
   ireftype            -- used to indicate which element types can be referenced as
                          response chains. --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    CDATA
    " ResponseChainRef ResponseChain"
   ResponseChainRef    -- used to reference an existing response chain --
    IDREF
    #REQUIRED
>
```

9.11 Control Flow

The control flow elements allow for conditional execution of response elements within response chains.

9.11.1 If statement

The **if statement** (*If*) element form shall be used to create "if–if–else" constructs.

If the *If* expression evaluates to true, the responses directly within the *If* element are executed. Otherwise, if one of the *Elseif* expressions evaluates to true, the responses within that *ElseIf* element are executed. If none of the *If* or *Elseif* expressions evaluates to true, the responses within the *Else* , if one exists, are executed.

```
<!element
  If                -- If statement --
                    -- Clause 9.11.1 --
  - O
  (Expression, (%Responses;)*, ElseIf*, Else?)
>
```

9.11.2 Elseif statement

```
<!element
  ElseIf           -- Elseif statement --
                   -- Clause 9.11.2 --
  - O
  ( Expression, (%Responses;)*)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
>
```

9.11.3 Else statement

```
<!element
  Else              -- Else statement --
                    -- Clause 9.11.3 --
  - O
  (%Responses;)*
>
```

9.11.4 While statement

The **while statement** (*While*) element form shall be used to create "while" constructs.

The expression is evaluated. If the expression is true, the responses are executed. The expression is reevaluated and the responses are re−executed until the expression evaluates to false.

```
<!element
  While           -- While statement --
                  -- Clause 9.11.4 --
  - O
  ( Expression, (%Responses;)*)
>
```

9.11.5 Switch, Case and Default Statements

The **switch statement** (*Switch*) element form shall be used to determine, based on the evaluation of an expression, which of several optional responses or series of responses should be executed. To accomplish this task, the *Switch* expression is evaluated and then each of the *Case* expressions is evaluated, in the order in which the cases appear, until the value resulting from evaluating a *Case* expression matches the value which resulted from evaluating the *Switch* expression. If a match is found, the responses under the matched *Case* are executed until a *Break* is encountered or until the end of the *Switch* is encountered. Control continues to the responses under the next *Case* if no *Break* is encountered; the interceding expression is not evaluated. If no *Case* expression is matched, the responses in the *Default* element, if one exists, are executed.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element
  Switch            -- Switch statement --
                    -- Clause 9.11.5 --
  - O
  ( Expression, Case+, Default?)
>


<!element
  Case              -- Case statement --
                    -- Clause 9.11.5 --
  - O
  ( Expression, (%Responses;)*, Break?)
>


<!element
  Default           -- Default statement --
                    -- Clause 9.11.5 --
  - O
  ( (%Responses;)+)
>
```

9.11.6 Break statement

The **break statement** (*Break*) element form may be used, but is not required, to terminate a *Case* statement.

```
<!element
  Break             -- Break statement --
                    -- Clause 9.11.6 --
  - O
  EMPTY
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

9.11.7 Expression

The **expression** (*Expression*) element form shall be used to identify and evaluate expressions.

The *Notation* attribute indicates the expression language that must be used to evaluate the content of the *Expression* element (see Clause 8 The ISMID expression language)

```
<!element
  Expression           -- Expression --
                       -- Clause 9.11.7 --
  - O
  (#PCDATA)
>

<!attlist
  Expression
   HyTime              -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"    -- HyTime bridge element form --
   ExpressionLanguage  -- used to identify the expression language used in the content of
                          this element --
    NOTATION  (DSSSL | HyFunk)
    "DSSSL"
   ID                  -- unique identifier --
    ID
    #IMPLIED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

9.11.8 Argument

The **argument** (*Argument*) element form shall be used to pass arguments when executing response chains using *ExecuteResponseChain* or external processes using *Execute* .

```
<!element
  Argument          -- Argument --
                    -- Clause 9.11.8 --
  - O
  (Expression)
>
```

9.11.9 Variable declaration

The **variable declaration** (*VariableDeclaration*) element form shall be used to define the variables that will be used within the MID. Variables must be declared before use. The variable can be assigned a value when it is declared by using the *Expression* element.

The *VariableType* attribute specifies the type of the variable.

```
<!element
  VariableDeclaration    -- Variable declaration --
                         -- Clause 9.11.9 --
  - O
  ( Name, Expression?)
>

<!attlist
  VariableDeclaration
   HyTime                -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"      -- HyTime bridge element form --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   VariableType          -- used to identify the type of variable being declared --
    (%VariableTypes;)
    #REQUIRED
>
```

9.11.10 Assignment statement

The **assignment statement** (*Assign*) element form shall be used to assign a value to a variable.

```
<!element
  Assign             -- Assignment statement --
                     -- Clause 9.11.10 --
  - O
  (Name, Expression)
>
```

9.11.11 Variable name

The **variable name** (*Name)* element form contains the name of the variable being defined or assigned a value.

```
<!element
  Name          -- Variable name --
                -- Clause 9.11.11 --
  - O
  (#PCDATA)
>
```

9.11.12 Execute an external process

The **execute an external process** (*Execute*) element form is used to a invoke an external process. There is no requirement that the MID communicate with the external process after it is launched.

The value of the *Process* attribute is used to specify the external process to be invoked. The value of this attribute is an entity that must have been declared prior to using it as the value of this attribute.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element
  Execute           -- Execute an external process --
                    -- Clause 9.11.12 --
  - O
  (Argument*)
>


<!attlist
  Execute
   HyTime           -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid" -- HyTime bridge element form --
   Process          -- used to identify the external process to be executed --
    ENTITY
    #REQUIRED
   ID               -- unique identifier --
    ID
    #IMPLIED
>
```

9.11.13 MID end

The **MID end** (*End*) element form is used to terminate the execution of the MID.

```
<!element
  End          -- MID end --
               -- 9.11.13 --
  - O
  EMPTY
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

10 ISMID Application Definition Document (IADD)

This clause states the requirements for creating and documenting a conforming ISMID Application. It defines how the syntax and semantics must be specified for each object type, stimulus, response, and property. A sample IADD is shown in AnnexB: Sample ISMID Application Definition Document (Informative) .

10.1 Architectural Form Usage

Each object declared in the IADD must be derived from one of the architectural forms defined in Clause 9 The ISMID architecture.

10.2 Requirements

A conforming IADD shall contain the following sections. Each section shall be written as described below.

10.2.1 Encompassing Architecture (Document Type Definition)

The syntax of a conforming IADD shall be governed by a valid SGML Document Type Definition or encompassing architecture.

This section of the IADD contains the declarations that are defined throughout the IADD. There need not be any comments within this section as each declaration is described in following sections.

10.2.2 Data Content Notations

A conforming IADD shall declare in this section all data content notations that the ISMID system must support. The semantics of each notation and the level to which the system must support the notation may be further described in this section, as deemed necessary by the IADD author.

10.2.3 Entities

A conforming IADD shall declare and define all parameter entities.

10.2.4 Stimuli

A conforming IADD shall declare and define the semantics of each stimulus that is listed in the content model of any element in the encompassing architecture (DTD). A unique name shall be given to each stimulus that has a unique semantic. Stimuli shall be categorized as follows:

10.2.4.1 User−generated

User−generated stimuli are invoked as a result of user action, such as a mouse click or key press in a graphical user interface or a spoken command in a voice−activated interface.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

10.2.4.1.1 [Stimulus Name]

This section structure shall be repeated in the IADD for each user−generated stimulus. The heading « [Stimulus Name] » shall be replaced with each element name derived from the Stimulus architectural form.

10.2.4.1.2 Semantic Definition

This section contains a semantic definition for the named stimulus.

10.2.4.1.3 Formal Declaration

This section contains the formal SGML element and attribute definition list declarations for the named stimulus. Arguments required to be passed to an expression via a stimulus shall be declared as attributes and described in the following section.

10.2.4.1.4 Parameter Descriptions

Each parameter that is passed to an expression as an argument shall be described here. This section shall be omitted if there are no parameters defined in the formal declaration. Parameters and descriptions shall be presented in the following format:

Note 19. Parameters and descriptions should be presented in a form similar to the terms and definitions in the glossary section.

10.2.4.2 Object−generated

Object−generated stimuli are invoked from within a created object such as a video container completing the playing of a video.

10.2.4.2.1 [Stimulus Name]

This section structure shall be repeated in the IADD for each object−generated stimulus. The heading « [Stimulus Name] » shall be replaced with each element name derived from the Stimulus architectural form.

10.2.4.2.2 Semantic Definition

This section contains a semantic definition for the named stimulus.

10.2.4.2.3 Formal Declaration

This section contains the formal SGML element and attribute definition list declarations for the named stimulus. Arguments required to be passed to an expression via a stimulus shall be declared as attributes and described in the following section.

10.2.4.2.4 Parameter Descriptions

Each parameter that is passed to an expression through an argument shall be described here. This section shall be omitted if there are no parameters defined in the formal declaration. Parameters and descriptions shall be presented in the following format:

Note 20. Parameters and descriptions should be presented in a form similar to the terms and definitions in the glossary section.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

10.2.4.3 Externally−generated

Externally−generated stimuli are initiated from external sources such as input from an operating environment or hardware diagnostic system.

10.2.4.3.1 [Stimulus Name]

This section structure shall be repeated in the IADD for each externally−generated stimulus. The heading « [Stimulus Name] » shall be replaced with each element name derived from the Stimulus architectural form.

10.2.4.3.2 Semantic Definition

This section contains a semantic definition for the named stimulus.

10.2.4.3.3 Formal Declaration

This section contains the formal SGML element and attribute definition list declarations for the named stimulus. Arguments required to be passed to an expression via a stimulus shall be declared as attributes and described in the following section.

10.2.4.3.4 Parameter Descriptions

Each parameter that is passed to an expression through an argument shall be described here. This section shall be omitted if there are no parameters defined in the formal declaration. Parameters and descriptions shall be presented in the following format:

Note 21. Parameters and descriptions should be presented in a form similar to the terms and definitions in the glossary section.

10.2.5 Properties

A conforming IADD shall define the semantics of each property declared in the attribute definition list of any object element in the encompassing architecture (DTD). A unique name shall be given to each property that has a unique semantic.

10.2.6 Objects

A conforming IADD shall define each object type that can be created, modified, and destroyed in this ISMID application. The objects shall be classified and documented as described in the following subparagraphs.

10.2.6.1 Container Objects

A conforming IADD shall list each container object type that is declared in the ISMID application being defined.

10.2.6.1.1 [Container Object Name]

Each container object type shall be listed by name. This section structure shall be repeated in the IADD for each container object. The heading « [Container Object Name] » shall be replaced with each element name derived from the CreateContainer architectural form.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

10.2.6.1.2 Semantics

The semantics of this container object type shall be defined.

10.2.6.1.3 Stimuli

Stimuli that can be accepted by this container object type shall be listed.

10.2.6.1.4 Properties

Properties valid for this container type shall be listed.

10.2.6.1.5 CreateContainer Declaration

The architectural form and declaration for creating this object type shall be defined.

10.2.6.1.6 ModifyContainer Declaration

The architectural form and declaration for modifying this object type shall be defined.

10.2.6.2 Content Objects

A conforming IADD shall list each content object type that is declared in the ISMID application being defined.

10.2.6.2.1 [Content Object Name]

Each content object type shall be listed by name. This section structure shall be repeated in the IADD for each content object. The heading « [Content Object Name] » shall be replaced with each element name derived from the CreateContentObject architectural form.

10.2.6.2.2 Semantics

The semantics of this content object type shall be defined.

10.2.6.2.3 Stimuli

Stimuli that can be accepted by this content object type shall be listed.

10.2.6.2.4 Properties

Properties valid for this content object type shall be listed.

10.2.6.2.5 CreateContentObject Declaration

The architectural form and declaration for creating this object type shall be defined.

10.2.6.2.6 ModifyContentObject Declaration

The architectural form and declaration for modifying this object type shall be defined.

--`,,,`-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

10.2.6.3 Control Objects

A conforming IADD shall list each control object type that is declared in the ISMID application being defined.

10.2.6.3.1 [Control Object Name]

Each control object type shall be listed by name. This section structure shall be repeated in the IADD for each control object. The heading « [Control Object Name] » shall be replaced with each element name derived from the CreateControl architectural form.

10.2.6.3.2 Semantics

The semantics of this control object type shall be defined.

10.2.6.3.3 Stimuli

Stimuli that can be accepted by this control object type shall be listed.

10.2.6.3.4 Properties

Properties valid for this control object type shall be listed.

10.2.6.3.5 CreateControl Declaration

The architectural form and declaration for creating this object type shall be defined.

10.2.6.3.6 ModifyControl Declaration

The architectural form and declaration for modifying this object type shall be defined.

10.2.6.3.7 External Objects

A conforming IADD shall list each external object type that is declared in the ISMID application being defined.

10.2.6.3.8 [External Object Name]

Each external object type shall be listed by name. This section structure shall be repeated in the IADD for each external object. The heading « [External Object Name] » shall be replaced with each element name derived from the CreateExternalObject architectural form.

10.2.6.3.9 Semantics

The semantics of this external object type shall be defined.

10.2.6.3.10 Stimuli

Stimuli that can be accepted by this external object type shall be listed.

10.2.6.3.11 Properties

Properties valid for this external object type shall be listed.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

10.2.6.3.12 CreateExternalObject Declaration

The architectural form and declaration for creating this object type shall be defined.

10.2.6.3.13 ModifyExternalObject Declaration

The architectural form and declaration for modifying this object type shall be defined.

Annex A
(normative)
ISMID Meta−DTD

While the meta−DTD that appears here is normative, it is also possible for conforming applications to use a meta−DTD that is the same in all respects except for differences dictated by the need to use a different SGML declaration.

Note 22. For example, the meta−DTD will differ in certain syntactic details when used in XML (WebSGML) applications.

```
<?IS10744 ArcBase HyTime>


<!NOTATION
  HyTimePUBLIC
   "ISO/IEC 10744:1997//NOTATION AFDR ARCBASEHypermedia/Time-based
    Structuring Language (HyTime)//EN"
>


<!ATTLIST #NOTATION HyTime
   ArcFormA  NAME              HyTime
   ArcNamrA  NAME              HyNames
   ArcSuprA  NAME              sHyTime
   ArcIgnDA  NAME              HyIgnD
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   ArcDocF   NAME                #FIXED HyDoc
   ArcDTD    CDATA               "HyTime"
   ArcQuant  CDATA               #FIXED "NAMELEN 9"
   ArcDataF  NAME                #FIXED HyBridN
   ArcBridF  NAME                #FIXED HyBrid
   ArcAuto  (ArcAuto|nArcAuto)   nArcAuto
   ArcOptSA  NAMES               "GenArc base links"
   GenArc    CDATA               "ireftype"
   base      CDATA               "HyFunk"
   locs      CDATA               "referatt refloc reftype mixedloc
                                  nameloc nmsploc treeloc"
   links     CDATA               ""
   sched     CDATA               ""
   rend      CDATA               ""
   anysgml   (anysgml|nanysgml)  anysgml
   exrefs    (exrefs|nexrefs)    exrefs
   refmodel  (SGMLmdl|nSGMLmdl)  nSGMLmdl
   hyqcnt    NUMBER              32
   manyanch  NUMBER              #IMPLIED
   manyaxes  NUMBER              "4"
>


<!NOTATION
  AFDRMeta
    PUBLIC "ISO/IEC 10744//NOTATION AFDR Meta-DTD Notation//EN">


<!ENTITY
   HyTime
    PUBLIC
    "ISO/IEC 10744:1997//DTD AFDR Meta-DTD Hypermedia/Time-based Structuring
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    Language (HyTime)//EN"
    CDATA AFDRMeta>


<!entity %
   CreateObject  -- Create an object --
    "CreateContainer  | CreateContentObject  | CreateControl | CreateExternalObject">


<!entity %
   ControlFlow -- Control flow --
    "If | While | Switch">


<!entity %
   VariableTypes -- Variable types --
    "Boolean | i1 | i2 | i4 | i8 | ui1 | ui2 | ui4 | ui8 | r4 | r8 | String | Char | Variant">


<!entity %
   ModifyObject -- Modify an object --
    "ModifyContainer | ModifyContentObject | ModifyControl | ModifyExternalObject">


<!entity %
   Responses  -- Responses --
    "%CreateObject; | %ControlFlow; | %ModifyObject; | DestroyObject |
     ExecuteResponseChain | Execute">


<!element
  MID              -- Multimedia Interactive Document document element --
                   -- Clause 9.2 --
  - O
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
  (ContentReference*, SystemChain, ResponseChain*)
>


<!attlist
  MID
   HyTime           -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid" -- HyTime bridge element form --
   ID               -- unique identifier --
    ID
    #REQUIRED
>


<!element
  ContentReference  -- Content reference --
                    -- Clause 9.2 --
  - O
  (nmlist)
>


<!attlist
  ContentReference
   HyTime             -- HyTime architectural form name --
    CDATA
    #FIXED "nameloc"  -- HyTime nameloc element form --
   ContentNotExist    -- Referenced content does not exist --
                      -- specifies whether or not the author wants the system to
                         report content reference errors or to ignore them --
    (error | ignore)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
    "ignore"
>


<!element
  nmlist      -- Name list --
              -- Clause 9.2 --
  - O
  (#PCDATA)
>


<!attlist
  nmlist
   HyTime           -- HyTime architectural form name --
    CDATA
    #FIXED "nmlist" -- HyTime nmlist element form --
   nametype         -- HyTime nametype attribute form --
    (element | entity)
    "entity"
   docorsub         -- HyTime docorsub attribute form --
    #ENTITY
    #IMPLIED
>


<!entity %
   CommonObjectAttributes "
    HyTime
     NAME
     #FIXED `HyBrid'
    valueref
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
     CDATA
     #IMPLIED
    ObjectNotExist
    (error | ignore)
    `ignore'
">


<!element
  CreateContainer  -- Create a container object --
                   -- Clause 9.5.2 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>


<!attlist
  CreateContainer
   %CommonObjectAttributes; -- Common object attributes --
   ireftype                 -- Used to indicate which elements are valid as parents of
                               this container object --
    CDATA
    #REQUIRED
   Parent                   -- Used to identify the parent of this container object.
                               Constraint:  must reference element types derived from either
                               the MID or the CreateContainer element forms --
    IDREF
    #REQUIRED
   ID                       -- Unique identifier --
    ID
    #REQUIRED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element
  ModifyContainer -- Modify a container object --
                  -- Clause 9.5.3 --
  - O
  (Assign*, Expression*, Stimulus*)
>


<!attlist
  ModifyContainer
   %CommonObjectAttributes;
   ireftype           -- used to indicate which object types can be modified
                         by this element.  It may also indicate which object types
                         are valid as parents for this object. --
    CDATA
    #REQUIRED
   Object             -- Identifies the container object to be modified
                         Constraint: the value of this attribute must be the unique
                         identifier of an existing container object. --
    IDREF
    #REQUIRED
   Parent             -- Used to identify the parent of this content object.
                         Constraint:  must reference element types derived from the MID or
                         CreateContainer element forms. --
    IDREF
    #IMPLIED
>


<!element
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
  CreateContentObject  -- Create a content object --
                       -- Clause 9.5.4 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>


<!attlist
  CreateContentObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                -- Used to indicate which element can be referenced by the
                              SemanticContent attribute and which elements are valid as
                              parents of this content object --
    CDATA
    "SemanticContent ContentReference"
   Parent                  -- Used to identify the parent of this content object.
                              Constraint:  must reference element types derived from the
                              CreateContainer element form. --
    IDREF
    #REQUIRED
   SemanticContent          -- used to reference the information to be rendered in
                               this content object --
    IDREF
    #IMPLIED
   ID                       -- unique identifier --
    ID
    #REQUIRED
>


<!element
  ModifyContentObject  -- Modify a content object --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                        -- Clause 9.5.5 --
  - O
  (Assign*, Expression*, Stimulus*)
>


<!attlist
  ModifyContentObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which element can be referenced by the
                               SemanticContent attribute; which object types can be modified
                               by this element; and which elements are valid as parents of
                               this content object. --
    CDATA
    "SemanticContent ContentReference"
   Object                   -- Identifies the content object to be modified
                               Constraint: the value of this attribute must be the unique
                               identifier of an existing content object. --
    IDREF
    #REQUIRED
   Parent                   -- Used to identify the parent of this content object.
                               Constraint:  must reference element types derived from the
                               CreateContainer element form. --
    IDREF
    #IMPLIED
   SemanticContent          -- used to reference the information to be rendered in
                               this content object --
    IDREF
    #IMPLIED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!element
  CreateControl             -- Create a control object --
                            -- Clause 9.5.6 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>


<!attlist
  CreateControl
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which elements are valid as parents of
                               this control object --
    CDATA
    #REQUIRED
   Parent                   -- Used to identify the parent of this control object.
                               Constraint:  must reference element types derived from the
                               CreateContainer or CreateContentObject element forms. --
    IDREF
    #REQUIRED
   ID                       -- unique identifier --
    ID
    #REQUIRED
>


<!element
  ModifyControl             -- Modify a control object --
                            -- Clause 9.5.7 --
  - O
  (Assign*, Expression*, Stimulus*)
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!attlist
  ModifyControl
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- used to indicate which object types can be modified
                               by this element.  It may also indicate which object types are valid a
                               parents for this object. --
    CDATA
    #IMPLIED
   Object                   -- Identifies the control object to be modified
                               Constraint: the value of this attribute must be the unique
                               identifier of an existing control object. --
    IDREF
    #REQUIRED
   Parent                   -- Used to identify the parent of this control object.
                               Constraint:  must reference element types derived from the
                               CreateContainer or CreateContentObject element forms. --
    IDREF
    #IMPLIED
>


<!element
  CreateExternalObject      -- Create an external object --
                            -- Clause 9.5.8 --
  - O
  (Label*, VariableDeclaration*, Expression*, Stimulus*)
>


<!attlist
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
  CreateExternalObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- Used to indicate which elements are valid as parents of
                               this external object --
    CDATA
    #IMPLIED
   Parent                   -- Used to identify the parent of this external object.
                               Constraint:  must reference element types derived from the
                               CreateContainer element form. --
    IDREF
    #REQUIRED
   ID                       -- unique identifier --
    ID
    #REQUIRED
>


<!element
  ModifyExternalObject      -- Modify an external object --
                            -- Clause 9.5.9 --
  - O
  (Assign*, Expression*, Stimulus*)
>


<!attlist
  ModifyExternalObject
   %CommonObjectAttributes; -- see clause 9.5.1 --
   ireftype                 -- used to indicate which object types can be modified
                               by this element.  It may also indicate which object types
                               are valid as parents for this object. --
    CDATA
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
    #IMPLIED
   Object                   -- Identifies the external object to be modified
                               Constraint: the value of this attribute must be the unique
                               identifier of an existing external object. --
    IDREF
    #REQUIRED
   Parent                   -- Used to identify the parent of this external object.
                               Constraint:  must reference element types derived from the
                               CreateContainer element form. --
    IDREF
    #IMPLIED
>


<!element
  DestroyObject      -- Destroy an object --
                     -- Clause 9.5.10 --
  - O
  EMPTY
>


<!attlist
  DestroyObject
   HyTime            -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"  -- HyTime bridge element form --
   ireftype          -- used to indicate which create object element types are valid
                        objects to be destroyed.  --
    CDATA
    #IMPLIED
   Object            -- Identifies the object to be destroyed
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
                        Constraint: the value of this attribute must be the unique
                        identifier of an existing object. --
    IDREF
    #REQUIRED
>


<!element
  Label              -- Object label --
                     -- Clause 9.6 --
  - O
  (#PCDATA)
>


<!attlist
  Label
   HyTime            -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid" -- HyTime bridge element form --
   xml:lang          -- used to specify the language used in the content of this element --
    CDATA
    #REQUIRED
>


<!element
  Stimulus            -- Stimulus --
                      -- Clause 9.7 --
  - O
  ((%Responses;)*, End?)
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!attlist
  Stimulus
   HyTime             -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"   -- HyTime bridge element form --
   ireftype           -- used to indicate which element type is the response chain
                         element type  --
    CDATA
    "ResponseChainRef ResponseChain"
   ResponseChainRef   -- used to reference an existing response chain --
    IDREF
    #IMPLIED
>


<!element
  (SystemChain | ResponseChain) -- System response chain --
                                -- Response chain --
                                -- Clause 9.8 --
  - O
  (Label*, ArgumentDeclaration*, (%Responses;)*, End?)
>


<!attlist
  ResponseChain
   HyTime                       -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"             -- HyTime bridge element form --
   ID                           -- unique identifier --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
    ID
    #IMPLIED
>


<!element
  ArgumentDeclaration   -- Argument declaration --
                        -- Clause 9.9 --
  - O
  (Name)
>


<!attlist
  ArgumentDeclaration
   HyTime               -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"     -- HyTime bridge element form --
   VariableType         -- used to identify the type of the argument being declared --
    (%VariableTypes;)
    #REQUIRED
>


<!element
  ExecuteResponseChain -- Execute a response chain --
                       -- Clause 9.10 --
  - O
  (Argument*)
>


<!attlist
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
  ExecuteResponseChain
   HyTime              -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"    -- HyTime bridge element form --
   ireftype            -- used to indicate which element types can be referenced as
                          response chains. --
    CDATA
    " ResponseChainRef ResponseChain"
   ResponseChainRef    -- used to reference an existing response chain --
    IDREF
    #REQUIRED
>


<!element
  If                -- If statement --
                    -- Clause 9.11.1 --
  - O
  (Expression, (%Responses;)*, ElseIf*, Else?)
>


<!element
  ElseIf            -- Elseif statement --
                    -- Clause 9.11.2 --
  - O
  ( Expression, (%Responses;)*)
>


<!element
  Else              -- Else statement --
```


--`,,`-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

```
                    -- Clause 9.11.3 --
  - O
  ( (%Responses;)*)
>


<!element
  While           -- While statement --
                  -- Clause 9.11.4 --
  - O
  ( Expression, (%Responses;)*)
>


<!element
  Switch            -- Switch statement --
                    -- Clause 9.11.5 --
  - O
  ( Expression, Case+, Default?)
>


<!element
  Case              -- Case statement --
                    -- Clause 9.11.5 --
  - O
  ( Expression, (%Responses;)*, Break?)
>


<!element
  Default           -- Default statement --
                    -- Clause 9.11.5 --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
  - O
  ( (%Responses;)+)
>


<!element
  Break             -- Break statement --
                    -- Clause 9.11.6 --
  - O
  EMPTY
>


<!element
  Expression          -- Expression --
                      -- Clause 9.11.7 --


  - O
  (#PCDATA)
>


<!attlist
  Expression
   HyTime             -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"   -- HyTime bridge element form --
   ExpressionLanguage -- used to identify the expression language used in the content of
                         this element --
    NOTATION  (DSSSL | HyFunk)
    "DSSSL"
   ID                 -- unique identifier --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
    ID
    #IMPLIED
>


<!element
  Argument          -- Argument --
                    -- Clause 9.11.8 --
  - O
  (Expression)
>


<!element
  VariableDeclaration    -- Variable declaration --
                         -- Clause 9.11.9 --
  - O
  ( Name, Expression?)
>


<!attlist
  VariableDeclaration
   HyTime                -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid"      -- HyTime bridge element form --
   VariableType          -- used to identify the type of variable being declared --
    (%VariableTypes;)
    #REQUIRED
>


<!element
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
  Assign              -- Assignment statement --
                      -- Clause 9.11.10 --
  - O
  (Name, Expression)
>


<!element
  Name           -- Variable name --
                 -- Clause 9.11.11 --
  - O
  (#PCDATA)
>


<!element
  Execute            -- Execute an external process --
                     -- Clause 9.11.12 --
  - O
  (Argument*)
>


<!attlist
  Execute
   HyTime            -- HyTime architectural form name --
    NAME
    #FIXED "HyBrid" -- HyTime bridge element form --
   Process           -- used to identify the external process to be executed --
    ENTITY
    #REQUIRED
   ID                -- unique identifier --
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    ID

    #IMPLIED

>



<!element

  End          -- MID end --

               -- 9.11.13 --

  - O

  EMPTY

>
```

Annex B
(informative)
Sample ISMID Application Definition Document

B.1 Encompassing Architecture (Document Type Definition)

This section of the IADD contains the declarations that are defined throughout this document for the encompassing architecture. There are no comments within this section; each declaration is described in following sections.

```
<!NOTATION

  DSSSLExp
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   PUBLIC
   "ISO/IEC 10179:1996//NOTATION DSSSL Expression Language Notation//EN"
>


<!NOTATION
  JPEG
   PUBLIC
   "ISO/IEC 10918-1:1994//NOTATION Digital Compression and
    Coding of Continuous-tone Still Images (JPEG)//EN"
>


<!NOTATION
  MPEG1vid
   PUBLIC
   "ISO/IEC 11172-2:1993//NOTATION Information technology - Coding of
    moving pictures and associated audio for digital storage media at up
    to about 1,5 Mbit/s - Part 2: Video//EN"
>


<!NOTATION
  MPEG1aud
   PUBLIC
   "ISO/IEC 11172-3:1993//NOTATION Information technology - Coding of
    moving pictures and associated audio for digital storage media at up to
    about 1,5 Mbit/s - Part 3: Audio//EN"
>


<!NOTATION
  CGMbin
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   PUBLIC
   "ISO/IEC 8632-3:1992//NOTATION Information technology - Computer graphics
    - Metafile for the storage and transfer of picture description information
    - Part 3: Binary encoding//EN"
>

<!NOTATION
  SGML
   PUBLIC
   "ISO 8879:1986//NOTATION Information processing - Text and office systems
     - Standard Generalized Markup Language (SGML)//EN"
>

<!entity % ControlFlow "If | While | Switch">

<!entity % VariableTypes "Boolean | i1 | i2 | i4 | i8 | ui1 | ui2 | ui4 |  ui8  | r4 | r8 | String |

<!entity % CreateObject  "CreateWindow | CreateImage | CreateButton |  CreateVideo  |  CreateHotSpot

<!entity % ModifyObject  "ModifyWindow | ModifyImage | ModifyButton |  ModifyVideo  |   ModifyHotSpo

<!entity % Responses "%CreateObject; | %ControlFlow; | %ModifyObject; |  DestroyObject  |  ExecuteRe

<!element MID - O (ContentReference*, SystemChain, ResponseChain*)>

<!attlist MID ID ID #REQUIRED>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!element ContentReference - O (nmlist)>


<!attlist ContentReference
   HyTime CDATA #FIXED "nameloc"
   ContentNotExist (error | ignore) "ignore"
>


<!element nmlist - O (#PCDATA)>


<!attlist nmlist
   HyTime CDATA #FIXED "nmlist"
   nametype (element | entity) "entity"
   docorsub ENTITY #IMPLIED
>


<!element DestroyObject - O EMPTY>


<!attlist DestroyObject
   ireftype CDATA #IMPLIED
   Object IDREF #REQUIRED
>


<!element (SystemChain | ResponseChain) - O (Label*, ArgumentDeclaration*, (%Responses;)*, End?)>


<!attlist ResponseChain   ID ID #IMPLIED>


<!element ArgumentDeclaration - O (Name)>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!attlist ArgumentDeclaration   VariableType (%VariableTypes;) #REQUIRED>

<!element ExecuteResponseChain - O (Argument*)>

<!attlist ExecuteResponseChain
   ireftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #REQUIRED
>

<!element If - O (Expression, (%Responses;)*, ElseIf*, Else?)>

<!element ElseIf - O ( Expression, (%Responses;)*)>

<!element Else - O (%Responses;)*>

<!element While - O ( Expression, (%Responses;)*)>

<!element Switch - O ( Expression, Case+, Default?)>

<!element Case - O ( Expression, (%Responses;)*, Break?)>

<!element Default - O ( (%Responses;)*, Break?)>

<!element Break - O EMPTY>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element Expression - O (#PCDATA)>


<!attlist Expression
   ExpressionLanguage NOTATION (DSSSL | HyFunk) "DSSSL"
   ID ID #IMPLIED
>


<!element Argument - O (Expression)>


<!element VariableDeclaration - O ( Name, Expression?)>


<!attlist VariableDeclaration   VariableType (%VariableTypes;) #REQUIRED>


<!element Assign - O (Name, Expression)>


<!element Name - O (#PCDATA)>


<!element Execute - O (Argument*)>


<!attlist Execute   Process ENTITY #Required   ID ID #IMPLIED>


<!element End - O EMPTY>


<!entity %
  CreateFontProperties
   'FontName CDATA #REQUIRED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    FontSize  NUMBER  "8"
    FontBold (FontBold | NFontBold) "FontBold"'
>

<!entity %
  ModifyFontProperties
   'FontName CDATA #IMPLIED
    FontSize NUMBER #IMPLIED
    FontBold (FontBold | NFontBold) #IMPLIED'
>

<!entity % WindowStimuli "OnResize?, OnDestroy?">

<!entity % ImageStimuli "OnResize?, OnLoadComplete?">

<!entity % ButtonStimuli "OnClick?">

<!entity % HotSpotStimuli "OnClick?">

<!entity % VideoStimuli "OnEnd?">

<!entity % AudioStimuli "OnEnd?">

<!entity % SGMLStimuli "OnNotationClick?">

<!entity % CGMStimuli "OnObjectClick?">
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element OnClick - O ((%Responses;)*, End?)>


<!ATTLIST  OnClick
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
   Button NUMBER #IMPLIED
>


<!element OnResize - O ((%Responses;)*, end?)>


<!ATTLIST  OnResize
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>


<!element OnDestroy - O ((%Responses;)*, End?)>


<!ATTLIST  OnDestroy
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>


<!element OnLoadComplete - O ((%Responses;)*, End?)>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!ATTLIST  OnLoadComplete
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>

<!element OnEnd - O ((%Responses;)*, End?)>

<!ATTLIST  OnEnd
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>

<!element OnNotationClick - O ((%Responses;)*, End?)>

<!ATTLIST  OnNotationClick
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
   Notation Name #IMPLIED
   EntityName Name #IMPLIED
   SystemID CDATA #IMPLIED
>

<!element OnObjectClick - O ((%Responses;)*, End?)>

<!ATTLIST  OnObjectClick
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>

<!element   CreateWindow  - O (Label*, VariableDeclaration*, (%WindowStimuli;))>

<!ATTLIST CreateWindow
   ISMID NAME #FIXED "CreateContainer"
   ireftype CDATA #FIXED "Parent MID"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Caption CDATA #IMPLIED
   TitleBar (TitleBar | NTitleBar) "TitleBar"
   WindowState (Normal | Maximized | Minimized) "Normal"
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
   %CreateFontProperties;
>

<!element ModifyWindow - O (Assign*, (%WindowStimuli;))>

<!ATTLIST ModifyWindow
   ISMID NAME #FIXED "ModifyContainer"
   ireftype CDATA #FIXED "Object CreateWindow"
   Object IDREF #REQUIRED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   Caption CDATA #IMPLIED
   TitleBar (TitleBar | NTitleBar) #IMPLIED
   WindowState (Normal | Maximized | Minimized) #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   %ModifyFontProperties;
>

<!element CreateImage - O (Label*, VariableDeclaration*, Expression*, (%ImageStimuli;))>

<!ATTLIST CreateImage
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Visible (Visible | NVisible) "Visible"

   ActualWidth NUMBER #IMPLIED
   ActualHeight NUMBER #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element ModifyImage - O (Assign*, Expression*, (%ImageStimuli;))>


<!ATTLIST ModifyImage
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   valueref CDATA "SemanticContent SemanticContent"
   ireftype CDATA #FIXED "Object CreateImage Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   Object IDREF #REQUIRED
   ActualWidth NUMBER #IMPLIED
   ActualHeight NUMBER #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
>


<!element CreateVideo - O (Label*, VariableDeclaration*, Expression*, (%VideoStimuli;))>



<!ATTLIST CreateVideo
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Playing (Playing | NPlaying) "NPlaying"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   CurrentFrame NUMBER "0"
   VideoHeight NUMBER #IMPLIED
   VideoWidth NUMBER #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>

<!element ModifyVideo - O (Assign*,  Expression*, (%VideoStimuli;))>

<!ATTLIST ModifyVideo
   ISMID NAME #FIXED "ModifyContentObject"
  SemanticContent DREF IMPLIED
   ireftype CDATA #FIXED "Object CreateVideo Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   Playing (Playing | NPlaying) #IMPLIED
   CurrentFrame NUMBER #IMPLIED
   Left  CDATA#IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
>

<!element CreateAudio - O (Label*, VariableDeclaration*, Expression*, (%AudioStimuli;))>

<!ATTLIST CreateAudio
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Playing (Playing | NPlaying) "NPlaying"
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
>

<!element ModifyAudio - O (Assign*, Expression*, (%AudioStimuli;))>

<!ATTLIST ModifyAudio
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateAudio Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   Playing (Playing | NPlaying) #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
>

<!element CreateSGML - O (Label*, VariableDeclaration*, Expression*, (%SGMLStimuli;))>

<!ATTLIST CreateSGML
   ISMID NAME #FIXED "CreateContentObject"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   StyleSheet ENTITY #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>

<!element ModifySGML - O (Assign*, Expression*, (%SGMLStimuli;))>

<!ATTLIST ModifySGML
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateSGML Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ActualWidth NUMBER #IMPLIED
   ActualHeight CDATA #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   %ModifyFontProperties;
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element CreateCGM - O (Label*, VariableDeclaration*, Expression*, (%CGMStimuli;))>


<!ATTLIST CreateCGM
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Visible (Visible | NVisible) "Visible"
   ActualWidth NUMBER #IMPLIED
   ActualHeight CDATA #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>


<!element ModifyCGM - O (Assign*, Expression*, (%CGMStimuli;))>


<!ATTLIST ModifyCGM
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateCGM Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ActualWidth NUMBER #IMPLIED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ActualHeight CDATA #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA#IMPLIED
>

<!element CreateButton - O (Label*, VariableDeclaration*, Expression*, (%ButtonStimuli;)*)>

<!ATTLIST CreateButton
   ISMID NAME #FIXED "CreateControlObject"
   ireftype CDATA #FIXED "parent CreateWindow LeftRef Expression
                          TopRef Expression WidthRef Expression
                          HeightRef Expression"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
                          Height HeightRef"
   Enabled (Enabled | NEnabled) "Enabled"
   Visible (Visible | NVisible) "Visible"
   BevelWidth NUMBER "2"
   Caption CDATA #IMPLIED
   ForeColor CDATA "000000"
   Left  NUMBER #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
   WidthRef IDREF #IMPLIED
   TopRef IDREF #IMPLIED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
   %CreateFontProperties;
>

<!element ModifyButton - O (Assign*, Expression*, (%ButtonStimuli;)*)>

<!ATTLIST ModifyButton
   ISMID NAME #FIXED "ModifyControl"
   ireftype CDATA #FIXED "Object CreateButton Parent CreateWindow
                          LeftRef Expression TopRef Expression
                          WidthRef Expression HeightRef Expression"
   Object IDREF #REQUIRED
   Parent IDREF #IMPLIED
   ObjectNotExist (error | ignore) "ignore"
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
                          Height HeightRef"
   Enabled (Enabled | NEnabled) #IMPLIED
   Visible (Visible | NVisible) #IMPLIED
   BevelWidth NUMBER #IMPLIED
   Caption CDATA #IMPLIED
   ForeColor CDATA #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   WidthRef IDREF #IMPLIED
   TopRef IDREF #IMPLIED
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   %ModifyFontProperties;
>


<!element CreateHotSpot - O (Label*, VariableDeclaration*, Expression*, (%HotSpotStimuli;)*)>


<!ATTLIST CreateHotSpot
   ISMID NAME #FIXED "CreateControlObject"
   ireftype CDATA #FIXED "Parent CreateImage LeftRef Expression
                          TopRef Expression WidthRef Expression
                          HeightRef Expression"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
                          Height HeightRef"
   Enabled (Enabled | NEnabled) "Enabled"
   Visible (Visible | NVisible) "Visible"
   Caption CDATA #IMPLIED
   ForeColor CDATA "000000"
   FillColor  CDATA "000000"
   DrawMode (CopyPen | MaskPen) "MaskPen"
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
   WidthRef IDREF #IMPLIED
   TopRef IDREF #IMPLIED
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
   %CreateFontProperties;
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
>


<!element ModifyHotSpot - O (Assign*, (%HotSpotStimuli;)*)>


<!ATTLIST ModifyHotSpot
   ISMID NAME #FIXED "ModifyControl"
   ireftype CDATA #FIXED "Object CreateHotspot Parent CreateImage
                          LeftRef Expression TopRef Expression
                          WidthRef Expression HeightRef Expression"
   Object IDREF #REQUIRED
   Parent IDREF #IMPLIED
   ObjectNotExist (error | ignore) "ignore"
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
                          Height HeightRef"
   Enabled (Enabled | NEnabled) #IMPLIED
   Visible (Visible | NVisible) #IMPLIED
   BevelWidth NUMBER #IMPLIED
   Caption CDATA #IMPLIED
   ForeColor CDATA #IMPLIED
   FillColor  CDATA #IMPLIED
   DrawMode (CopyPen | MaskPen) #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   WidthRef IDREF #IMPLIED
   TopRef IDREF #IMPLIED
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
   %ModifyFontProperties;
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

>

B.2 Data Content Notations

```
<!NOTATION
   BMP
    PUBLIC
     "-//ISMB 0-7923-9432-1:: Graphic Notation//NOTATION Microsoft Device Dependant Bitmap//EN">


<!NOTATION
   JPEG
    PUBLIC
     "ISO/IEC 10918-1:1994//NOTATION Digital Compression and Coding of Continuous-tone
      Still Images (JPEG)//EN" >


<!NOTATION
   MPEG1vid
    PUBLIC
     "ISO/IEC 11172-2:1993//NOTATION Information technology - Coding of moving pictures
      and associated audio for digital storage media at up to about 1,5 Mbit/s - Part 2:
      Video//EN" >


<!NOTATION
   MPEG1aud
    PUBLIC
     "ISO/IEC 11172-3:1993//NOTATION Information technology - Coding of moving pictures
      and associated audio for digital storage media at up to about 1,5 Mbit/s - Part 3:
      Audio//EN" >
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!NOTATION

   CGMbin

    PUBLIC

     "ISO/IEC 8632-3:1992//NOTATION Information technology - Computer graphics -

      Metafile for the storage and transfer of picture description information -

      Part 3: Binary encoding//EN" >



<!NOTATION

   SGML

    PUBLIC

     "ISO 8879:1986//NOTATION Information processing - Text and office systems -

      Standard Generalized Markup Language (SGML)//EN" >
```

B.3 Entities

B.3.1 Create Object

This entity lists all possible object types that can be created in an instance of this ISMID application.

```
<!entity % CreateObject  "CreateWindow | CreateImage | CreateButton | CreateHotSpot | CreateVideo |
```

B.3.2 Modify Object

This entity lists all possible object types that can be modified in an instance of this ISMID application.

```
<!entity % ModifyObject  "ModifyWindow | ModifyImage | ModifyButton | ModifyHotSpot | ModifyVideo |
```

B.3.3 Control Flow

This entity lists all elements used to control the sequence of an interactive process defined by an instance of this ISMID application. Evaluation of the element types listed in this entity declaration will require processing of expressions.

```
<!entity % ControlFlow "If | While | Switch">
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

B.3.4 Responses

This entity lists the possible responses to a given stimulus.

```
<!entity % Responses "%CreateObject; | %ControlFlow; | %ModifyObject; | DestroyObject | ExecuteRespo
```

B.3.5 Window stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateWindow container object element type.

```
<!entity % WindowStimuli "OnResize?, OnDestroy?">
```

B.3.6 Image stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateImage content object element type.

```
<!entity % ImageStimuli "OnResize?, OnLoadComplete?">
```

B.3.7 Video stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateVideo content object element type.

```
<!entity % VideoStimuli "OnEnd?">
```

B.3.8 Audio stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateAudio content object element type.

```
<!entity % AudioStimuli "OnEnd?">
```

B.3.9 SGML stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateSGML content object element

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

type.

```
<!entity % SGMLStimuli "OnNotationClick?">
```

B.3.10 CGM stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateCGM content object element type.

```
<!entity % CGMStimuli "OnObjectClick?">
```

B.3.11 Button stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateButton control object element type.

```
<!entity % ButtonStimuli "OnClick?">
```

B.3.12 Hotspot stimuli

This entity lists the stimuli that can be accepted by objects created with the CreateHotSpot control object element type.

```
<!entity % HotSpotStimuli "OnClick?">
```

B.3.13 Font properties for creating objects

This entity lists the font properties that can be set when the object is created.

```
<!entity %
  CreateFontProperties
   'FontName CDATA #REQUIRED
    FontSize  NUMBER  "8"
    FontBold (FontBold | NFontBold) "FontBold"'
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.3.14 Font properties for modifying objects

This entity lists the font properties that can be modified using a modified object element.

```
<!entity %

  ModifyFontProperties

   'FontName CDATA #IMPLIED

    FontSize NUMBER #IMPLIED

    FontBold (FontBold | NFontBold) #IMPLIED'

>
```

B.4 Stimuli

B.4.1 User−generated

B.4.1.1 OnClick

B.4.1.1.1 Semantic Definition

The OnClick stimulus is initiated by a single click of a user's pointing device. If there are multiple buttons on the pointing device, the left button will initiate this event. The pointing device shall send an OnClick stimulus to a single object at a time.

B.4.1.1.2 Formal Declaration

```
<!element OnClick - O ((%Responses;)*, End?)>
```

```
<!ATTLIST  OnClick

   ISMID NAME #FIXED "Stimulus"

   reftype CDATA "ResponseChainRef ResponseChain"

   ResponseChainRef IDREF #IMPLIED

   Button NUMBER #IMPLIED

>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

B.4.1.1.3 Parameter Descriptions

Button

The system uses this parameter to indicate which mouse button has been used to execute the click. Possible values are 1 (Left Button), 2 (Right Button), or 4 (Middle Button). The value of this attribute is supplied by the system and is read−only.

B.4.1.2 OnNotationClick

B.4.1.2.1 Semantic Definition

The OnNotationClick stimulus is initiated by a single click of a user's pointing device on a data content notation (DCN) entity represented by an icon in the formatted SGML document. If there are multiple buttons on the pointing device, the left button will initiate this event. The pointing device shall send an OnNotationClick stimulus to a single DCN entity icon at a time.

B.4.1.2.2 Formal Declaration

```
<!element OnNotationClick - O ((%Responses;)*, End?)>



<!ATTLIST  OnNotationClick

   ISMID NAME #FIXED "Stimulus"

   reftype CDATA "ResponseChainRef ResponseChain"

   ResponseChainRef IDREF #IMPLIED

   Notation Name #IMPLIED

   EntityName Name #IMPLIED

   SystemID CDATA #IMPLIED

>
```

B.4.1.2.3 Parameter Descriptions

Notation

The system uses this parameter to indicate the notation name of the DCN entity represented by the icon that received the click. Possible values are any valid SGML name. The value of this attribute is supplied by the system and is read−only.

EntityName

The system uses this parameter to indicate the name of the DCN entity represented by the icon that received the click. Possible values are any valid SGML name. The value of this attribute is supplied by the system and is

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

read−only.

SystemID

The system uses this parameter to indicate the system identifier of the DCN entity represented by the icon that received the click. The value of this attribute is supplied by the system and is read−only.

B.4.1.3 OnObjectClick

B.4.1.3.1 Semantic Definition

The OnObjectClick stimulus is initiated by a single click of a user's pointing device on an object within the CGM graphic. If there are multiple buttons on the pointing device, the left button will initiate this event. The pointing device shall send an OnObjectClick stimulus to a single CGM object at a time.

B.4.1.3.2 Formal Declaration

```
<!element OnObjectClick - O ((%Responses;)*, End?)>


<!ATTLIST  OnObjectClick
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>
```

B.4.2 Object−generated

B.4.2.1 OnResize

B.4.2.1.1 Semantic Definition

Container and content objects generate the OnResize stimulus when a user resizes the object.

B.4.2.1.2 Formal Declaration

```
<!element OnResize - O ((%Responses;)*, end?)>


<!ATTLIST  OnResize
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>
```

B.4.2.2 OnDestroy

B.4.2.2.1 Semantic Definition

Container and content objects generate the OnDestroy stimulus when a user closes the objects.

B.4.2.2.2 Formal Declaration

```
<!element OnDestroy - O ((%Responses;)*, End?)>

<!ATTLIST  OnDestroy
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
   ResponseChainRef IDREF #IMPLIED
>
```

B.4.2.3 OnLoadComplete

B.4.2.3.1 Semantic Definition

A content object generates an OnLoadComplete stimulus when it finishes loading the content specified in the ContentRef property.

B.4.2.3.2 Formal Declaration

```
<!element OnLoadComplete - O ((%Responses;)*, End?)>

<!ATTLIST  OnLoadComplete
   ISMID NAME #FIXED "Stimulus"
   reftype CDATA "ResponseChainRef ResponseChain"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   ResponseChainRef IDREF #IMPLIED

>
```

B.4.2.4 OnEnd

B.4.2.4.1 Semantic Definition

A content object generates an OnEnd stimulus when it finishes playing the content of a video or audio clip.

B.4.2.4.2 Formal Declaration

```
<!element OnEnd - O ((%Responses;)*, End?)>



<!ATTLIST  OnEnd

   ISMID NAME #FIXED "Stimulus"

   reftype CDATA "ResponseChainRef ResponseChain"

   ResponseChainRef IDREF #IMPLIED

>
```

B.5 Properties

B.5.1 Caption

The Caption property is a text string that appears in the TitleBar of a Window or as the text on a button. The Caption is optional. On a window, the caption shall appear only if the TitleBar property value = "TitleBar"

B.5.2 TitleBar

The TitleBar property indicates whether a Window is to be rendered with a title bar. The title bar shall be used to move the Window .

B.5.3 WindowState

The WindowState property allows the author to specify one of the following states for a Window object:

- Normal– Object remains or is reset to a default or last displayed size

- Maximized– Object is set to the maximum sized allowed by the client area of the display

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

- Minimized– Object and its contents are reduced to an icon

B.5.4 FontName

The FontName property defines the name of a font or font family used in the object.

B.5.5 FontSize

Numerical indicator of the font size, in points.

B.5.6 FontBold

The font weight shall be defined as either bold or not bold (i.e., normal).

B.5.7 Visible

The Visible property indicates whether to show or hide a content object.

B.5.8 ActualWidth

This property applies to the Image object only. The ActualWidth property contains a read–only value supplied by the system that indicates the actual width in pixels of the image being displayed.

B.5.9 ActualHeight

This property applies to the Image object only. The ActualHeight property contains a read–only value supplied by the system that indicates the actual height in pixels of the image being displayed.

B.5.10 Left

The Left property sets the position of the object by placing its left edge at the horizontal dimension specified as the value of this property. The value is in pixels from the left edge of the parent container.

B.5.11 LeftRef

This property is used to reference an expression element for the purpose of setting the Left property for this object.

B.5.12 Top

The Top property sets the position of the object by placing its top edge at the vertical dimension specified as the value of this property. The value is in pixels from the top edge of the parent container.

--`,,`-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.5.13 TopRef

This property is used to reference an expression element for the purpose of setting the Top property for this object.

B.5.14 Width

The Width property sets the width of the object to the indicated value, given in pixels.

B.5.15 WidthRef

This property is used to reference an expression element for the purpose of setting the Width property for this object.

B.5.16 Height

The ViewHeight property sets the height of the object to the indicated value, given in pixels.

B.5.17 HeightRef

This property is used to reference an expression element for the purpose of setting the Height property for this object.

B.5.18 Enabled

The Enabled property determines whether a control object is active or inactive.

B.5.19 BevelWidth

The BevelWidth property sets the visual bevel at the edge of a button that makes it appear three dimensional.

B.5.20 ForeColor

The ForeColor property sets the color of captions. Colors are specified as #RRGGBB in hexadecimal.

B.5.21 Playing

This property is used to set a Video or Audio object to a Playing or Not Playing state. When the value is set to Playing, the video or audio clip in the content object will start to play. When the value is set to NPlaying, the video or audio clip in the content object will pause playing. Setting this property has no effect on the current location within the audio or video.

B.5.22 CurrentFrame

This propery specifies the frame number from which the video referenced by the video content object starts

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

playing.

B.5.23 StyleSheet

This property specifies the name of the entity containing the style sheet to be applied to the SGML doucment referenced by the SGML content object.

B.5.24 FillColor

This property specifies the color with which the object will be filled.

B.5.25 DrawMode

This property specifies how the object will be rendered. If value is set to CopyPen, the object is rendered with the fill color solid. If the value is set to MaskPen the object is rendered with the fill color transparent.

B.6 Objects

B.6.1 Container Objects

B.6.1.1 Window

B.6.1.1.1 Semantics

A window object is a general container for grouping content objects, control objects, and other container objects.

B.6.1.1.2 Stimuli

OnResize, OnDestroy

B.6.1.1.3 CreateWindow

```
<!element   CreateWindow  - O (Label*, VariableDeclaration*, (%WindowStimuli;))>


<!attlist CreateWindow
   ISMID NAME #FIXED "CreateContainer"
   ireftype CDATA #FIXED "Parent MID"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   Caption CDATA #IMPLIED
   TitleBar (TitleBar | NTitleBar) "TitleBar"
   WindowState (Normal | Maximized | Minimized) "Normal"
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
   %CreateFontProperties;
>
```

B.6.1.1.4 ModifyWindow

```
<!element   ModifyWindow  - O (Assign*, (%WindowStimuli;))>

<!ATTLIST ModifyWindow
   ISMID NAME #FIXED "ModifyContainer"
   ireftype CDATA #FIXED "Object CreateWindow"
   Object IDREF #REQUIRED
   Caption CDATA #IMPLIED
   TitleBar (TitleBar | NTitleBar) #IMPLIED
   WindowState (Normal | Maximized | Minimized) #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   %ModifyFontProperties;
>
```

B.6.2 Content Objects

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

B.6.2.1 Image

B.6.2.1.1 Semantics

An image object is a content object that contains a graphic.

B.6.2.1.2 Stimuli

OnResize, OnLoadComplete

B.6.2.1.3 Properties

Visible, ActualWidth, ActualHeight, Left, Top, Width, Height

B.6.2.1.4 CreateImage

```
<!element CreateImage - O (Label*, VariableDeclaration*, Expression*, (%ImageStimuli;))>


<!ATTLIST CreateImage
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   ID ID #REQUIRED
   Visible (Visible | NVisible) "Visible"
   ActualWidth NUMBER #IMPLIED
   ActualHeight NUMBER #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

B.6.2.1.5 ModifyImage

```
<!element ModifyImage - O (Assign*, Expression*, (%ImageStimuli;))>


<!ATTLIST ModifyImage
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   valueref CDATA "SemanticContent SemanticContent"
   ireftype CDATA #FIXED "Object CreateImage Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "ignore"
   Object IDREF #REQUIRED
   ActualWidth NUMBER #IMPLIED
   ActualHeight NUMBER #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
>
```

B.6.2.2 Video

B.6.2.2.1 Semantics

An video object is a content object that contains a video.

B.6.2.2.2 Stimuli

OnEnd

B.6.2.2.3 Properties

Playing, CurrentFrame, ActualHeight, ActualWidth, Height, Width, Top, Left

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.6.2.2.4 CreateVideo

```
<!element CreateVideo – O (Label*, VariableDeclaration*, Expression*, (%VideoStimuli;))>


<!ATTLIST CreateVideo
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "error"
   ID ID #REQUIRED
   Playing (Playing | NPlaying) "NPlaying"
   CurrentFrame NUMBER "0"
   ActualHeight NUMBER #IMPLIED
   ActualWidth NUMBER #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>
```

B.6.2.2.5 ModifyVideo

```
<!element ModifyVideo – O (Assign*,  Expression*, (%VideoStimuli;))>


<!ATTLIST  ModifyVideo
    ISMID  NAME  #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateVideo Parent CreateWindow"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   Object IDREF #REQUIRED

   Parent IDREF #REQUIRED

   ObjectNotExist (error | ignore) "error"

   Playing (Playing | NPlaying) #IMPLIED

   CurrentFrame NUMBER #IMPLIED   Left  CDATA #IMPLIED

   Top  CDATA #IMPLIED

   Width  CDATA #IMPLIED

   Height  CDATA #IMPLIED

>
```

B.6.2.3 Audio

B.6.2.3.1 Semantics

An audio object is a content object that contains an audio clip.

B.6.2.3.2 Stimuli

OnEnd

B.6.2.3.3 Properties

Playing, Left, Top

B.6.2.3.4 CreateAudio

```
<!element CreateAudio – O (Label*, VariableDeclaration*, Expression*, (%AudioStimuli;))>


<!ATTLIST CreateAudio

   ISMID NAME #FIXED "CreateContentObject"

   SemanticContent IDREF #IMPLIED

   ireftype CDATA #FIXED "Parent CreateWindow"

   Parent IDREF #REQUIRED

   ObjectNotExist (error | ignore) "error"

   ID ID #REQUIRED

   Playing (Playing | NPlaying)
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   Left  CDATA #REQUIRED

   Top  CDATA #REQUIRED

>
```

B.6.2.3.5 ModifyAudio

```
<!element ModifyAudio - O (Assign*, Expression*, (%AudioStimuli;))>


<!ATTLIST ModifyAudio

   ISMID NAME #FIXED "ModifyContentObject"

   SemanticContent IDREF #IMPLIED

   ireftype CDATA #FIXED "Object CreateAudio Parent CreateWindow"

   Object IDREF #REQUIRED

   Parent IDREF #REQUIRED

   ObjectNotExist (error | ignore) "error"

   Playing (Playing | NPlaying) #IMPLIED

   Left  CDATA #IMPLIED

   Top  CDATA #IMPLIED

>
```

B.6.2.4 SGML

B.6.2.4.1 Semantics

An SGML object is a content object that contains an SGML document.

B.6.2.4.2 Stimuli

OnNotationClick

B.6.2.4.3 Properties

StyleSheet, Left, Top, Width, Height

B.6.2.4.4 CreateSGML

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
<!element CreateSGML - O (Label*, VariableDeclaration*, Expression*, (%SGMLStimuli;))>


<!ATTLIST CreateSGML
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "error"
   ID ID #REQUIRED
   StyleSheet ENTITY #IMPLIED
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>
```

B.6.2.4.5 ModifySGML

```
<!element ModifySGML - O (Assign*, Expression*, (%SGMLStimuli;))>


<!ATTLIST ModifySGML
   ISMID NAME #FIXED "ModifyContentObject"

   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateSGML Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "error"
   ActualWidth NUMBER #IMPLIED
   ActualHeight CDATA #IMPLIED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   %ModifyFontProperties;
>
```

B.6.2.5 CGM

B.6.2.5.1 Semantics

An CGM object is a content object that contains a Computer Graphics Metafile image.

B.6.2.5.2 Stimuli

OnObjectClick, OnLoadComplete

B.6.2.5.3 Properties

Visible, ActualWidth, ActualHeight, Left, Top, Width, Height

B.6.2.5.4 CreateCGM

```
<!element CreateCGM - O (Label*, VariableDeclaration*, Expression*, (%CGMStimuli;))>

<!ATTLIST CreateCGM
   ISMID NAME #FIXED "CreateContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Parent CreateWindow"
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "error"
   ID ID #REQUIRED
   Visible (Visible | NVisible) "Visible"
   ActualWidth NUMBER #IMPLIED
   ActualHeight CDATA #IMPLIED
   Left  CDATA #REQUIRED
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
>
```

B.6.2.5.5 ModifyCGM

```
<!element ModifyCGM - O (Assign*, Expression*, (%CGMStimuli;))>

<!ATTLIST ModifyCGM
   ISMID NAME #FIXED "ModifyContentObject"
   SemanticContent IDREF #IMPLIED
   ireftype CDATA #FIXED "Object CreateCGM Parent CreateWindow"
   Object IDREF #REQUIRED
   Parent IDREF #REQUIRED
   ObjectNotExist (error | ignore) "error"
   ActualWidth NUMBER #IMPLIED
   ActualHeight CDATA #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
>
```

B.6.3 Control Objects

B.6.3.1 Button

B.6.3.1.1 Semantics

A Button is a control object used to initiate a stimulus.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

B.6.3.1.2 Stimuli

OnClick

B.6.3.1.3 Properties

Enabled, Visible, BevelWidth, Caption, ForeColor, Left, LeftRef, Top, TopRef, Width, WidthRef, Height, HeightRef, FontName, FontSize, FontBold

B.6.3.1.4 CreateButton

```
<!element CreateButton - O (Label*, VariableDeclaration*, OnClick)?)>



<!ATTLIST CreateButton

   ISMID NAME #FIXED "CreateControlObject"

   ireftype CDATA #FIXED "Parent CreateWindow LeftRef Expression

                          TopRef Expression WidthRef Expression

                          HeightRef Expression"

   Parent IDREF #REQUIRED

   ObjectNotExist (error | ignore) "ignore"

   ID ID #REQUIRED

   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef

                          Height HeightRef"

   Enabled (Enabled | NEnabled) "Enabled"

   Visible (Visible | NVisible) "Visible"

   BevelWidth NUMBER "2"

   Caption CDATA #IMPLIED

   ForeColor CDATA "000000"

   Left  NUMBER #REQUIRED

   Top  CDATA #REQUIRED

   Width  CDATA #REQUIRED

   Height  CDATA #REQUIRED

   WidthRef IDREF #IMPLIED
```

--`,,`-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
   TopRef IDREF #IMPLIED
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
   %CreateFontProperties;
>
```

B.6.3.1.5 ModifyButton

```
<!element ModifyButton - O (Assign*, %ButtonStimuli;))>

<!ATTLIST ModifyButton
   ISMID NAME #FIXED "ModifyControl"
   ireftype CDATA #FIXED "Object CreateButton Parent CreateWindow
                          LeftRef Expression TopRef Expression
                          WidthRef Expression HeightRef Expression"
   Object IDREF #REQUIRED
   Parent IDREF #IMPLIED
   ObjectNotExist (error | ignore) "ignore"
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
                          Height HeightRef"
   Enabled (Enabled | NEnabled) #IMPLIED
   Visible (Visible | NVisible) #IMPLIED
   BevelWidth NUMBER #IMPLIED
   Caption CDATA #IMPLIED
   ForeColor CDATA #IMPLIED
   Left  CDATA #IMPLIED
   Top  CDATA #IMPLIED
   Width  CDATA #IMPLIED
   Height  CDATA #IMPLIED
   WidthRef IDREF #IMPLIED
```


--`,,,`-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   TopRef IDREF #IMPLIED

   LeftRef IDREF #IMPLIED

   HeightRef IDREF #IMPLIED

   %ModifyFontProperties;

>
```

B.6.3.2 HotSpot

B.6.3.2.1 Semantics

A Hotspot is a coordinate zone that overlays part of an image, identifying an object on the image.

B.6.3.2.2 Stimuli

OnClick

B.6.3.2.3 Properties

Enabled, Visible, Caption, ForeColor, FillColor, DrawMode, Left, LeftRef, Top, TopRef, Width, WidthRef, Height, HeightRef, FontName, FontSize, FontBold

B.6.3.2.4 CreateHotSpot

```
<!element CreateHotSpot - O (Label*, VariableDeclaration*, Expression*, (%HotSpotStimuli;)*)>



<!ATTLIST CreateHotSpot

   ISMID NAME #FIXED "CreateControlObject"

   ireftype CDATA #FIXED "Parent CreateImage LeftRef Expression

                          TopRef Expression WidthRef Expression

                          HeightRef Expression"

   Parent IDREF #REQUIRED

   ObjectNotExist (error | ignore) "error"

   ID ID #REQUIRED

   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef

                          Height HeightRef"

   Enabled (Enabled | NEnabled) "Enabled"
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
   Visible (Visible | NVisible) "Visible"
   Caption CDATA #IMPLIED
   ForeColor CDATA "000000"
   FillColor  CDATA "000000"
   DrawMode (CopyPen | MaskPen) "MaskPen"
   Left  CDATA #REQUIRED
   Top  CDATA #REQUIRED
   Width  CDATA #REQUIRED
   Height  CDATA #REQUIRED
   WidthRef IDREF #IMPLIED
   TopRef IDREF #IMPLIED
   LeftRef IDREF #IMPLIED
   HeightRef IDREF #IMPLIED
   %CreateFontProperties;
>
```

B.6.3.2.5 ModifyHotSpot

```
<!element ModifyHotSpot - O (Assign*, (%HotSpotStimuli;)*)>

<!ATTLIST
  ModifyHotSpot
   ISMID NAME #FIXED "ModifyControl"
   ireftype CDATA #FIXED "Object CreateHotspot Parent CreateImage
                          LeftRef Expression TopRef Expression
                          WidthRef Expression HeightRef Expression"
   Object IDREF #REQUIRED
   Parent IDREF #IMPLIED
   ObjectNotExist (error | ignore) "error"
   valueref CDATA #FIXED "Left LeftRef Top TopRef Width WidthRef
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                        Height HeightRef"
    Enabled (Enabled | NEnabled) #IMPLIED
    Visible (Visible | NVisible) #IMPLIED
    BevelWidth NUMBER #IMPLIED
    Caption CDATA #IMPLIED
    ForeColor CDATA #IMPLIED
    FillColor  CDATA #IMPLIED
    DrawMode (CopyPen | MaskPen) #IMPLIED
    Left  CDATA #IMPLIED
    Top  CDATA #IMPLIED
    Width  CDATA #IMPLIED
    Height  CDATA #IMPLIED
    WidthRef IDREF #IMPLIED
    TopRef IDREF #IMPLIED
    LeftRef IDREF #IMPLIED
    HeightRef IDREF #IMPLIED
    %ModifyFontProperties;
>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale