INTERNATIONAL STANDARD

IEC 62298-4

First edition
2005-10

TeleWeb application –

Part 4:
Hyperteletext profile

Reference number
IEC 62298-4:2005(E)

Publication numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series. For example, IEC 34-1 is now referred to as IEC 60034-1.

Consolidated editions

The IEC is now publishing consolidated versions of its publications. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Further information on IEC publications

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology. Information relating to this publication, including its validity, is available in the IEC Catalogue of publications (see below) in addition to new editions, amendments and corrigenda. Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is also available from the following:

- **IEC Web Site (www.iec.ch)**

- **Catalogue of IEC publications**

  The on-line catalogue on the IEC web site (www.iec.ch/searchpub) enables you to search by a variety of criteria including text searches, technical committees and date of publication. On-line information is also available on recently issued publications, withdrawn and replaced publications, as well as corrigenda.

- **IEC Just Published**

  This summary of recently issued publications (www.iec.ch/online_news/ justpub) is also available by email. Please contact the Customer Service Centre (see below) for further information.

- **Customer Service Centre**

  If you have any questions regarding this publication or need further assistance, please contact the Customer Service Centre:

  Email: custserv@iec.ch
  Tel: +41 22 919 02 11
  Fax: +41 22 919 03 00

INTERNATIONAL STANDARD

IEC 62298-4

First edition
2005-10

TeleWeb application –

Part 4:
Hyperteletext profile

International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland
Telephone: +41 22 919 02 11 Telefax: +41 22 919 03 00 E-mail: inmail@iec.ch Web: www.iec.ch

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

PRICE CODE W

For price, see current catalogue

CONTENTS

FOREWORD 3
INTRODUCTION 5

1 Scope 6
2 Normative references 6
3 Terms, definitions and abbreviations 7
3.1 Terms and definitions 7
3.2 Abbreviations 7
4 Mandatory Superteletext profile features 8
4.1 Font issues 8
4.2 Dithering 8
4.3 TeleWeb EPG 8
4.4 Cross-linking between services 8
4.5 Memory requirements 8
5 General display-related extensions 9
5.1 Downloadable fonts 9
5.2 Support for non-Latin alphabets 9
5.3 Page size and page scrolling 9
6 Hypertext extensions 9
6.1 HTML frames 10
6.2 Forms 16
6.3 Internationalization 25
6.4 <FONT> tag 26
6.5 <TIME> tag 27
7 User profiles 29
8 ISP interface 29
8.1 The ISP special function 29
8.2 Network congestion control 31
9 Device ID (OPTIONAL) 33
10 Cross-linking and context-switching 33
10.1 Additional URI schemes 33
11 Audio clips (OPTIONAL) 33
12 Style sheets 34
13 Scripting 34
13.1 ECMAScript 34
13.2 <SCRIPT> tag 35
13.3 Events 35
13.4 Document object model (DOM) 35
14 Additional content types 35
14.1 Graphic files 35
14.2 Text files 35

Annex A (informative) Security considerations 36

INTERNATIONAL ELECTROTECHNICAL COMMISSION

TELEWEB APPLICATION –

Part 4: Hyperteletext profile

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62298-4 has been prepared by IEC technical committee 100: Audio, video and multimedia systems and equipment.

This standard cancels and replaces IEC/PAS 62298 published in 2002.

The text of this standard is based on the following documents:

FDIS	Report on voting
100/1000/FDIS	100/1023/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

IEC 62298 consists of the following parts, under the general title *TeleWeb applications*:

Part 1: General description

Part 2: Delivery methods

Part 3: Superteletext profile

Part 4: Hyperteletext profile

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

INTRODUCTION

The aim of TeleWeb is to deliver World Wide Web-style content to the living room TV to give the viewer an enhanced television experience. A TeleWeb service broadcasts data files containing text and high-definition graphics to suitable decoders. The data transmitted can be closely linked to events within the accompanying TV programmes, or can be more general in nature to emulate a traditional, but higher definition, super teletext service. Different profiles are defined.

It is the intention that TV-based decoders can be implemented in a cost-effective manner without recourse to the technology normally associated with personal computers. In part, this is achieved by limiting the number of different types of multimedia data that can be used within a service. By careful design of the user interface, decoder manufacturers will be able to offer easy-to-use equipment for accessing TeleWeb services without requiring the consumer to be computer-literate. In addition, they will be able to customize their products to differentiate them from those of their competitors.

This document specifies the TeleWeb Hyperteletext profile and focuses on the presentation layer especially the implementation of TeleWeb HTML and scripting. It further defines the graphical requirements like fonts and the content formats used.

TELEWEB APPLICATION –

Part 4: Hyperteletext profile

1 Scope

This part of IEC 62298 specifies the TeleWeb Hyperteletext profile that allows Web-style text and graphics to be displayed on suitable decoders. A TeleWeb service comprises multimedia data files whose format and attributes are defined by this specification.

This standard is backwards compatible with IEC 62298-3 and extends it with features like scripting and style-sheets. The graphical capability is extended with features like frames and forms. For information regarding general information and the transport layer, refer to IEC 62298-1 and IEC 62298-2.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 62298-1: *TeleWeb application – Part 1: General description*

IEC 62298-2: *TeleWeb application – Part 2: Delivery methods*

IEC 62298-3: *TeleWeb application – Part 3: Superteletext profile*

ISO/IEC 11172-3:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 3: Audio*

ISO/IEC 14496-3:2001, *Information technology – Coding of audio-visual objects – Part 3: Audio*

ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*

ETSI EN 300 468, *Digital Video Broadcasting (DVB) – Specification for Service Information (SI) in DVB systems*

W3C Recommendation, *Cascading Style Sheets, level 1 (CSS1)*

W3C Recommendation, *HyperText Markup Language, version 4.0*

SMPTE 363M:2002, *Television – Declarative Data Essence – Content Level 1*

SMPTE 366M:2002, *Television – Document Object Model Level 0 (DOM-0) and Related Object Environment*

IETF RFC 2046, *Multipurpose Internet Mail Extensions (MIME) – Part Two: Media types*

PFR v1.2, *Bitstream Inc. Coding of Outline Fonts – PFR Specification, version 1.2*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

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

3.1.1
button
part of the user interface that enables the viewer to select a page or trigger an event, etc. It may not necessarily exist as a physical button on a remote control handset

3.1.2
CDATA
character data in an HTML document. Character entities and HTML mark-up is not recognized

3.1.3
conditional access (CA)
mechanism by which user access to service components can be restricted

3.1.4
PCDATA
parsed character data in an HTML document. Character entities (numeric and named entities) as well as HTML mark-up is recognized in the data

3.2 Abbreviations

CATV	Cable TV
CRC	Cyclic Redundancy Check
DECT	Digital European Cordless Telecommunications
DTD	Document Type Definition
DVB	Digital Video Broadcasting
GIF	Graphics Interchange Format
GSM	Global System for Mobile Communication
HTML	Hyper Text Mark-up Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Security
IDL	Interface Definition Language
ISDN	Integrated Services Digital Network
LMDS	Local Multipoint Distribution Service
MJD	Modified Julian Date
PSTN	Plain Old Telephone System
RFC	Internet Request for Comment
SMATV	Satellite Master Antenna Television
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTC	Universal Time Coordinated
WWW	World Wide Web

4 Mandatory Superteletext profile features

This clause identifies the features listed as optional in IEC 62298-3, which are made mandatory for receivers conforming to this specification. All other optional features of IEC 62298-3 remain optional in this profile.

4.1 Font issues

Support of "bold", "italics" and "bold italics" is mandatory for all Latin fonts whether fixed or proportional, resident or downloaded.

4.2 Dithering

If the content is authored without reference to the TeleWeb default colour palette defined in IEC 62298-3, and a decoder is unable to reproduce a colour faithfully, the decoder shall employ a dithering algorithm and shall not use colour-matching techniques.

NOTE Examples of such algorithms are ordered dithering or error-diffusion algorithms (for example, Floyd-Steinberg).

4.3 TeleWeb EPG

A TeleWeb EPG as described in IEC 62298-3, Clause 5, is mandatory in the decoder.

4.4 Cross-linking between services

Cross-linking refers to the ability to navigate between different TeleWeb services or to Teletext services.

A decoder shall support cross-linking between any TeleWeb services through absolute TeleWeb URLs defined in IEC 62298-3. It is up to the device to decide if user activation of a cross-link will result in a channel switch.

It is good practice to inform the user of the consequences of a possible re-tuning.

The usefulness of cross-linking is ultimately a question of resources in the device. A larger memory capacity and an extra tuner will significantly enhance the cross-linking capability of the device and therefore the user experience.

In a DVB-compliant device cross-linking is more useful than in an analogue device, since cross-linking within the same multiplex does not result in re-tuning. The same holds for cross-linking between short and full TeleWeb services of the same channel.

Code of Practice: if the service is not available in the memory, the decoder can try to retrieve the content through the return channel.

4.5 Memory requirements

The broadcast size for a full service according to this profile shall not exceed 9,8 Mbytes, including up to 4,9 Mbytes of Superteletext profile data.

The broadcast size for a short service according to this profile shall not exceed 0,2 Mbytes, including up to 0,1 Mbytes of Superteletext profile data. The maximum transmission cycle time for the short service shall remain at 60 s.

A device according to this profile shall support at least one full TeleWeb service and one short TeleWeb service providing storage for the specified broadcast sizes (9,8 + 0,2 Mbytes). If a full service and/or a short service exceed the indicated cache sizes, the behaviour of the decoder is not covered by this specification.

A device according to this profile must provide some persistent storage to store at least the following parameters: user group Id, decoder configuration data, user profiles, ISP information. Decoder configuration data might include channel presets, preferred TeleWeb provider, etc.

NOTE The term broadcast size refers to the data transmitted and may include compressed data.

5 General display-related extensions

5.1 Downloadable fonts

In order to support fonts other than the resident one(s) and character sets other than Latin-1, this profile shall support downloadable fonts. Downloadable fonts can also be used to gain access to true bold or italic styles. The requirements are as follows.

- Fonts shall be bitmap fonts.
- Unicode character encoding shall be used.
- Replacing a built-in font with a downloaded font shall be supported.
- Font data shall not be encrypted.
- Font data may be compressed through methods allowed by this profile.
- Only the used characters shall be downloaded (provided by the service provider).
- Supplementing a built-in or downloaded font shall be supported.

Downloadable fonts shall use the file format specified in PFR V1.2.

5.2 Support for non-Latin alphabets

HTML documents can specify the character set to use for rendering them, by using the META tag. See 6.3.1. This data shall be copied to the http header.

5.3 Page size and page scrolling

Horizontal scrolling shall not be supported.

Vertical scrolling must be limited to 1 440 pixels vertically (3×480).

The choice to limit vertical scrolling to three pages greatly simplifies its implementation.

The implementation details of vertical scrolling is at the discretion of the decoder manufacture, hence it is up to the manufacturer to decide whether scrolling is done one line at a time or a display size at a time, etc.

6 Hypertext extensions

A decoder shall parse the full syntax of the 3 DTDs of HTML 4.0, as extended by IEC 62298-3. Unless specified in the HTML document metadata, the default DTD shall be “transitional” plus the extensions of IEC 62298-3. Certain constraints are imposed as further defined in this clause.

6.1 HTML frames

A decoder shall support HTML frames.

A restriction is placed on the way the content is authored. Each direct selection key (colour keys and digit keys) shall be referenced only once within all the frames displayed at any one time. If the same key is referenced more than once, the behaviour of the decoder is not specified by this document.

6.1.1 Cursor navigation within frames

HTML frames introduce a navigational problem on a device without a freely moving cursor. To overcome these problems, navigational rules must be enacted. Those rules are described in this clause.

The following general rules apply to the use of frames.

- Only one frameset per page is allowed. The maximum number of frames in the frameset shall not exceed four. If these rules are broken, the behaviour of the decoder is no longer specified.
- Only one of these frames can have a vertical scrollable area. This frame is further referenced as the main frame.
- The whole content presented by the frameset shall not host more than one instance of each direct selection link (0..9, colour links). In case this happens, the behaviour of the decoder is no longer covered by this specification.
- When the frameset is rendered the first time, the cursor is always positioned within the main frame. If not overruled by cursor control, the cursor will be positioned on the link closest to the upper left corner of the main frame.

For navigating, the whole frameset shall be seen as one screen in which the cursor navigates over the frame borders, which may be invisible to the user. The result of this behaviour removes the need for frame selection buttons on the remote. Two main navigation behaviours can be identified: “Navigation in a frame without scrolling content” and “Navigation in a frame with scrollable content, the main frame”.

The cursor position wrap-around at the borders of the screen is an option of the decoder manufacturer.

The way in which scrolling is implemented is defined by the decoder manufacturer. It can be a smooth pixel scroll or a more jumpy scroll moving bigger parts of the content (for example, one-quarter screen) at once.

6.1.1.1 Navigation in a frame without scrolling content

The navigation of such a frame is fully compliant to the navigation defined for a screen content without frames, the only difference being the size of the content, which will be at the most 640 × 480. Upon reaching the border of a frame, the cursor will cross the border to the adjacent frame.

6.1.1.2 Navigation in a frame with scrollable content, the main frame

The horizontal navigation does not differ and is identical to the navigation described in 6.1.1.1.

In the case of vertical navigation, the frame boundary is only crossed at the borders of the content, not the frame! This means that when the cursor is moving down, the content in the frame is moved up until the bottom of the content is fully visible, then, and only then, is it allowed to cross the border to the adjacent frame below. The same rule applies if the cursor is moving up. If, during the scroll operation, all links within the frame become invisible, the

decoder manufacturer shall implement one of the following cursor behaviours to enable further navigation, until a link within the scrolled content becomes visible again or the end of the content is reached.

- The cursor embraces the whole frame until a link becomes visible, after which the cursor moves again in focus of the visible link. While the cursor embraces the whole frame, the user can scroll up or down using the vertical navigation keys. If needed, the cursor can be rendered in another colour to indicate this special behaviour. Due to the fact that the cursor is not really indicating a link, it cannot be selected.
- The cursor moves to a parking spot allowing the user to further scroll through the content until a link becomes visible again. The parking spot can be any graphics defined by the decoder manufacturer. The position of the parking spot can be outside the content area or overlaying the content area. If the parking spot is overlaying the content area, the user should be able to read the covered content by scrolling up or down. If needed, the cursor can be rendered in another colour to indicate this special behaviour. Due to the fact that the cursor is not really indicating a link, it cannot be selected.

If the cursor is moved horizontally to an adjacent frame, when no links are visible within the current frame, the link best in line with the navigation direction shall be selected.

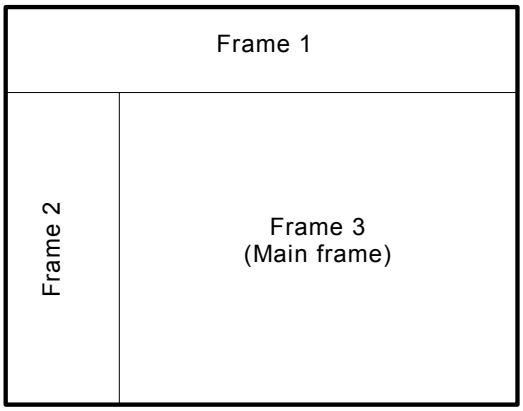
Moving the cursor from a frame to the main frame with scrollable content can have different results, depending on the current visible content of the main frame.

- If links are available, the link best in line to the navigation direction shall be selected.
- If no links are available, the cursor will embrace the frame or move to the parking spot depending on the decoder implementation. From here on, the user can navigate through the scrollable content of the frame.
- The content is, however, not re-rendered to the top of the content when moving into the main frame from an adjacent frame.

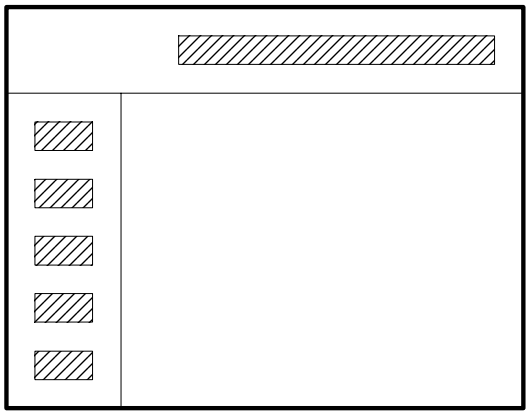
Moving the cursor from a frame to the main frame, which contains neither links nor scrollable content, is not possible.

6.1.1.2.1 Clarifying example of navigation in a frame with scrollable content

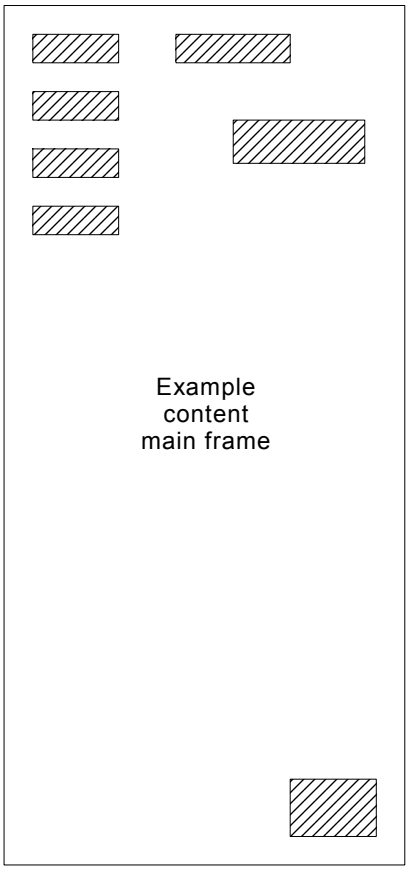
Start from the following frameset. Frame 1 and 2 have the links indicated by the shaded areas in Figure 1. In Figures 1 and 2 below, the thick line around the link (shaded) shows the current location of the cursor.



Frameset layout example



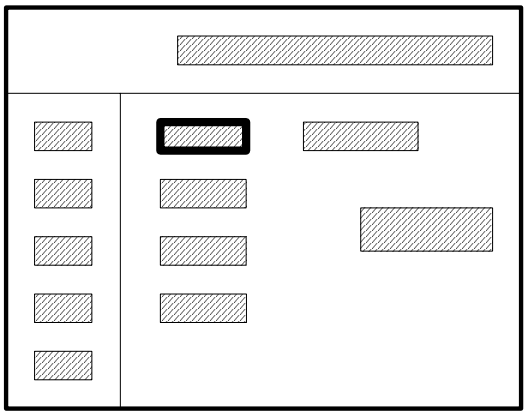
Frame 1 and 2 with links



Example content for frame 3

The “example content” for the main frame (frame 3) has links on the top of the content and one link at the bottom. In between, there is an area (bigger than the size of frame 3) without any links. If the user is scrolling the content in frame 3, there can be a situation where no links in the content are visible.

Initially rendering the frameset results in the screen layout below.

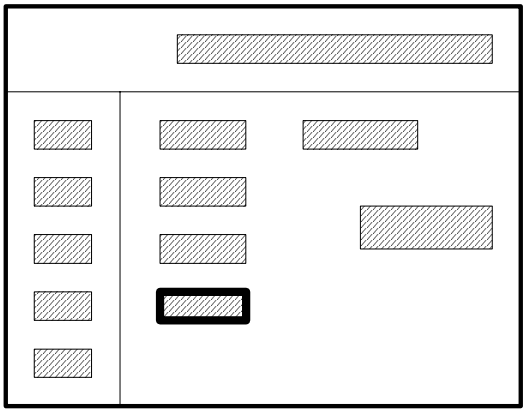


Initial rendered screen

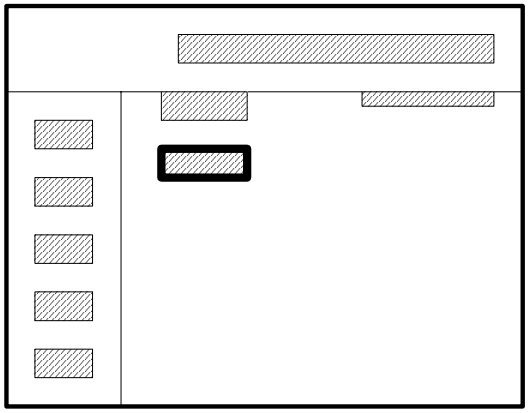
IEC 1731/05

Figure 1 – Navigation in a frame with scrollable content, part 1

When the cursor is moved down it will initially move within the frame to the available links below. It is up to the decoder manufacturer if scrolling occurs at this early stage. However, at some moment, trying to move the cursor further down will result in a scroll-up of the content, revealing more available content as shown in Figure 2.

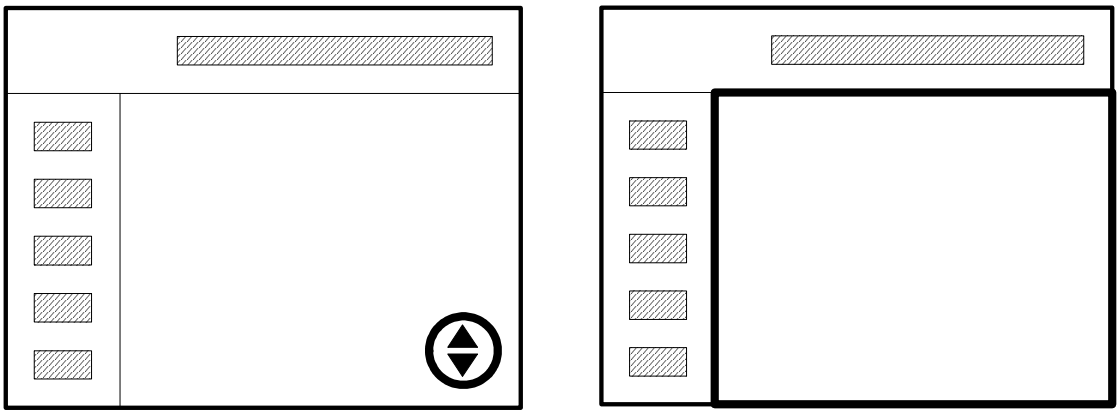


Moving cursor down, down, down …



Content starts scrolling …

When scrolling has made all links within the frame invisible, the cursor moves to a parking spot or embraces the main frame, depending on the decoder implementation. Embracing indicates the unavailability of links within the current view but permits scrolling the content.



Moving the cursor further down scrolls the content up until the link at the bottom of the page becomes visible again. It is up to the implementation to define when a link is selected again, either when it is partially or fully visible. The focus of the cursor now moves again to a visible link within the content. The cursor parking spot is removed.

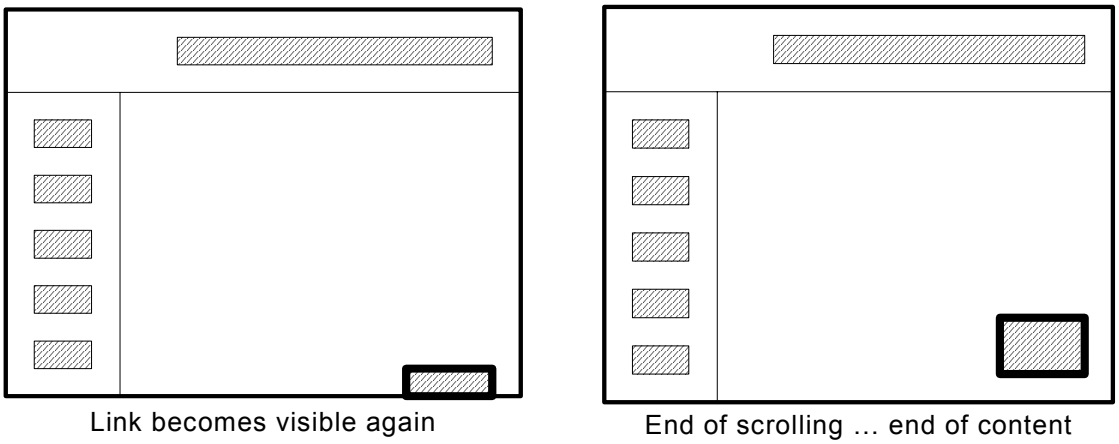


Figure 2 – Navigation in a frame with scrollable content, part 2

6.1.2 <FRAMESET> frameset definition tag

Function: This element is used to define the organization of a set of independent window regions known as frames, as defined by the <FRAME> element. This element replaces the <BODY> element in the framing documents.

Format:

<FRAMESET
BORDER = “border thickness”
BORDERCOLOR = “colour name” or #RRGGBB
FRAMEBORDER = “NO” or “YES” or “0” or “1”
FRAMESPACING = “pixels”
COLS = “list of lengths”
ROWS = “list of lengths”
TITLE = “advisory title”>
<FRAME> elements and <NOFRAME>
</FRAMESET>

Attributes supported: BORDER, BORDERCOLOR, FRAMEBORDER, FRAMESPACING, COLS, ROWS, TITLE

Use: Frames allow updates to small portions of a page while leaving the rest of the display unchanged. This is transmission efficient, as only those parts of the display that change need to be broadcast.

A potential application for frames is in the context of an EPG service. NexTView, for example, defines a screen template with three specific display areas. Two are fixed (the header and the message areas), the third is a scrollable area for the programme listing. This styling can be realized easily with three frames, one for each of these areas. Consequently, it is possible to scroll a long programme listing and at the same time display a fixed header and a message, for example, an advertisement. Without the use of frames, the advertisement would move out of sight as soon as the user scrolls down the listing.

Example:

```
<!--This example defines a frame set of three columns. The middle column is 50 pixels wide.
    The first and last columns fill the remaining space. The last column takes twice as much
    space as the first. -->
<FRAMESET COLS="*,50,*">
    <FRAME SRC="column1.htm>
    <FRAME SRC="column2.htm>
    <FRAME SRC="column3.htm>
</FRAMESET>
<!--This example defines a frame set of two columns, one of which is 20 % of the screen and
    the other 80 % -->
<FRAMESET COLS="20%,80%">
    <FRAME SRC="controls.htm">
    <FRAME SRC="display.htm">
</FRAMESET>
```

BORDER attribute

This attribute sets the width (in pixels) of frame borders within the frame set. Setting BORDER="0" eliminates all frame borders.

BORDERCOLOR attribute

This attribute sets the colour for frame borders within the frame set using either a named colour or a colour specified in the #RRGGBB format.

COLS attribute

This attribute contains a comma-delimited list to specify the number and size of the columns contained within a set of frames. The column definitions appear in the list in the display order left to right. Individual column sizes can be specified in one of three formats – a fixed width (in pixels), a percentage of the available width, and by setting the value to * which has the effect of allowing the column to expand to fill the space available.

FRAMEBORDER attribute

This attribute controls whether or not frame borders should be displayed. Values 0, NO, 1 and YES are supported.

FRAMESPACING attribute

This attribute indicates the space between frames (in pixels).

ROWS attribute

This attribute contains a comma-delimited list to specify the number and size of the rows contained within a set of frames. The number of entries in the list indicates the number of rows. Row size is specified in the same formats as used for column width.

6.1.3 <FRAME> window region

Function: This attribute defines a named window region, known as a frame, which can independently display its own content.

Format:

<FRAME
BORDERCOLOR =”colour name” or #RRGGBB
FRAMEBORDER = “NO” or “YES” or “0” or “1”
MARGINHEIGHT = “pixels”
MARGINWIDTH = “pixels”
NAME = “string”
SRC = “URL of frame contents”>

Attributes supported: BORDERCOLOR, FRAMEBORDER, MARGINHEIGHT, MARGIN WIDTH, NAME, SRC.

NOTE HTML V4.0 attributes NORESIZE; SCROLLING and LONGDESC are not supported.

Use: See <FRAMESET> element. Frames cannot be resized.

Example:

```
<FRAMESET ROWS=”20 %,80 %”>
    <FRAME SRC=”controls.htm” NAME=”controls”>
    <FRAME SRC=”content.htm>
</FRAMESET>
```

produces two rows containing the documents controls.htm and content.htm. The first document will cover 20 % of height of the frameset, the second document will cover the rest.

BORDERCOLOR attribute

This attribute sets the colour of the frame’s border using either a named colour or a colour specified in #RRGGBB format.

FRAMEBORDER attribute

This attribute determines whether an outlined three-dimensional border surrounds the frame. YES or 1 switches the border on, NO or 0 switches the border off.

MARGINHEIGHT attribute

This attribute sets the distance (in pixels) between the content of the frame and its top and bottom borders.

MARGINWIDTH attribute

This attribute sets the distance (in pixels) between the content of the frame and its left and right borders.

NAME attribute

This attribute assigns a name to the frame so that it can be the target destination of hyperlinks.

SRC attribute

This attribute contains the URL of the contents to be displayed in the frame. If this attribute is absent, nothing will be loaded into the frame.

6.1.4 <A> anchor tag

Function: see definition in the Superteletext profile given in IEC 62298-3.

Additional attribute supported: TARGET.

Example:

<A HREF="webpage.html#Interesting section" TARGET=Interesting>Go to Interesting Section</A>

produces a "Go to Interesting Section" link in the sidebar.

TARGET attribute

This attribute specifies the frame where the document should be displayed.

This attribute can only hold a valid frame name (existing in this frame set) or the keywords "_SELF" and "_TOP".

_SELF means that the HTML document is displayed in the frame where this anchor tag is contained.

_TOP means that the whole frame set is replaced by the new HTML document.

6.1.5 <AREA> area tag

Function: see definition in the Superteletext profile given in IEC 62298-3.

Additional attribute supported: TARGET.

Example:

<AREA HREF="webpage.html" TARGET=frame1 SHAPE=rect COORDS="50,50,100,100">

produces a link, part of an image map, that replaces the content of frame1 when selected.

TARGET attribute

This attribute specifies the frame where the document should be displayed.

This attribute can only hold a valid frame name (existing in this frame set) or the keywords "_SELF" and "_TOP".

_SELF means that the HTML document is displayed in the frame where this anchor tag is contained.

_TOP means that the whole frame set is replaced by the new HTML document.

6.2 Forms

A decoder shall support HTML forms. This profile needs forms, because of its support for a return channel. A form is mostly used to define a user interface for interaction over the return channel. The form will ask the user to input some data, which will then be extracted and sent to a server at the other end of the return channel.

6.2.1 <FORM> tag

Function: This element is used to define a user interface for interaction over the return channel. Together with the FORM data the DeviceID can be sent to the server (see clause 9).

Format:

<FORM
ACTION = URL
ENCTYPE = “application/x-www-form-urlencoded”
METHOD = “GET”
NUMRESPONSETHRESHOLD=“0”, …,“65535”
RESPONSESTARTTIME=start-time
RESPONSESTOPTIME=stop-time
RETRYWAITTIME=wait-time
RESPONSEATTEMPTLIMIT=“0”, …,“255” (“0” means unrestricted) >
… Form Content …
</FORM>

Attributes supported: ACTION, ENCTYPE, METHOD, NUMRESPONSETHRESHOLD, RESPONSESTARTTIME, RESPONSESTOPTIME, RETRYWAITTIME, RESPONSEATTEMPT LIMIT.

Example:

<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl"
NUMRESPONSETHRESHOLD="32768"
RESPONSESTARTTIME="0×C079120000"
RESPONSESTOPTIME="0×C079130000"
RETRYWAITTIME="0×0000000005"
RESPONSEATTEMPTLIMIT="10" >
name: <INPUT NAME=realname>
email: <INPUT NAME=email>
<INPUT TYPE=SUBMIT>
</FORM>

produces a form with two single-line-text-input-elements and a submit-button.

ACTION attribute

This attribute specifies the URL that will process the form. Default is the current URL. Only the HTTP and HTTPS protocols are supported. Using these protocols the scheme of the URL specifies whether or not to establish a secure connection when submitting the form to the server.

Example:

Normal connection: <FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
Secure connection: <FORM ACTION="https://www.idocs.com/cgi-bin/mycgi.pl">

ENCTYPE attribute

This attribute specifies the encoding of the form values sent to the server when the method is POST.

METHOD attribute

This attribute specifies which HTTP method will be used to submit the form data to the server.

NUMRESPONSESTHRESHOLD attribute

This attribute defines the threshold above which the decoder may initiate a call to a service provider. The value is a number in the range 0 and 65535. The decoder shall compare this value with an internally generated 16-bit unsigned integer random number. This attribute specifies which HTTP method will be used to submit the form data to the server. See Subclause 8.2 about congestion control.

RESPONSESTARTTIME attribute

Defines the start time of the responses in UTC and MJD.

The time format conforms to the ISO 8601 standard, except that it is assumed to be UTC. A recommended usage is the form yyyymmddThhmmss, where the capital letter "T" separates the date from the time. It is possible to shorten the time string by reducing the resolution. For example, yyyymmddThhmm (no seconds specified) is valid.

This attribute specifies which HTTP method will be used to submit the form data to the server.

RESPONSESTOPTIME attribute

Defines the stop time of the responses in UTC and MJD. The format is identical to the RESPONSESTARTTIME attribute format.

This attribute specifies which HTTP method will be used to submit the form data to the server.

RETRYWAITTIME attribute

This attribute defines the time in minutes after which a decoder is allowed to retry. The value is a number in the range 0 and 65535.

RESPONSEATTEMPTLIMIT attribute

This attribute defines how many times a decoder may try to call a service provider. The value is a number in the range 0 and 255. A value of zero (0) means unrestricted calling.

6.2.2 <INPUT type = checkbox> tag

Function: This element is used to create a checkbox input-element.

Format:

<INPUT type=checkbox

CHECKED

NAME = “input-element name”

VALUE = “input-element value”>

Attributes supported: CHECKED, NAME, VALUE.

Example:

<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">

<INPUT TYPE=CHECKBOX NAME="maillist">Yes! Put me on the list!<P>

<INPUT TYPE=SUBMIT VALUE="submit">

</FORM>

produces a checkbox labelled “Yes! Put me on the list!” and a submit-button.

CHECKED attribute

This attribute – if present – specifies whether the checkbox should be checked by default.

NAME attribute

This attribute specifies a name for the input-element, which will be passed to the form processor if the input-element is selected. Checkboxes with the same name attribute form a group, allowing users to select multiple options with the same property.

VALUE attribute

This attribute specifies a value for the input-element, which will be passed to the form processor if the input-element is selected. If no value is set, the value “on” will be passed.

6.2.3 <INPUT type = hidden> tag

Function: This element is used to create a hidden input-element.

Format:

<INPUT type=hidden

NAME = “input-element name”

VALUE = “input-element value”>

Attributes supported: NAME, VALUE.

Example:

<FORM METHOD=POST ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">

<INPUT TYPE=HIDDEN NAME="postingID" value="98765">

name: <INPUT NAME="realname" SIZE=30>

email: <INPUT NAME="email">

subject: <INPUT NAME="subject" VALUE="Re: Hamlet and hesitation" SIZE=30> <P>

comments:

<TEXTAREA NAME="comments" COLS=50 ROWS=10 WRAP=VIRTUAL> Joe Smiley wrote: I think Hamlet doesn't act because if he does, the play's over. </TEXTAREA> <P>

<INPUT TYPE=SUBMIT VALUE="Send It!">

</FORM>

produces three single-line-text-input-elements (labelled name, email, subject), a multi-line-text-input-element (labelled comments) and a submit-button.

NAME attribute

This attribute specifies a name for the input-element.

VALUE attribute

This attribute specifies a value for the input-element, which will be passed to the form processor.

6.2.4 <INPUT type = image> tag

Function: This element is used to allow an image to be used as a submit-button.

Format:

<INPUT type=image
ALIGN = “TOP” or “MIDDLE” or “BOTTOM”
NAME = “image-name”
SRC = “URL”>

Attributes supported: ALIGN, NAME, SRC.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
    name: <INPUT NAME="realname">
    <INPUT
      TYPE=IMAGE
      SRC="../graphics/submit.gif"
      HEIGHT=110 WIDTH=160
      ALT="Send It In!" ALIGN=ABSMIDDLE>
</FORM>
```

produces a single-line-text-input-element labelled name and a picture, which can be selected.

ALIGN attribute

This attribute specifies the alignment of the image relative to the surrounding text lines.

NAME attribute

This attribute specifies a name for the image.

SRC attribute

This attribute specifies the URL for the image.

6.2.5 <INPUT type = password> tag

Function: This element is used to create a text-input element in which the text is rendered in a way that makes it unreadable, for example, replacing each character with a “*” character.

Format:

<INPUT type=password
MAXLENGTH = “maximum password length”
NAME = “input-element name”
SIZE = ” input-element size”
VALUE = “input-element default value”>

Attributes supported: MAXLENGTH, NAME, SIZE, VALUE.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl" METHOD=POST>
    name: <INPUT TYPE=TEXT NAME="realname"><BR>
    password: <INPUT TYPE=PASSWORD NAME="mypassword"> <P>
    <INPUT TYPE=SUBMIT VALUE="submit">
</FORM>
```

produces two single-line-text-input-element (labelled name and password) and a submit-button.

MAXLENGTH attribute

This attribute specifies the maximum number of character that the user may input.

NAME attribute

This attribute specifies a name for the input-element.

SIZE attribute

This attribute specifies the size of the input-element measured in number of characters.

VALUE attribute

This attribute specifies the default text for the input-element.

6.2.6 <INPUT type = radio> tag

Function: This element is used to create a radio button input-element.

Format:

```
<INPUT type=radio
    CHECKED
    NAME = “input-element name”
    VALUE = “input-element value”>
```

Attributes supported: CHECKED, NAME, VALUE.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
    What size T-shirt?<P>
    <INPUT TYPE=RADIO NAME="shirtsize" VALUE="S"       >small<BR>
    <INPUT TYPE=RADIO NAME="shirtsize" VALUE="M" CHECKED >medium<BR>
    <INPUT TYPE=RADIO NAME="shirtsize" VALUE="L"       >large<P>
    <INPUT TYPE=SUBMIT VALUE="submit">
</FORM>
```

produces three radio buttons (labelled small, medium, large), with the button labelled medium being the default selection.

CHECKED attribute

This attribute – if present – specifies whether the radio button should be “on” by default.

NAME attribute

This attribute specifies a name for the input-element, which will be passed to the form processor if the input-element is selected. Radio buttons with the same name attribute form a group, in which only one option at a time can be selected.

VALUE attribute

This attribute specifies a value for the input-element, which will be passed to the form processor.

6.2.7 <INPUT type = reset> tag

Function: This element is used to create a reset button, which clears the contents of the elements in a form.

Format:

<INPUT type=reset

VALUE = “button label”>

Attributes supported: VALUE.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
    <INPUT TYPE=TEXT>
    <INPUT TYPE=SUBMIT>
    <INPUT TYPE=RESET>
</FORM>
```

produces one single-line-text-input-element, a submit-button and a reset-button.

VALUE attribute

This attribute specifies a name for the input-element, which will appear as its label.

6.2.8 <INPUT type = submit> tag

Function: This element is used to create a submit button, which immediately sends the contents of the form to the server.

Format:

<INPUT type=submit

VALUE = “button label”

NAME = “input-element name”>

Attributes supported: VALUE, NAME.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
    <INPUT TYPE=TEXT>
    <INPUT TYPE=SUBMIT>
</FORM>
```

produces a single-line text input-element and a submit-button.

VALUE attribute

This attribute specifies a name for the input-element, which will appear as its label.

NAME attribute

This attribute specifies the name for the input-element, which will be passed to the form processor.

6.2.9 <INPUT type = text> tag

Function: This element is used to create a single-line text input-element.

Format:

<INPUT type=text
MAXLENGTH = “maximum password length”
NAME = “input-element name”
SIZE = ” input-element size”
VALUE = “input-element default value”>

Attributes supported: MAXLENGTH, NAME, SIZE, VALUE.

Example:

<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl">
name: <INPUT TYPE=TEXT NAME="realname"><P>
<INPUT TYPE=SUBMIT VALUE="submit">
</FORM>

produces a single-line-text-input-element labelled name and a submit-button.

MAXLENGTH attribute

This attribute specifies the maximum number of character that the user may input. Default value is unlimited.

NAME attribute

This attribute specifies a name for the input-element.

SIZE attribute

This attribute specifies the size of the input-element measured in number of characters.

VALUE attribute

This attribute specifies the default text for the input-element.

6.2.10 <TEXTAREA> tag

Function: This element is used to create a multi-line text input-element.

Format:

<TEXTAREA
COLS = value
NAME = input-element name
ROWS = value>

Attributes supported: COLS, NAME, ROWS.

Example:

```
<FORM ACTION="http://www.idocs.com/cgi-bin/mycgi.pl" METHOD=POST>
    your address:<BR>
    <TEXTAREA NAME="comments" COLS=40 ROWS=5></TEXTAREA>
    <P><INPUT TYPE=SUBMIT VALUE="submit">
</FORM>
```

produces a text-area labelled your address and a submit-button.

COLS attribute

This attribute specifies the width of the input-element measured in number of characters.

NAME attribute

This attribute specifies a name for the input-element.

ROWS attribute

This attribute specifies the height of the input-element measured in number of character lines.

6.2.11 <OPTION> tag

Function: This element is used to create an input-element within a select-tag.

Format:

```
<OPTION
    SELECTED
    VALUE = “input-element default value”>
    ...
</OPTION> (end tag is optional)
```

Attributes supported: SELECTED, VALUE.

Example:

```
<FORM>
    <SELECT NAME="cutlery">
        <OPTION VALUE="123-4">fork
        <OPTION VALUE="123-5">knife
        <OPTION VALUE="123-6">spoon
    </SELECT>
    <P>
    <INPUT TYPE=SUBMIT VALUE="submit">
</FORM>
```

Produces a pull-down menu with the selection items fork, knife and spoon, and a submit-button.

SELECTED attribute

This attribute – if present – specifies whether the input-element should be selected by default.

VALUE attribute

This attribute specifies the value for the input-element.

6.2.12 <SELECT> tag

Function: This element is used to create a selection input-element. It can be either a pull-down menu or a scrolling list. Each element in the selection input-element is called an option and is declared with the OPTION tag.

Format:

<SELECT
MULTIPLE
NAME = “input-element name”
SIZE = “number of rows”>
</SELECT>

Attributes supported: MULTIPLE, NAME, SIZE.

Example:

See <OPTION>.

MULTIPLE attribute

This attribute – if present – specifies whether more than one option may be selected.

NAME attribute

This attribute specifies the name for the selection input-element. The name will be passed to the form processor together with the value of each option.

SIZE attribute

This attribute specifies the number of rows displayed simultaneously in the selection input-element. The default value is 1 when a multiple is not used.

6.3 Internationalization

A decoder may support languages other than the Western European ones. To help the decoder determine what language was used for editing a particular HTML document, the document shall include the META tag in the HEAD portion of the document.

6.3.1 <META> tag

Function: The META element can be used to include name/value pairs describing properties of the document, such as content type, character set, etc.

Formats:

<META
HTTP-EQUIV = “Content-Type”
CONTENT = “text/html; CHARSET = character set name”>

<META
NAME = “Content-Type”
CONTENT = “text/html; CHARSET = character set name”>

Attributes supported: HTTP-EQUIV, NAME, CONTENT.

Example:

<META HTTP-EQUIV ="Content-Type" CONTENT="text/html; CHARSET=ISO-8859-7">

NAME attribute

Specifies the property name.

HTTP-EQUIV attribute

This attribute can be used in place of the name attribute and has a special significance when the document is retrieved via the HTTP protocol. HTTP servers may use the property name specified by the HTTP-EQUIV attribute to create an HTTP header.

In a broadcast environment, the attributes specified by the HTTP-EQUIV must be trans-coded to the corresponding file attributes. Table 1 lists the file attributes that can be used with HTTP-EQUIV. See IEC 62298-3, Clause 12, for their semantics.

CONTENT attribute

This attribute specifies the property value.

Table 1 – Attributes to use in HTTP-EQUIV and HTTP headers

Attribute	Function	Covered by http	No. per file
Type	Type of data in the file	Yes	0 or 1
Name	Name of the file	Yes	1
CRC	Cyclic redundancy code (CRC) checksum	Yes	0 or 1
Encryption/ Conditional access	Indicates that the contents of the file have been encrypted	Yes	0 or 1 Note 1
Compression	Indicates that the contents of the file have been compressed	Yes	0 or 1 Note 1
Parental rating	Age rating of the content according to ETSI EN 300 468	No	0 or 1
Language	Principal language used to author the text	Yes	0 or 1
Character set (encoding)	Character encoding used to author the content	Yes	0 or 1
Expire time	Time after which the file is no longer valid and can be deleted by the decoder	Yes	0 or 1
User group ID	Allows files to be distributed to particular decoders in order to support closed user group or conditional access services	No	0 or 1
Profile	Defines the TeleWeb profiles the page is intended for	No	0 or 1
Version	Defines the version of the file	No	1
NOTE 1 These attributes are mandatory for the files supporting or requiring these particular functions. NOTE 2 A decoder will process file-related attributes defined within this subclause. A decoder should ignore file-related attributes not listed in the table above.			

6.4 <FONT> tag

Function: This inline style tag is used to specify the size, colour, and font face of the enclosed text.

Format:

<FONT
FACE = "Face name "
COLOR = "#RRGGBB"
SIZE = "1", ...,"7", or, "-2", ...,"+4">
... text in the specified font ...
</FONT>

Additional attributes supported: FACE

Example:

<FONT FACE="download_font1, monospace" COLOR="#FFFFFF" SIZE="+1">

FACE attribute

This attribute specifies a list of font face names to be used to render the text. The list is checked from left to right using the first match. If none of the face names is available, the default proportional font is used to render the text.

The name of the resident proportional font is "proportional".

The name of the resident monospace font is "monospace".

6.5 <TIME> tag

Function: This element displays real-time clock information available in the decoder. The date and time up to a resolution of a second are supported in different formats.

Format:

<TIME
FORMAT = "string"
MONTHS = "string"
WEEKDAYS = "string"
AMPM ="string">

Attributes supported: FORMAT, MONTHS, WEEKDAYS, AMPM

Examples:

<TIME FORMAT="yyyy-MM-dd">	2001-07-04
<TIME FORMAT="yyyy.MM.dd 'at' HH:mm:ss">	2001.07.10 at 12:08:56
<TIME FORMAT="EEE, MMM d,''yy">	Wed, Jul 4, '01
<TIME FORMAT="hh:mm a">	12:08 PM
<TIME FORMAT="hh 'o''clock' a.">	12 o'clock PM.
<TIME FORMAT="K:mm a">	0:08 PM
<TIME FORMAT="yyyy.MMMMM.dd hh:mm aaa">	2001.July.04 12:08 PM
<TIME FORMAT="EEE, d MMM yyyy HH:mm:ss">	Wed, 4 Jul 2001 12:08:56
<TIME FORMAT="yyMMddHHmmss" >	010704120856

FORMAT attribute

Date and time formats are specified by date and time pattern strings. Within date and time pattern strings, unquoted letters from 'A' to 'Z' and from 'a' to 'z' are interpreted as pattern letters representing the components of a date or time string. Text can be quoted using single quotes (') to avoid interpretation. The string " ' ' " represents a single quote. All other characters are not interpreted; they are simply copied into the output string during formatting.

The following pattern letters are defined (all other characters from 'A' to 'Z' and from 'a' to 'z' are reserved):

Table 2 – Format attributes for real-time clock

Symbol	Meaning	Presentation	Example
Y	Year	(Number)	1996
M	Month in year	(Text and number)	July and 07
D	Day in month	(Number)	10
I	Hour in am/pm (1~12)	(Number)	12
H	Hour in day (0~23)	(Number)	0
M	Minute in hour	(Number)	30
S	Second in minute	(Number)	55
E	Day in week	(Text)	Tuesday
D	Day in year	(Number)	189
F	Day of week in month	(Number)	2 (2nd Wed in July)
U	Week in year	(Number)	27
W	Week in month	(Number)	2
A	AM/PM marker	(Text)	PM
'	Escape for text	(Delimiter)	
''	Single quote	(Literal)	'

Pattern letters are usually repeated, as their number determines the exact presentation.

Text If the number of pattern letters is 4 or more, the full form is used; otherwise, a short or abbreviated form is used if available.

Number The number of pattern letters is the minimum number of digits, and shorter numbers are zero-padded to this amount.

Year If the number of pattern letters is 2, the year is truncated to 2 digits; otherwise, it is interpreted as a number.

Month If the number of pattern letters is 3 or more, the month is interpreted as text; otherwise, it is interpreted as a number.

MONTHS attribute

This attribute defines a comma-separated list of names to be used instead of the default names for the months.

MONTHS=”JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC”

WEEKDAYS attribute

This attribute defines a comma-separated list of names to be used for the days.

WEEKDAYS=”MON,TUE,WED,THU,FRI,SAT,SUN”

AMPM attribute

This attribute defines a comma-separated list of names used for a.m. and p.m. indication.

AMPM=”am,pm”

7 User profiles

A device according to this profile can provide more than one user profile.

User profiles can contain sensitive information like user IDs, passwords and credit card data. For this reason, user profiles need to be protected against unauthorized access or modification. It must be possible to edit and view user profiles.

Sensitive parts of a user profile may only be used after entering the correct password, for example, for acquiring access to an ISP.

It should be possible to remove all data in some or all user profiles simultaneously in a simple way. This feature is necessary for when a device changes owners.

The decoder manufacturer must define a user interface to alter the profile information and protect the information with appropriate methods.

8 ISP interface

The ISP interface defines the parameters used to build an IP connection to an ISP. The parameters determine which ISP to connect to, which address and access parameters to use and in which time-window. The ISP interface can be acquired from a broadcast or through the return-channel, or be resident within the platform. Because the ISP interface can be used to favour the use of certain ISPs, the control over it can have commercial value. An extra special function URL is defined to separate the interface from its presentation. Whatever format is chosen, the device manufacturer will be able to change both the ISP interface and its presentation.

A special function URL will be used to implement the machine-to-machine interface between the server and the decoder.

8.1 The ISP special function

An ISP configuration can be defined by a hyperlink using the URL syntax:

function:isp ? *isp-parameters*

where the sequence is not case-sensitive and

function:isp Identifies the special function ISP URL.

? *isp-parameters* A list of keyword-value pairs separated by semicolons. A keyword is separated from a value by an equals sign. The order and presence of keyword-value pairs is arbitrary. The keyword-value pairs that must be recognized are listed in Table 3.

Table 3 – ISP parameters

Item	Syntax	Description	Example
name	name=isp-name	text-string with a max. length of 32 characters	name=AOL
type	type=isp-type	text-string with a max. length of 16 characters	type=PSTN
telephone number	number=telephone-number	text-string with a max. length of 20 characters	number=00494012345 67
dns1	dns1=ip dot address	dot address with a max. length of 15 characters	dns1= 205.245.172.72
dns2	dns2=ip dot address	see description of dns1	see dns1 example
proxy ip address	proxyip=ip dot address	see description of dns1	see dns1 example
proxy port number	proxyport=4-digit hexadecimal number	text-string containing a 4-digit hexadecimal number	proxyport=8080
user id	userid=username	text-string with a max. length of 32 characters	Userid=john smith
user password	userpw=password	text-string with a max. length of 32 characters	Userpw=SecrET
charge	charge=CCNN.nnUU	text-string with a max. length of 32 characters	Charge=EU01.00PM
portal	portal=isp-portal-URL	URL string with a max. length of 128 characters. No query nor fragment part are allowed	portal=http://www.someportal.com

8.1.1 name

This parameter specifies the name of the internet service provider. It can contain any Latin-1 character.

8.1.2 type

This parameter specifies the type of connection to the internet service provider. Valid values are “PSTN”, “CATV”, “ISDN”, “DECT”, “GSM”, “LMDS” and “SMATV”.

8.1.3 telephone number

This parameter specifies the telephone number of the internet service provider. Valid characters are in the range [0,9].

8.1.4 dns1

This parameter specifies the IP dot address of the primary DNS server.

8.1.5 dns2

This parameter specifies the IP dot address of the secondary DNS server.

8.1.6 proxy ip address

This parameter specifies the IP dot address of the proxy server.

8.1.7 proxy port number

This parameter specifies the IP port number of the proxy server. Valid characters are in the ranges [0,9] and [A,F].

8.1.8 user id

This parameter specifies the login name used to access the internet service provider. It can contain any Latin-1 character.

8.1.9 user password

This parameter specifies the password associated with the user id. It can contain any Latin-1 character.

8.1.10 charge

This parameter specifies the charge associated with the connection session. It consists of a currency unit CC followed by an amount NN.nn followed by a unit UU. CC is a two-character string indicating the currency unit. NN.nn is a numerical string indicating an amount of money. UU can be either “TC” indicating total cost of the connection, or, “PM” indicating the connection cost per minute.

8.1.11 portal

This parameter specifies the URL of the internet service provider.

8.2 Network congestion control

The network congestion parameters NUMRESPONSESTHRESHOLD, RESPONSESTART TIME, RESPONSEENDTIME, RETRYWAITTIME and RESPONSEATTEMPTLIMIT are extra attributes to the FORM tag.

This subclause considers interactive broadcast services, where the broadcast signal is delivered through a distribution delivery system (for example, cable or satellite) and the return channel for interactivity is provided through a telecommunications network (for example, PSTN or ISDN).

Typical services that can be offered via interactive broadcast are services where a large number of consumers may react on broadcast programmes like tele-voting, quiz-games, reaction on commercials, ordering pizzas, etc. For this type of services the following three parameters are involved.

a) Reaction time: within what time shall all the reactions of the consumers be recorded and processed?
b) Accuracy: how many consumers' reactions have to be processed?
c) Investment: the number of lines (or modems) to service the reactions of consumers.

In general, it can be said that the service provider would like to get as many reactions in as short as possible a time with little investments, i.e. only a few lines and modems. This cannot be realized in practical situations. Two extreme cases are considered.

EXAMPLE 1

A service provider would like to have the opinions of as large a population as possible. The time allowed to retrieve this information may be rather long, for example, a day. A typical example would be to have a government election polling where the service provider would like to collect the opinions of as many people as possible and he announces the results the next day.

EXAMPLE 2

A service provider asks the watching audience about their favourite artist and he would like to show the results within 5 min (for example, after the commercial break). In this case, the service provider is probably satisfied if he gets an estimation of the result from a representative population, for example, 3 000 samples out of a much larger watching population.

These two examples are two extremes of a wide range of criteria between choosing the amount of people to react and the time within which these reactions have to be recorded. Figure 3 shows the two extremes.

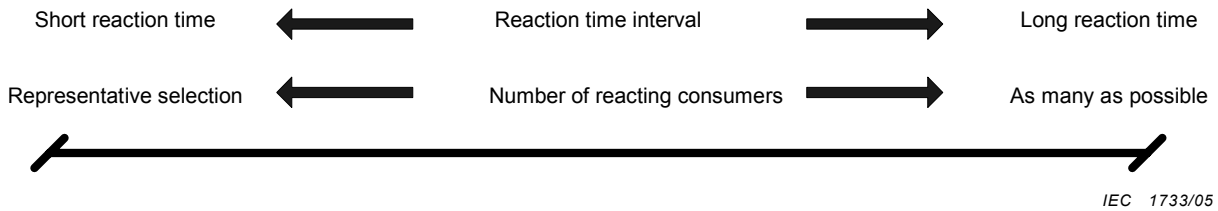


Figure 3 – Range of criteria for reactions

8.2.1 Traffic shaping

After a question from the service provider (whether it is the government election polling or the favourite artist selection), the consumers choose right away and press their selection via the remote control. In order to prevent that all decoders dial the service provider at the same time, a "time randomizer" is required to spread the reactions on the query over time. For instance, in the case of the government election polling application, several million people may select their favourite political party at exactly the same time. In order to avoid overloading the telephone network, the decoder dials the service provider to cast the vote, at a random moment within a time interval. This "time randomizer" function is sufficient for the first example (government election polling) where the service provider would like to have the opinion of as many people as possible within a rather long time interval. However, if the service provider would like to have the reaction within 5 min, the network operator cannot allow several million decoders to dial within a 5-min time interval. Therefore, a second function a "threshold randomizer" is required. This "threshold randomizer" determines whether or not the decoder will actually dial the service provider to make the response of the consumer known. If the service provider has a rough idea about how many people watch the programme and how many reactions will be sufficient to have a representative selection, he can determine the value of the "threshold randomizer".

8.2.2 IDL description of traffic shaping

To allow for traffic shaping an object implementing two mechanisms is defined as follows.

- Reduce the number of callers if necessary (the "threshold randomizer").
- Spread the responses in time (the "time randomizer").

The first point is achieved by signalling a threshold to the receiving decoder. If the user wants to participate in an interactive programme, the decoder generates a random 16-bit number. Only if this number is higher than the threshold, the box will place a call to the service provider.

The second point is implemented by giving start and stop times in between which a decoder may place its call. This should happen at a random moment in this interval. If the service provider is unreachable, the decoder may try again after a specified time. Again a random time in the resulting interval should be chosen. This minimum waiting time after a try is introduced to avoid traffic explosions when there is very little response time left.

9 Device ID (OPTIONAL)

A device according to this profile may contain a unique identifier according to EUI-64.

The identifier shall be

- uncorruptible;
- available for the device owner to view;
- available for transmission to an ISP.

NOTE A EUI-64 identifier consists of 64 bits, beginning with a 24-bit company-id followed by 40 bits. Such an identifier can be useful for applications like E-commerce and adds possibilities for new business models. Example: an advertiser can give a bonus to viewers with a TeleWeb device of a particular brand, if the viewer is tuned to a particular channel at a particular moment. The device owner could be allowed to register his device with the device ID, thereby diminishing the financial risk in case the device gets stolen. This feature requires an additional chip in the device, which adds to the cost of the device. Therefore, it is optional.

If enabled by the user, the Device ID is sent back to the server as part of the action URL of a FORM. The Device ID is a parameter in the query part of this action URL.

The attribute is defined as: DEVICEID=HHHHHHHHHHHHHHHH where H is a hexadecimal digit. This DEVICEID parameter is reserved and shall not be used within a FORM for any other purpose.

Example:
http://www.superteletext.tv/interactive/form_34e2.cgi?deviceid=324555dd89bba2f4&…

10 Cross-linking and context-switching

In order to have good consistency when the user navigates between services, it is necessary to save the context before switching to another service. Consequently, in a device according to this profile, each service – whether TeleWeb or Teletext – shall have its own context.

A context for a TeleWeb service must contain at least

- the URL that was displayed before the switch occurred and all the content it references;
- the bookmarked URLs and all the content they reference.

A context for a Teletext service must contain at least the page number or subpage number that was displayed before the switch occurred and their content.

10.1 Additional URI schemes

A decoder shall support the tv:, lid: and javascript: URI schemes as defined in SMPTE 363M, subclause 4.6. The lid: is used for content referencing and may be used in triggers as well as in HTML content. The javascript: is used for immediate ECMAScript execution, and the tv: is used for referencing related video – either as full-screen or sub-screen PiP-like functionality.

11 Audio clips (OPTIONAL)

Decoders may support “audio/basic” (8-bit PCM) content according to IETF RFC 2046, subclause 4.3.

Audio clips are files containing audio information. When used together with triggers they can be made synchronous with the A/V content. One utilization of audio clips is to provide audio description of programme content for people with impaired vision.

Devices supporting audio clips shall support the CELP (code-excited linear predictive), HVXC (harmonic vector excitation coding) low-bitrate audio coding formats and MP3.

Table 4 – Audio clip parameters

	Min bitrate (bits/s)	Max bitrate (bits/s)	Fileformat
CELP	6 000	12 000	See ISO/IEC 14496-3
HVXC	1 500	4 000	See ISO/IEC 14496-3
MP3	16 000	64 000	See ISO 11172-3

A broadcast containing synchronized audio clips shall broadcast a trigger, which indicates the availability of audio clips and enables the user to choose whether or not to listen to them. If the user chooses to listen to the audio clips, they shall be played until the end of the programme, or until the user no longer wishes to hear them. At the end of the broadcast a trigger shall be broadcast, which indicates that audio clips are no longer available.

The broadcaster shall make sure that audio clips do not overlap, i.e. one audio clip should have finished playing before the next audio clip commences. If due to unforeseen circumstances, overlap does occur, the device shall abort the first audio clip, and start playing the next one. If an audio clip has expired before being played, it shall not be played.

12 Style sheets

A device according to this profile shall implement cascading style sheets, level 1 (CSS1).

13 Scripting

Procedural capabilities extend the scope of TeleWeb. New classes of applications and features that become feasible include

- local score-keeping for local interactivity game-shows;
- checking the user input in forms;
- customized congestion control;
- local tools, like a currency converter;
- interfacing to programmable devices like a VCR;
- games with semi-static screen objects, like memory, slide;
- games with moving screen objects, like action games, needs at least DOM-1 or access to style attributes.

A notable feature of scripting is that it can reduce the return channel traffic.

13.1 ECMAScript

A device according to this profile shall implement ECMAScript, with the limitations described in this clause.

The following ECMAScript limitations are imposed.

- The prototype property for objects (which is used for implementing inheritance).
- Array sizes shall be limited to 32 768 elements.
- The minimum precision for the number datatype shall be 32 bits.

13.2 <SCRIPT> tag

Function: Defines the beginning of a script and the language used in the script.

Format:

<SCRIPT LANGUAGE = language>

…

</SCRIPT>

Attributes supported: LANGUAGE.

Example:

<script language="ECMAScript">

… // script statements

</script>

LANGUAGE attribute

This attribute specifies the scripting language within the tag. The only value supported is "ECMAScript". The decoder shall ignore any script element with a different language attribute value.

13.3 Events

All intrinsic HTML 4 events shall be supported as specified in W3C Recommendation, HyperText Markup Language, version 4.0. Support of mouse events is optional.

13.4 Document object model (DOM)

A device according to this profile shall implement both the SMPTE 366 DOM-0 and the W3C document object model (DOM) Level 1 core interfaces, with the limitations described in this subclause.

The following shall not be supported.

- Interface text.
- Extended interfaces (CDATASection, DocumentType, Notation, Entity, EntityReference, ProcessingInstruction) defined in subclause 1.3 of DOM.

In HTML, object positioning is performed by the browser according to some algorithm, which considers the order in which objects occur in the source document, their sizes and other mark-up information. For games that feature moving objects, this algorithm will not work. Hence, it is necessary to be able to position some objects based on user input and the computations of a script. The style attribute allows most HTML elements to be exempt from the layout algorithm of the browser and to be positioned independently of other elements on the page.

14 Additional content types

14.1 Graphic files

A decoder shall support the portable network graphic (PNG) format as constrained in SMPTE 363, subclause 4.3.1.

14.2 Text files

A decoder shall support the "text/plain" content type as defined in IETF RFC 2046, subclause 4.1.

Annex A
(informative)

Security considerations

This standard assumes that all insertion points of the content described in this document are trusted sources (or “hosts” in Internet thinking). Therefore, the use of scripts in triggers and ECMAscript in general is not presumed to be a security problem for malicious content or viruses any more than the notion of rogue video and audio being inserted into a broadcaster’s emission. As with video and audio, the broadcaster is responsible for the quality of the data content of its programming.

Standards Survey

The IEC would like to offer you the best quality standards possible. To make sure that we continue to meet your needs, your feedback is essential. Would you please take a minute to answer the questions overleaf and fax them to us at +41 22 919 03 00 or mail them to the address below. Thank you!

Customer Service Centre (CSC)

International Electrotechnical Commission
3, rue de Varembé
1211 Genève 20
Switzerland

or

Fax to: **IEC**/CSC at +41 22 919 03 00

Thank you for your contribution to the standards-making process.

Nicht frankieren
Ne pas affranchir

Non affrancare
No stamp required

A Prioritaire

RÉPONSE PAYÉE

SUISSE

Customer Service Centre (CSC)
International Electrotechnical Commission
3, rue de Varembé
1211 GENEVA 20
Switzerland

Q1 Please report on **ONE STANDARD** and **ONE STANDARD ONLY**. Enter the exact number of the standard: *(e.g. 60601-1-1)*

.....

Q2 Please tell us in what capacity(ies) you bought the standard *(tick all that apply).* I am the/a:

purchasing agent ❑
librarian ❑
researcher ❑
design engineer ❑
safety engineer ❑
testing engineer ❑
marketing specialist ❑
other.....

Q3 I work for/in/as a: *(tick all that apply)*

manufacturing ❑
consultant ❑
government ❑
test/certification facility ❑
public utility ❑
education ❑
military ❑
other.....

Q4 This standard will be used for: *(tick all that apply)*

general reference ❑
product research ❑
product design/development ❑
specifications ❑
tenders ❑
quality assessment ❑
certification ❑
technical documentation ❑
thesis ❑
manufacturing ❑
other.....

Q5 This standard meets my needs: *(tick one)*

not at all ❑
nearly ❑
fairly well ❑
exactly ❑

Q6 If you ticked NOT AT ALL in Question 5 the reason is: *(tick all that apply)*

standard is out of date ❑
standard is incomplete ❑
standard is too academic ❑
standard is too superficial ❑
title is misleading ❑
I made the wrong choice ❑
other .....

Q7 Please assess the standard in the following categories, using the numbers:
(1) unacceptable,
(2) below average,
(3) average,
(4) above average,
(5) exceptional,
(6) not applicable

timeliness.....
quality of writing.....
technical contents.....
logic of arrangement of contents .....
tables, charts, graphs, figures.....
other .....

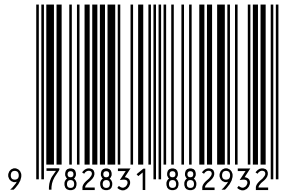
Q8 I read/use the: *(tick one)*

French text only ❑
English text only ❑
both English and French texts ❑

Q9 Please share any comment on any aspect of the IEC that you would like us to know:

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

ISBN 2-8318-8293-1



ICS 33.160.25; 35.240.99

Typeset and printed by the IEC Central Office
GENEVA, SWITZERLAND