

**Proposal for introducing
a trigger mechanism
into TV transmissions**

PUBLICLY AVAILABLE SPECIFICATION



INTERNATIONAL
ELECTROTECHNICAL
COMMISSION



Reference number
IEC/PAS 62297

IECNORM.COM : Click to view the full PDF of IEC PAS 62297:2002

Withdrawn

EACEM Technical Report

TR-037-r01

Title:

Proposal for introducing a trigger mechanism into TV transmissions

Proposed ETSI Title:

“Specification of a protocol defining remote triggering of data-broadcast applications”

Proposed ETSI keywords:

“trigger, broadcast, activation, signalling, interactive, data-broadcast”

Date: 23 April 2001

IECNORM.COM : Click to view the full PDF of IEC PAS 62297:2002

Withdrawn

INTERNATIONAL ELECTROTECHNICAL COMMISSION

PROPOSAL FOR INTRODUCING A TRIGGER MECHANISM INTO TV TRANSMISSIONS

FOREWORD

A PAS is a technical specification not fulfilling the requirements for a standard, but made available to the public and established in an organization operating under given procedures.

IEC-PAS 62297 was submitted by the EACEM (European Association of Consumer Electronics Manufacturers) and has been processed by IEC technical committee 100: Audio, video and multimedia systems and equipment.

The text of this PAS is based on the following document:

This PAS was approved for publication by the P-members of the committee concerned as indicated in the following document:

Draft PAS	Report on voting
100/406/PAS	100/437/RVD

Following publication of this PAS, the technical committee or subcommittee concerned will investigate the possibility of transforming the PAS into an International Standard.

An IEC-PAS licence of copyright and assignment of copyright has been signed by the IEC and EACEM and is recorded at the Central Office.

- 1) The IEC (International Electrotechnical Commission) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of the 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, the IEC publishes International Standards. 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. The 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 the 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 National Committees.
- 3) The documents produced have the form of recommendations for international use and are published in the form of standards, technical specifications, technical reports or guides and they are accepted by the National Committees in that sense.
- 4) In order to promote international unification, IEC National Committees undertake to apply IEC International Standards transparently to the maximum extent possible in their national and regional standards. Any divergence between the IEC Standard and the corresponding national or regional standard shall be clearly indicated in the latter.
- 5) The IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with one of its standards.
- 6) Attention is drawn to the possibility that some of the elements of this PAS may be the subject of patent rights. The IEC shall not be held responsible for identifying any or all such patent rights.

Version History

This document was drafted by EACEM Project Team 1.4 "Data Broadcasting" and replaces document **TP-14-99-16** and all of its versions as outlined below.

Version	Date	Author	Description
0.1	Feb 2000	Jo Vandale	Initial 'reworked' version after the TP1.4 meeting in Brussels
0.2	20 July 2000	Jo Vandale	Adaptations, EACEM TP1.4 and TeleWeb meeting in Rennes
0.3	8 Aug 2000	Jo Vandale	Adaptations during the EACEM TP1.4 meeting in Brugge
0.4	31 Aug 2000	Jo Vandale	Minor adaptations during the TeleWeb meeting in Rousset
0.5	11 Sept 2000	Jo Vandale	Editorial changes after internal review.
0.6	15 Sept 2000	Jo Vandale	Editorial changes after review by David Tarrant. Addition of Annex C 'Simple trigger protocol for page-format Teletext (Informative)' by David Tarrant.
0.7	29 Sept 2000	Jo Vandale	Removing the 'event-null' and replacing it with 'dummy URL' to realise the same goal, the ability to display the icon without running an application.
0.8	11 Oct 2000	Jo Vandale	Editorial changes after review by David Tarrant. Addition of the icon proposal to Annex B (Designed by Bob Vranken, Philips).
1.0	3 Nov 2000	Jo Vandale	Released version after review by the TeleWeb Technical Group.
2.0	5 Feb 2001	Jo Vandale	Adapting to the new EACEM references.
2.1 / r00	15 Feb 2001	Jo Vandale	Adding the new EACEM template (Cover, Header, Footer, ...)
3.0 / r01	23 April 2001	Jo Vandale	Adding the minor changes requested by Jorg Polkowski + meeting remarks EACEM Project Team 1.4

Contents

1	Scope	5
2	References	5
3	Definitions, Abbreviations and Conventions.....	5
3.1	Definitions and Tokens.....	5
3.2	Abbreviations.....	7
4	Trigger message.....	7
4.1	General.....	7
4.1.1	Viewer interaction	7
4.1.2	Priority ratings	8
4.1.3	Character coding.....	8
4.1.4	Future compatibility.....	8
4.2	Lifecycles.....	8
4.2.1	The trigger message and event message lifecycle.....	8
4.2.2	The event message preparation lifecycle.....	9
4.2.3	The application lifecycle.....	10
4.3	Trigger format.....	11
4.3.1	Trigger text length	12
4.3.2	Trigger text	12
4.3.3	Trigger repetition	16
Annex A	Requirements (Informative).....	17
Annex B	Code of Practice (Informative)	18
B.1	Time reference	18
B.2	Modifying triggers.....	18
B.3	Trigger expiry	18
B.4	Countdown values	18
B.5	Initial state	18
B.6	Multiple sources	18
B.7	Receivers not supporting concurrent ApplicationObject instances	18

B.8	Receivers supporting concurrent ApplicationObject instances	19
B.9	The ICON bitmap	19
B.10	The use of priority	19
B.11	Interaction of name and script attributes.....	20
B.12	Supporting multiple applications	20
Annex C	Simple trigger protocol for page-format Teletext (Normative)	21
C.1	Introduction	21
C.2	Trigger Message page.....	21
C.3	Coding of packets X/27	21
C.4	Allocation of packet X/27 links.....	21
C.5	Receiver actions.....	21

1 Scope

Existing data broadcasting specifications do not allow a service provider to **trigger** an **application** in a receiver. This document specifies a triggering scheme for TV broadcasting based on the requirements defined in Annex A. Examples of possible use include displaying information to warn of severe weather conditions and advising of extreme content in TV programmes. In an interactive system, a message or icon might be displayed inviting on-line access to vote, to register an interest in an advertised product, or to browse programme-related content.

For the purposes of this document a **trigger** is defined as information sent from a service provider as part of a data broadcasting transmission and intended to control an **application** in a receiver. Additional information can be supplied along with the basic **trigger** to allow filtering or prioritisation techniques to be applied at the receiver. The transmission aspects of trigger messages are specified in [1].

Annex C describes a simple method for triggering the display of a Teletext page from within a standard page-format Teletext transmission.

2 References

- [1] EACEM TR-038: "Trigger Transport Layers"
- [2] ISO, ISO-8859-1: "Information processing – 8-bit single byte-coded graphic character sets, Latin alphabets".
- [3] IETF, RFC 1738 (1994): "Uniform Resource Locators".
- [4] ETSI, TR 101 231: "Television Systems: Register of Country and Network Identification (CNI) and of Video Programming System (VPS) codes".
- [5] ETSI, ETS 300 706: "Enhanced Teletext Specification".
- [6] EACEM TR-047: "TeleWeb Application Part 2 Profile 1 Enhanced".
- [7] ATVEF, "Enhanced Content Specification", v1.1 r26.
- [8] ECMA, ECMA Standard 262, "ECMAScript Language Specification".
- [9] IETF, RFC 791 (1981): "Internet Protocol".
- [10] IETF, RFC 1071 (1988): "Computing the Internet checksum".

3 Definitions, Abbreviations and Conventions

3.1 Definitions and Tokens

For the purpose of this document the following terms and definitions apply, in singular or plural form:

ActiveTimeValue A member of the **ApplicationObject**. The value decrements at video frame rate. It is updated on every reception of an **event message**.

application	<p>Software running on a receiver that is addressed by the URL of a trigger message and provides the following modes of operation:</p> <ol style="list-style-type: none"> 1. The display of information, the playback of sound, the download of data, ... 2. The initiation of any action. <p>Application examples include the display of a simple text message sent as part of the trigger message, the display of a Teletext, TeleWeb or Internet page, information from an EPG, electronic voting, an emergency alert, ...</p>
ApplicationObject	An object storing the information about an application started or modified by triggers referencing the same URL.
attribute	A member of an ApplicationObject or TriggerObject storing the information transported via an attribute element .
attribute element	An attribute name/value pair.
attribute string	Any sequence of characters with codes in the range 0x20 to 0x7E inclusive, excluding square brackets (0x5B and 0x5D).
CountdownValue	A member of a TriggerObject . The value decrements at video frame rate. It is updated on every reception of a trigger mes .
DateTime	<p>A date and time instance of UTC expressed in the form: <i>yyyymmddThhmmss</i>, where <i>yyyy</i> represents a year, <i>mm</i> represents a month (range 1 - 12), <i>dd</i> represents the day of the month (range 1 - 31), the capital letter "T" separates the date component from the time component, <i>hh</i> represents an hour (range 0 - 23), <i>mm</i> represents the minutes (range 0 - 59) and <i>ss</i> represents the seconds (range 0 - 59).</p> <p>It is possible to shorten the description by reducing the resolution. For example <i>yyyymmddThhmm</i> (no seconds specified) is valid, as is simply <i>yyyymmdd</i> (no time specified at all). When no date component is specified, the date reference shall be assumed to be the current day. When no time component is specified, the time reference shall be assumed to be midnight at the beginning of the specified day.</p>
Dummy URL	A URL that does not reference any application or data. It is used in the mandatory URL field of a trigger message when the intention is to display only the trigger icon (together with its text) and not to control an application.
event message	Information extracted from a trigger message that is used to create an ApplicationObject .
event start	An event message with its 'script' attribute element set to 'start'.
event stop	An event message with its 'script' attribute element set to 'stop'.
pending trigger	The state where a trigger message has created a TriggerObject but the conditions to create an ApplicationObject have not yet occurred.
priority filtering	Rejecting a trigger message on account of the value assigned to its 'priority' attribute element .

RelativeTime	A time period measured in seconds and video frames. It can be expressed in one of the following forms: <i>Fff</i> , <i>s</i> , <i>sFff</i> , <i>ss</i> , <i>ssFff</i> , <i>sss</i> , <i>sssFff</i> , <i>ssss</i> or <i>ssssFff</i> , where <i>s</i> represents a time in seconds, <i>f</i> represents a number of video frames and the capital letter "F" separates the seconds component from the frames component. The seconds component may contain up to four decimal digits. Thus the maximum number of seconds that can be specified is 9999. The maximum number of frames that can be specified is 25 for 50Hz systems and 30 for 60Hz systems. A frame value, if specified, must always be defined using two digits (e.g. 5 frames are encoded as F05).
string	Any sequence of characters with codes in the range 0x20 to 0x7E inclusive. Throughout this document strings are not case sensitive unless otherwise indicated.
trigger	A signal sent from a service provider as part of a data broadcasting transmission with the intention to start or modify an application at a certain time.
trigger character	A character with a code in the range 0x20 to 0x7E inclusive.
trigger del	A trigger message with a 'delete' attribute element .
trigger event	The instant in time when a trigger fires and an event message is created.
trigger mes	A trigger message without a 'delete' attribute element .
trigger message	The information embedded in a trigger and intended to control an application in a receiver.
TriggerObject	An object storing the information from all the triggers referencing the same URL.
trigger_text	The descriptive part of a trigger message .
URL string	Any sequence of characters with codes in the range 0x20 to 0x7E inclusive, excluding angle brackets (0x3C and 0x3E).

3.2 Abbreviations

- CLUT Colour Look Up Table
- CNI Country and Network Identification.
- EPG Electronic Programme Guide
- URL Uniform Resource Locator.
- UTC Co-ordinated Universal Time.
- VPS Video Programming System.

4 Trigger message

The **trigger message** allows **triggers** to be coded and implemented for different **applications**.

4.1 General

4.1.1 Viewer interaction

The mechanism through which the viewer enables or disables trigger handling or sets priority threshold levels is at the receiver manufacturer's discretion.

The appearance of an icon and the viewer interaction when responding to it is also at the receiver manufacturer's discretion.

4.1.2 Priority ratings

Triggers labelled with the "emergency" priority rating must always be processed, even if the viewer has disabled trigger handling. The "emergency" priority shall only be used by service providers for genuine emergency situations.

4.1.3 Character coding

All characters used to code **triggers** are taken from the ISO-8859-1 character set [2] and are in the range 0x20 to 0x7E inclusive. A character outside of this range shall be encoded using the percent character (%) followed by the two-digit hexadecimal value of the character represented in ASCII in ISO-8859-1 [2]. The '%' character itself is represented by the string '%25'.

4.1.4 Future compatibility

To ensure future compatibility, a receiver should ignore data it does not understand, such as **attribute elements** not defined by this edition.

4.2 Lifecycles

4.2.1 The trigger message and event message lifecycle

Figure 1 describes the lifecycle of a **trigger message** and an **event message** referencing the same resource (URL).

An incoming **trigger message** is acquired through the transport layer. The **priority filtering** process provides the opportunity to reject a **trigger message** on account of its 'priority' **attribute element**. However, it is not allowed to reject a **trigger message** with its 'priority' **attribute element** set to '0' (zero) as this value is reserved for emergency **trigger messages**.

The **Event message preparation** processes the filtered **trigger messages** and provides robustness to the **trigger** protocol when carried over a unidirectional transport layer where the reception of the information is not always guaranteed. Each **trigger message** carries a countdown value indicating the time delay before the **trigger** should fire. To aid robustness, the **trigger message** can be transmitted at intervals before the **trigger event**, each time with an updated countdown value. When the **trigger** fires, an **event message** is generated to the **application** referenced by the URL.

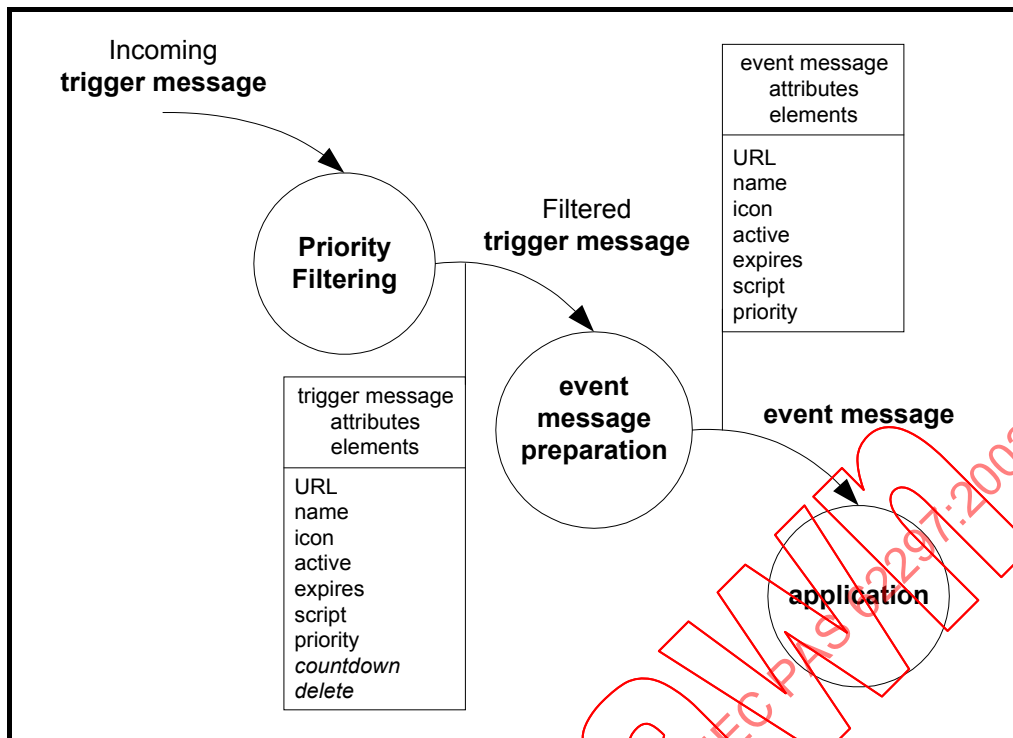
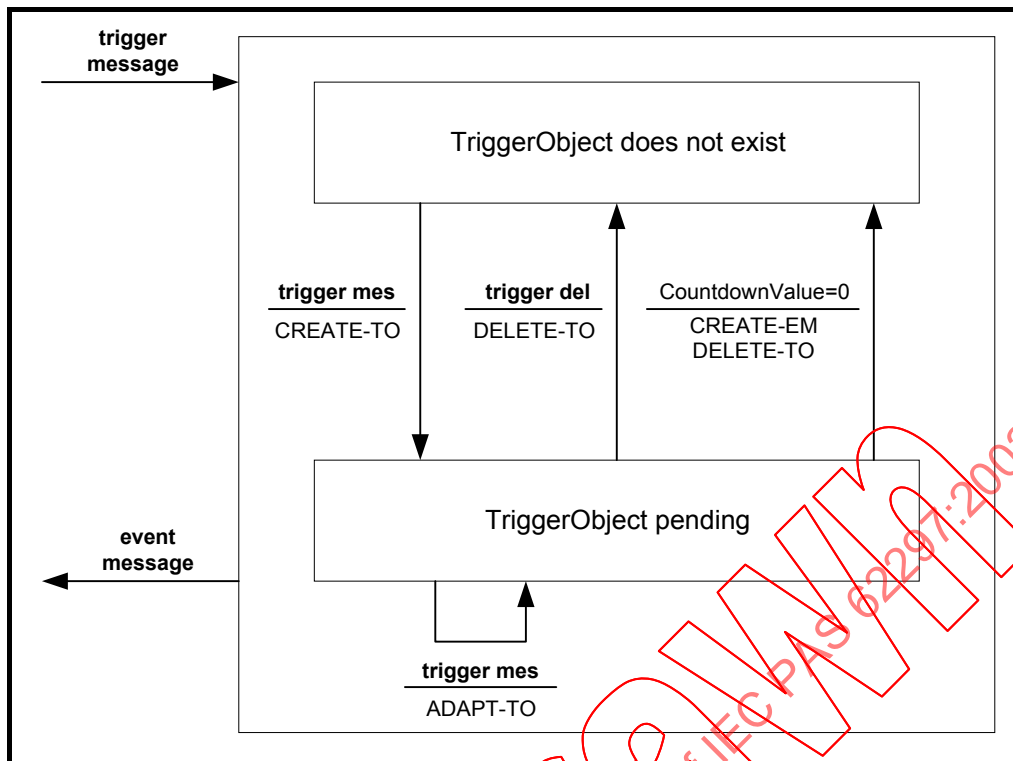


Figure 1: Trigger message and event message lifecycle

4.2.2 The event message preparation lifecycle

Figure 2 describes the state transitions within the **event message preparation** process. The initial state of a **TriggerObject** is "TriggerObject does not exist". On first reception of a **trigger message** without a 'delete' **attribute element**, a **TriggerObject** referencing the defined URL is created and the state becomes "TriggerObject pending". The **attributes** of the **TriggerObject** are adapted on subsequent arrivals of **trigger messages** referencing the same URL.

If the **CountdownValue** equals 0, either explicitly or as a result of decrementing at frame rate a value received previously, an **event message** is signalled to the **application** referenced by the URL. The **event message** inherits the **attribute elements** of the original **trigger message**, excluding the 'delete' and 'countdown' **attribute elements**. After signalling the **event message** to the application, the **TriggerObject** is deleted.



CREATE-TO

The creation of a **TriggerObject** referenced by the URL.

ADAPT-TO

The adaptation of a **TriggerObject** referenced by the URL.

DELETE-TO

The deletion of a **TriggerObject** referenced by the URL.

CREATE-EM

An **event message** is created and signalled to the **application**.

CountdownValue=0

The **CountdownValue** equals 0.

Figure 2: TriggerObject lifecycle

4.2.3 The application lifecycle

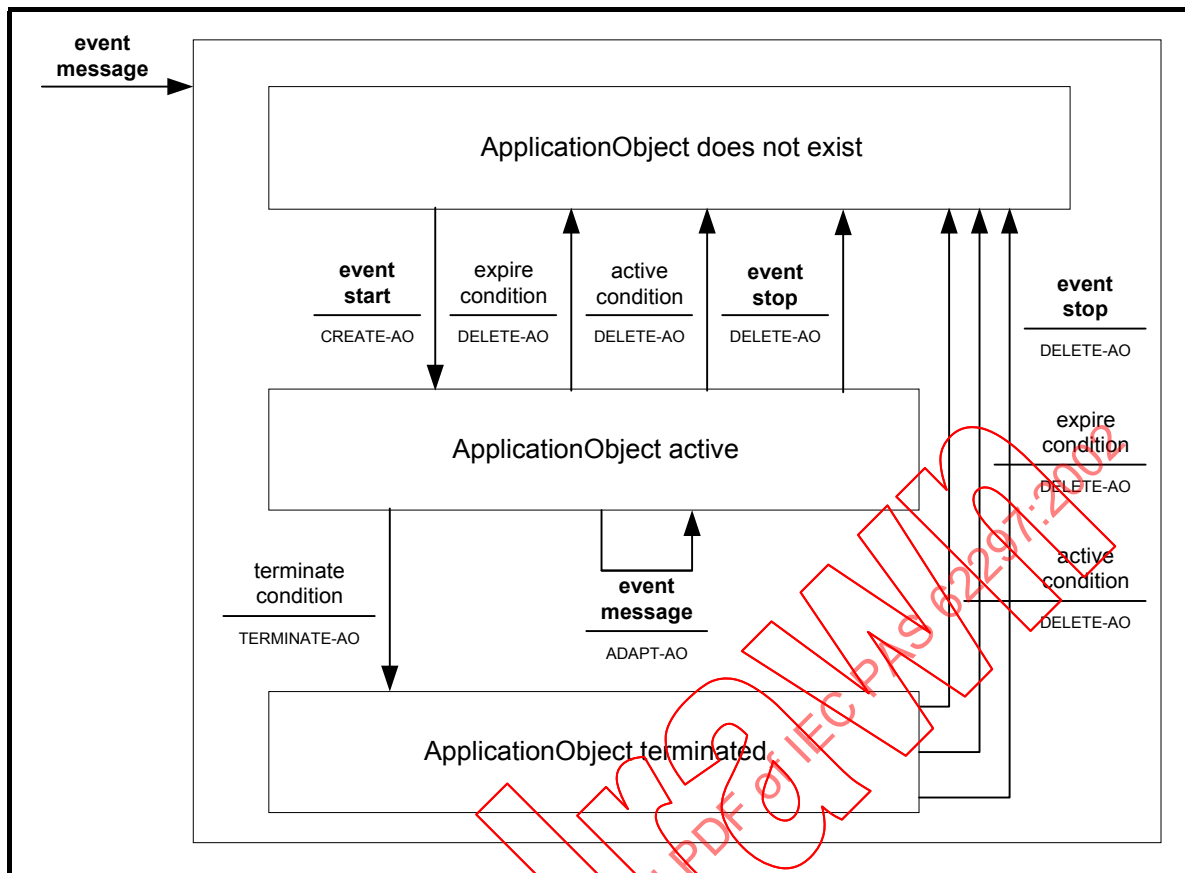
Figure 3 describes the state transitions within the **application** process. The initial state of an **ApplicationObject** is "ApplicationObject does not exist". An **ApplicationObject** is created as a result of an **event start**. An icon is displayed before the application is started if the 'name' attribute element is defined.

On first reception of an **event start**, an **ApplicationObject** referencing the defined URL is created and the state becomes "ApplicationObject active". The **attributes** of the **ApplicationObject** are adapted on subsequent arrivals of **event messages** referencing the same URL.

An **ApplicationObject** is deleted on reception of an **event stop**, on reaching the (absolute) 'expires' time or when the (relative) 'active' time period has been completed.

If the viewer terminates the **application**, the state becomes "ApplicationObject terminated". Once in this state, the **application** cannot be restarted until the **ApplicationObject** has been deleted. This adds robustness to the procedure and prevents the **application** restarting if the same **trigger** is repeated at a later time.

Note: This is needed if the broadcaster wants to address viewers who join the programme later.



expire condition

The DateTime value from the 'expires' attribute element is greater than or equal to the current Date Time value. The two values should be compared at video frame rate. The 'expires' attribute element may be updated on every instance of an **event message** referencing the same URL.

active condition

The **ActiveTimeValue** equals 0.

terminate condition

The **ApplicationObject** is terminated due to viewer action or other reason.

CREATE-AO

Create the **ApplicationObject** referenced by the URL.

ADAPT-AO

Adapt the **ApplicationObject** referenced by the URL.

DELETE-AO

Delete the **ApplicationObject** referenced by the URL.

TERMINATE-AO

Terminate the **ApplicationObject** referenced by the URL.

Figure 3: ApplicationObject lifecycle

On the reception of an **event start** with a 'name' **attribute element** defined, an **ApplicationObject** referencing the defined URL is created but the application itself is not run immediately. Instead, an icon defined by the 'name' **attribute element** is displayed. After the confirmation of the icon by the viewer, the **ApplicationObject** is fully started. The icon is removed if the **ApplicationObject** is deleted before a positive response from the viewer.

4.3 Trigger format

The format of a **trigger message** is shown in Table 1.

Table 1: Syntax of trigger_message

Syntax	No. of Bytes
trigger_message() {	
trigger_text_length	2
trigger_text()	trigger_text_length
}	

4.3.1 Trigger text length

The **trigger_text_length** field defines the number of characters in the following **trigger_text** field.

Note: For practical reasons, the maximum length of the **trigger message** may be limited by the application or transport protocol.

4.3.2 Trigger text

The format of the trigger_text field is shown in Table 2.

Table 2: Syntax of trigger_text

Syntax	No. of Bytes
Trigger_text() {	
for (i=0; i<trigger_text_length; i++) {	
trigger_character	1
}	
}	

The sequence of **trigger characters** starts with a **URL string** delimited by angle brackets. This is followed by one or more **attribute strings** delimited by square brackets. Optionally, the final element is a checksum delimited by square brackets:

<url> [attr₁:val₁][attr₂:val₂]...[attr_n:val_n][checksum]

4.3.2.1 URL element

The first element of the **trigger_text** must be a **URL string** enclosed in angle brackets. The URL element is used twofold: a) it uniquely identifies the **trigger**, and b) it locates the resource of the **application**. This implies that for each resource only one current **trigger** can be defined.

Internet URL

An Internet URL shall be identified by the sequence `http://` and shall be defined according to [3].

Example: `<http://xyz.com/fun.html>`

Local Identifier URL

A Local Identifier URL is identified by the sequence `lid://`. The Local Identifier URL scheme is defined in [7].

Example: `<lid://xyz.com/fun.html>`

TeleWeb URL

A string with the following syntax shall be used to specify a TeleWeb URL:

```
tw://service_name/filename.filetype#position
```

where the sequence is not case sensitive and :

- tw:// : Identifies the following sequence as a TeleWeb URL.
- service_name : identifies the service provider or the service being provided.
- filename : The name of the file. It can include directory style elements, e.g. /home_page/news/.
- . : Separator between the filename and filetype elements.
- filetype : The type of the file.
Note: The receiver may be limited in the number of file types they can handle.
- #position : Optional position element. # is the separator between the filetype and position elements, position is a location (anchorname) within the specified file.

Example: <tw://tvwest/name.type>

Teletext page URL

A string with the following syntax shall be used to reference a page in a teletext service:

```
ttx://cni/page_number/page_subcode
```

where the sequence is not case sensitive and:

- ttx:// : Identifies the following sequence as a teletext page URL.
- cni : The country and network identification (CNI) code for the network broadcasting the page. CNI codes are defined in [4]. This element shall always consist of four hexadecimal characters. If the network has only been assigned a three character VPS CNI code, it shall be preceded with 0. The default value of 0000 shall be interpreted as the "current channel", i.e. the page is being broadcast on the same channel via which the trigger message was received. The 16 bit NI code delivered through packet 8/30 format 1 is excluded here.
- /page_number : A three hexadecimal character value in the range 100 to 8FF representing the magazine, page tens and page units values respectively of a teletext page as defined in [5]. Setting both the page tens and page units values to 0xF shall indicate that no valid page number is being defined and no page should be displayed.
- /page_subcode : A four hexadecimal character value in the range 0000 to 3F7F representing the S4, S3, S2 and S1 values respectively of a teletext page subcode as defined in [5]. The value 3F7F shall indicate that no particular subcode value is being defined. S2 has a valid range of 0 to 7, and S4 a valid range of 0 to 3. The inclusion of a /page_subcode element is optional.

Example: <ttx://0DC2/456/3F7F>

When triggering the display of a teletext page, a receiver should display the most recent version of the page available. The handling of sequences of rolling pages is at the discretion of the receiver manufacturer.

Dummy URL

The Dummy URL does not reference any application or data. It is used when the purpose of the trigger is simply to display the trigger icon defined by the '**name**' **attribute element**, without running an application. As a result a Dummy URL is only valid if there is '**name**' **attribute element** defined within the same trigger message. The ApplicationObject is deleted after the confirmation of the icon by the viewer.

The Dummy URL is represented by the word 'dummy' (case insensitive) followed by two decimal digit characters within the range 00 to 99. This allows a maximum of 100 Dummy trigger instances at the same time.

Example: <dummy34>

4.3.2.2 Attribute elements

The body of the **trigger_text** consists of none, some or all of the **attribute elements** shown below. They are each described through an **attribute string** enclosed in square brackets. They can occur in any order and there should be no more than one instance of any one particular **attribute element** per trigger message. Attributes name can be shortened to a single character as indicated. All other single character attribute names are reserved for possible future use.

The following **attribute elements** are defined:

[active:RelativeTime]
[a:RelativeTime]

The **active** attribute specifies a time period in '**RelativeTime**' format. This represents an offset from the time at which the **trigger** is fired, after which the URL is no longer valid. As a consequence, the **ApplicationObject** defined by the URL is deleted. The value '0' shall be used if the receiver can determine the active duration.

If an '**expires**' **attribute element** is also defined, a receiver shall ignore the '**active**' **attribute element**.

Default: If this attribute is not present, the **ApplicationObject** is allowed to remain activated indefinitely, unless terminated by a viewer action or an **event stop**.

Example: [active:120] The **ApplicationObject** exists for 120 seconds.

[countdown:RelativeTime]
[c:RelativeTime]

The **countdown** attribute specifies a time period in '**RelativeTime**' format. This represents the delay from the current time until the **trigger** should fire thus allowing a **trigger event** to be scheduled in advance. The value '0' shall cause the **trigger** to fire immediately.

Default: If this attribute is not present, the **trigger event** should occur immediately.

Example: [countdown:F19] The **trigger** should fire in 19 frames time.

Note: To aid robustness, a **trigger** for a particular **trigger event** can be transmitted at intervals before the **trigger event**, each time with an updated countdown value. Receivers should not expect to receive every value in a countdown sequence.

[delete:void]
[d:void]

The **delete** attribute provides a means of deleting a **trigger** that has yet to fire. It takes precedence over all other attributes.

Default: If this attribute is not present, the **trigger** is not deleted.

Example: [delete:] Delete the **trigger**.

[expires:DateTime]
[e: DateTime]

The **expires** attribute specifies an absolute date and optional time in '**DateTime**' format, after which the URL is no longer valid. As a consequence, the **ApplicationObject** defined by the URL is deleted.

This **attribute element** takes precedence over an 'active' **attribute element**.

Default: If this attribute is not present, no automatic expiry time is defined.

Example: [expires:20000621T1700] The **Application Object** expires on 21st June 2000 at 17:00 UTC.

[name:string]
[n:string]

The **name** attribute indicates that an icon, defined by the receiver, must be displayed at the point the **ApplicationObject** is created. The **ApplicationObject** will be fully started if the viewer makes a positive response to the appearance of the icon. It is left to the equipment manufacturer to determine the look-and-feel of the user interface under these conditions. A code of practice may provide an example of a common logo for this purpose and may place a limit on the number of concurrent applications that can use this attribute.

The *string* provides information about the **trigger** in **attribute string** format. The text should be displayed together with the icon. The display format of the *string* is under control of the receiver but provision shall be made for the display of at least 20 characters. The receiver may ignore additional characters.

Default: If this attribute is not present, the **ApplicationObject** referenced by the URL shall be started immediately, priority level permitting, when the **trigger** fires.

Examples: [name:Weather]
 [name:Subtitles - page 888]

[priority:value]
[p:value]

The **priority** attribute defines an absolute rating for the importance of the **trigger**. The lower the value, the higher the priority. The priority range has ten levels and is divided into four categories:

a) emergency	value = 0
b) high	value = 1 or 2
c) medium	value = 3, 4 or 5
d) low	value = 6, 7, 8 or 9

Default: If this attribute is not present, a value of 9, i.e. lowest priority, shall be assumed.

Example: [priority:3]

[script:string]
[s:string]

The **script** attribute provides a script fragment to execute within the context of the **ApplicationObject** addressed through the URL. The *string* is an ECMAScript fragment [8] in **attribute string** format. The script can only act on an existing **ApplicationObject**.

Two specific strings are defined:

[script:start]	Creates the ApplicationObject addressed through the URL.
[script:stop]	Destroys the ApplicationObject addressed through the URL.

Default: If this attribute is not present, [script:start] shall be assumed.

Example: [script:frame1.src="http://xyz.com/f1"]
 The content of an HTML frame within a page is changed to a new URL.

4.3.2.3 Checksum element

The optional checksum element is provided to allow a receiver to detect corrupted data.

To compute the checksum, adjacent characters in the **trigger_text**, starting with the left angle bracket of the URL element and finishing with the right square bracket of the final attribute element (or the right angle bracket of the URL element if there are no attribute elements) are paired to form 16-bit integers. If there are an odd number of characters, the final character is paired with a byte of zeros. The checksum is computed so that the one's complement of all of these 16-bit integers plus the checksum equals the 16-bit integer with all 1 bits (0 in one's complement arithmetic). This checksum is identical to that used in the Internet Protocol [9] and further details on its computation are given in [10].

The calculated 16-bit checksum is appended to the trigger message as four hexadecimal digits, in ASCII representation and most significant byte first, and shall be enclosed within square brackets. The sequence shall follow the right square bracket of the final attribute element (or the right angle bracket of the URL element if there are no attribute elements).

Example: [A17C]

Because the checksum characters themselves (including the surrounding square brackets) are not included in the calculation of the checksum, they must be stripped from the **trigger_text** by the reception equipment before the received checksum is calculated.

4.3.3 Trigger repetition

A new instance of a **trigger message** with the same `<url>` element shall cause a receiver to update in full the resulting **TriggerObject** or **ApplicationObject** with the new values of the **attribute elements**. If a particular **attribute element** is not repeated, the object should revert to the appropriate default value for that **attribute element**.

Annex A Requirements (Informative)

A trigger mechanism for TV use needs to satisfy the following requirements:

- The method of conveying trigger information should be flexible and extendible.
- The trigger mechanism should provide the following modes of operation:
 1. The trigger initiates the display of information.
 2. The trigger initiates the display of an icon or message, the design of which is under the control of the receiver manufacturer. A positive response by the viewer to its appearance results in the display of information.
 3. The trigger initiates an action that does not require information to be presented on the screen.
- The general principles and coding should be applicable to a wide range of systems and transport protocols.
- URLs for all major data services should be supported.
- The protocols should be relatively easy to implement in all systems and on any appropriate platform.
- The protocol should have low transmission overheads to allow real-time operation.
- It should be possible to send the trigger information in advance and on several occasions to reinforce reception robustness and allow the receiver to allocate resources or pre-fetch data.
- The service provider should be able to clear triggers that have yet to fire.
- The trigger message should be able to specify the amount of time for which its application should remain active unless cancelled.
- The service provider or the viewer can cancel a trigger event.
- Time-related parameters should be compatible with the timing information that can be derived from the accompanying broadcast.
- The accuracy of the trigger event is application dependent but should be at least within 5 frames.
- Additional information can be supplied to customise the trigger according to the receiver or viewer profile.
- Triggers can have different priorities.
- Triggers can be targeted to particular applications or specific equipment types.

Annex B Code of Practice (Informative)

B.1 Time reference

To ensure a receiver has a suitable time reference, the service should provide a source of UTC time. Where the service is teletext based, a packet 8/30 format 1 should be broadcast.

B.2 Modifying triggers

If the definition of a trigger changes before it has completed, the pending **trigger** should first be cancelled and a new **trigger** issued. It is not recommended to modify a pending **trigger** other than to update the countdown attribute.

B.3 Trigger expiry

The expiry of a **trigger** should normally be determined by the 'expires' **attribute element**. The 'active' **attribute element** is intended only as a backup solution for realising a timeout counter in the receiver if time management is not possible or no reference time is available.

If the receiver is unable to manage the reference time or **pending triggers** while in standby or power-off modes, any **pending trigger** should be cancelled when entering these modes.

B.4 Countdown values

Receivers should not expect to encounter every value between the initial countdown value and 0. A countdown timer should be started in the receiver on first reception of the **trigger message**. This should be updated with the new countdown value on each subsequent reception and should decrement automatically at the appropriate time intervals. When the timer countdown reaches 0, the **trigger** should occur irrespective of whether a **trigger message** is received with a countdown value of 0, unless the **trigger** is deleted.

B.5 Initial state

In the initial state neither **TriggerObjects** nor **ApplicationObjects** exist. The initial state is entered each time the receiving equipment is switched on or after a channel change.

B.6 Multiple sources

When multiple sources of triggers are used simultaneously (e.g. dual screen display or picture-in-picture video features), at least one **trigger** stream should be processed. It is left to the receiver to select the source(s).

B.7 Receivers not supporting concurrent ApplicationObject instances

When a receiver does not support concurrent **ApplicationObject** instances, an active **ApplicationObject** should be terminated on the creation of an **ApplicationObject** with equal or higher priority. When the lower priority **ApplicationObject** resulted in the display of an icon that has yet to be confirmed by the viewer, it should be deleted.

B.8 Receivers supporting concurrent **ApplicationObject** instances

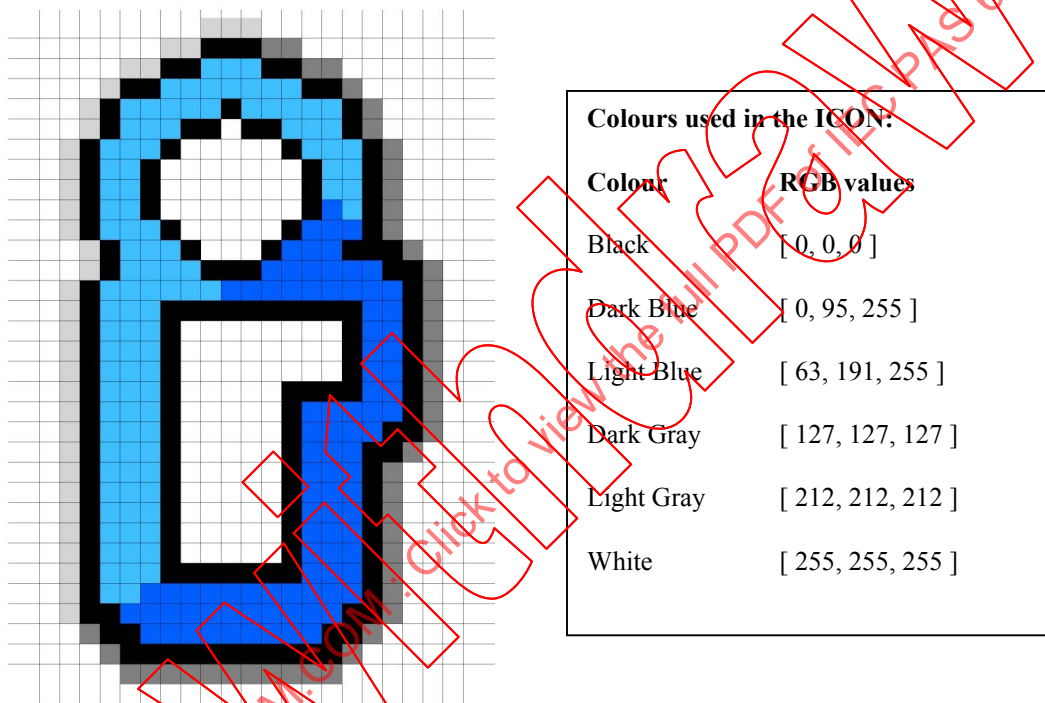
When a receiver supports concurrent **ApplicationObject** instances, an active **ApplicationObject** should be interrupted by the creation of a new **ApplicationObject** with equal or higher priority. 'Interruption' implies suspending the current **application** in order to run the new **application**. When the latter expires, the suspended **application** resumes.

The display of concurrent **ApplicationObject** instances is under control of the receiver. However, the latest created **ApplicationObject** with the highest priority should always be visible.

B.9 The ICON bitmap

If the 'name' attribute element is included in a trigger message, an icon is displayed before the application is started and the viewer is expected to react to its appearance. A proposal for an icon is presented here in an attempt to make its display platform and receiver independent so that it is always recognisable to the viewer. It incorporates the well-known lower-case "i" symbol, meaning information (Unicode code value 0x 2139).

The proposal is tailored for a display with a resolution of 640 by 480. The size of the icon is 22 by 32.



It is recommended to adopt both the shape and colours of this proposal. If the colours are incompatible with the receiver's user interface, it is still recommended to use the proposed shape. If a manufacturer wishes to design his own icon, the inclusion of a prominent "i" symbol is recommended.

B.10 The use of priority

Because **applications** can terminate abruptly, service providers should allocate priorities such that the resulting receiver behaviour described in B.7 and B.8 is not confusing to the viewer. Priorities should never be used to implement categories or themes, the latter should be handled within the **application** itself.

B.11 Interaction of name and script attributes

Service providers using the 'name' **attribute element** should note that if the content of the 'script' **attribute element** changes, the application might begin part way through a sequence of received scripts. This situation could arise if the viewer responded rather late to the display of the icon.

B.12 Supporting multiple applications

A service provider may wish to trigger the display of the same message via different applications using multiple triggers with the same priority, e.g. for Teletext, TeleWeb and TAK applications. Under these circumstances, it is left to the receiver supporting more than one of the applications being addressed to decide which single trigger to process or the order in which to process multiple triggers.

Withdrawing
IECNORM.COM : Click to view the full PDF of IEC PAS 62297:2002

Annex C Simple trigger protocol for page-format Teletext (Normative)

C.1 Introduction

This annex describes a simple method for triggering the display of a Teletext page from within a standard page-format Teletext transmission. A page designated as carrying trigger information is inserted in the transmission when there is a need to trigger the display of a page. The actual page to be displayed and possibly the reason for displaying it are indicated via a packet 27 link of this page.

C.2 Trigger Message page

The Trigger Message page for broadcasting the trigger messages of this document as part of a page-format Teletext transmission is fully defined in [1]. The simple trigger protocol defined in this annex adds up to four packets X/27 (with designation codes 0 up to 3) to such a page. If there are no trigger messages to be broadcast, the body of the page, packets X/1 up to packet X/25, will not exist. The inclusion of a packet X/28/0 to identify the page as a trigger message page is optional

C.3 Coding of packets X/27

The packets X/27 shall be coded according to section 9.6.1 of [5]. As there can be up to four packets, up to 24 page addresses can be specified. However, it is recommended that only one valid page number is included so that it is obvious to the receiver which page should be acquired and displayed

If there is no current requirement to trigger the display of a page but it is desired to keep an empty Trigger Message page in the transmission, all links should be set to the 'not defined' page value, i.e. page address mFF.3F7F.

C.4 Allocation of packet X/27 links

The first four page links of the packet X/27/0 should be used to trigger the display of pages carrying the following information:

Link 0 - a page warning of an unspecified type of emergency.

Link 1 - a page advising of (severe) weather conditions.

Link 2 - a page advising of extreme content in the current TV programme, i.e. parental advice.

Link 3 - a page providing backup details (e.g. contact addresses, URLs, recipe information) for the current TV programme.

All other links are unallocated by this edition and can be used when the situation requiring the trigger is not covered by one of the above categories.

C.5 Receiver actions

The receiver should monitor the transmission for the inclusion of a Trigger Message page, either using the default page number or an alternative number indicated via a MIP.

On reception of a Trigger Message page with valid packets X/27, a receiver should inspect them for valid links in order, starting with the first link in packet X/27/0. The first valid page number encountered should be assumed to be the page the service provider intended to be displayed.

A receiver may chose to allow the viewer to enable/disable the display of triggered pages. This could be implemented on a category basis using the categories shown above.

The length of time the triggered page remains visible, unless cancelled early by the viewer, is at the discretion of the receiver manufacturer.

Withdawn
IECNORM.COM : Click to view the full PDF of IEC PAS 62297:2002

EACEM Technical Report

TR-038-r01

Title:
Trigger Transport Layers

Proposed ETSI Title:
“Transmission protocols for the remote triggering of data broadcast applications”

Proposed ETSI keywords:
“data broadcasting, interactive TV, remote triggering, Teletext, TV”

Date: 23 April 2001

Version History

This document was drafted by EACEM Project Team 1.4 “Data Broadcasting” and replaces document **TP-14-00-004** and all of its versions as outlined below.

Version	Date	Author	Description
1.0	6 Nov 2000	Vincent Tauzia	Released version after review by the TeleWeb Technical Group.
2.0	5 Feb 2001	Jo Vandale	Adapting to the new EACEM references.
2.1 / r00	15 Feb 2001	Jo Vandale	Adding the new EACEM template (Cover, Header, Footer, ...)
3.0 / r01	23 April 2001	Jo Vandale	EACEM Project Team 1.4 approved version.

Withdrawing
IECNORM.COM : Click to view the full PDF of IEC PAS 62297:2002

Contents

1	Introduction	26
2	Scope	26
3	References	26
4	Definitions and abbreviations.....	26
4.1	Definitions.....	26
4.2	Abbreviations	27
5	Teletext based methods	27
5.1	Page format method.....	27
5.2	Independent data line method.....	28
Annex A	Code of Practice (informative)	31
A.1	Page format teletext transmissions	31
A.2	Independent data line teletext transmissions.....	31

1 Introduction

This report defines how the trigger messages defined in [11] are to be broadcast via Teletext. It is proposed that sections 2 to 5 of this report form the basis of a new European standard. It has been drafted in the ETSI format and references existing and proposed ETSI documents.

2 Scope

This document specifies how the trigger messages defined in [11] are transmitted. A trigger is defined as information sent from a service provider as part of a data broadcasting transmission that initiates an application in a receiver. Additional information in the trigger message allows filtering or prioritisation techniques to be applied at the receiver. Examples of possible use include the forcing of the display of information to warn of severe weather conditions and advising of extreme content in the TV programme. In an interactive system, a message or icon might be displayed inviting on-line access to vote, to register an interest in an advertised product, or to browse programme-related content.

The current document specifies how trigger messages are broadcast using Teletext technology. Both Page Format-Clear and Independent Data Line methods are defined, using protocols defined in [12].

3 References

- [11] EACEM TR-037: "Proposal for introducing a trigger mechanism into TV transmissions".
- [12] ETS 300 708: "Data transmission within Teletext" (edition 2).
- [13] ETS 300 706: "Enhanced Teletext specification" (edition 2).
- [14] EACEM TR-054: "Register of Application Codes for Teletext based systems".
- [15] ATVEF specification, v1.1 r26.
- [16] RFC 768: "User Datagram Protocol".

4 Definitions and abbreviations

4.1 Definitions

Application Data Block: One instance of the application data to be transmitted via the Page Format-Clear method and following the addition of any error protection/correction data.

Bundle: A group of 490 application bytes for transmission via IDL format B and to which the FEC algorithm is applied.

Bundle Information Structure: Data within the stream carried by the Page Format-Clear method that indicates the applications present in the stream.

Data Stream: A continuous or bursty sequence of data comprising application and transmission related components.

Hamming 8/4: A method of protecting data against transmission path errors. A protection bit is added for every data bit. Single bit errors per byte can be corrected, two bit errors can be detected.

Independent Data Line: A Teletext packet with address 30 or 31 that does not form part of a Teletext page. It can be inserted at any point in the transmission cycle.

Page Format-Clear: A page-based method of broadcasting serial data streams via Teletext when the application does not require encryption techniques to be applied to the data.