

ETSI TS 102 652 V1.1.1 (2007-10)

Technical Specification

Digital Audio Broadcasting (DAB); Intellitext; Application specification

European Broadcasting Union



Union Européenne de Radio-Télévision

EBU-UER

DAB
Digital Audio Broadcasting



Reference

DTS/JTC-DAB-50

Keywords

DAB, text

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

http://portal.etsi.org/chaicor/ETSI_support.asp

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2007.

© European Broadcasting Union 2007.

All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members.
TIPHONTM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.
3GPPTM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights	4
Foreword.....	4
1 Scope	5
2 References	5
2.1 Normative references	5
3 Definitions and abbreviations.....	5
3.1 Definitions	5
3.2 Abbreviations	6
4 Introduction	6
5 Message format	7
6 Updating and deleting messages	8
6.1 Updating messages	8
6.2 Deleting messages	8
7 Receiver behaviour.....	8
7.1 Validation	8
7.2 Parsing	9
7.3 Automatic deletion of messages	9
7.4 Segregation of data.....	9
Annex A (informative): Example Intellitext strings	10
A.1 Example 1.....	10
A.2 Example 2.....	11
A.3 Example 3.....	11
A.4 Example 4.....	13
Annex B (informative): Intellitext 1.0 and backwards compatibility	14
B.1 Differences between Intellitext 1.1 and 1.0.....	14
B.2 Receiver behaviour.....	14
B.2.1 Validation	14
B.2.2 Parsing	14
B.2.3 Automatic deletion of messages	15
B.2.4 Segregation of data	15
B.3 Examples	15
History	16

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<http://webapp.etsi.org/IPR/home.asp>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECTrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE 1: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

The Eureka Project 147 was established in 1987, with funding from the European Commission, to develop a system for the broadcasting of audio and data to fixed, portable or mobile receivers. Their work resulted in the publication of European Standard, EN 300 401 [1], for DAB (see note 2) which now has worldwide acceptance. The members of the Eureka Project 147 are drawn from broadcasting organizations and telecommunication providers together with companies from the professional and consumer electronics industry.

NOTE 2: DAB is a registered trademark owned by one of the Eureka Project 147 partners.

1 Scope

The present document describes the Intellitext DL extension, including structure and formatting of data and receiver and broadcast requirements. It also covers backwards compatibility with earlier versions of Intellitext.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
 - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
 - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://docbox.etsi.org/Reference>.

For online referenced documents, information sufficient to identify and locate the source shall be provided. Preferably, the primary source of the referenced document should be cited, in order to ensure traceability. Furthermore, the reference should, as far as possible, remain valid for the expected life of the document. The reference shall include the method of access to the referenced document and the full network address, with the same punctuation and use of upper case and lower case letters.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

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

- [1] ETSI EN 300 401: "Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

element: syntactic part of an Intellitext message

message: complete set of associated dynamic label segments

whitespace: one or more of the space character (code 20 hex)

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

DAB	Digital Audio Broadcasting
DL	Dynamic Label
X-PAD	eXtended Programme Associated Data

4 Introduction

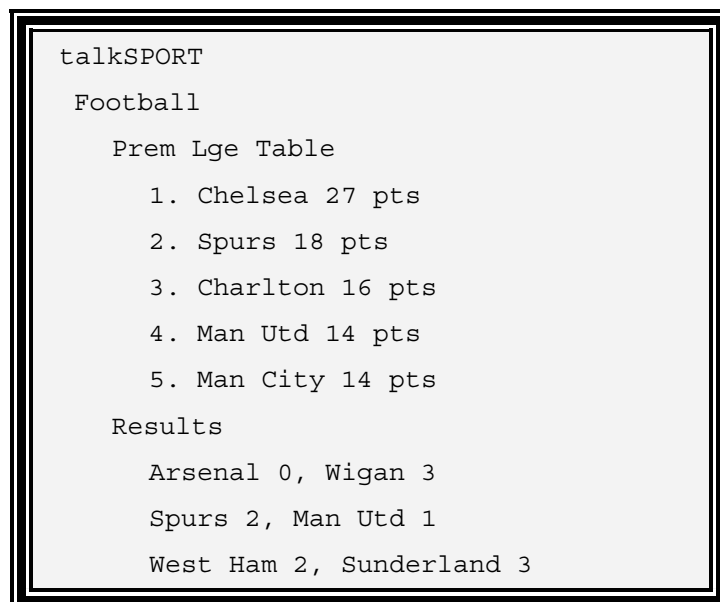
The present document describes an extension to the existing DAB Dynamic Label X-PAD application (see EN 300 401 [1] clause 7.4.5.2) to enable hierarchical-menu-driven text services on a compatible receiver. The data is compiled into a simple Teletext-like database of information which the user of any DAB radio equipped with this application can browse on demand.

Intellitext messages are a special form of DL messages, formatted in such a way that receivers not supporting Intellitext will continue to function normally. The Intellitext system provides a means for broadcasters to control the lifetime and basic formatting of broadcast information, while the display of information is user-driven.

The Intellitext system allows the broadcaster to dictate the structure and design of menus, including menu naming. The information provided by each service provider is stored in such a way that it cannot be altered by any other service provider.

Navigation is usually via a simple up/down/select interface, with the actual display being tailored to the resources available to a given receiver.

Intellitext messages consists of a category, a sub-category and some data. Within a given category, the sub-categories may be ordered by using a numerical index; similarly the data items are ordered within the sub-category. An example of the type of user display is shown in figure 1.



```
talkSPORT
Football
  Prem Lge Table
    1. Chelsea 27 pts
    2. Spurs 18 pts
    3. Charlton 16 pts
    4. Man Utd 14 pts
    5. Man City 14 pts
  Results
    Arsenal 0, Wigan 3
    Spurs 2, Man Utd 1
    West Ham 2, Sunderland 3
```

Figure 1: Example of display of Intellitext message data

5 Message format

Intellitext data is transmitted in the DAB DL, and will thus be displayed to users of non-compatible receivers (usually as "scrolling text"). It is therefore necessary that no "special" characters are used as mark-up characters, as these may be misinterpreted by receivers. Broadcasters should account for non-compatible receivers when deciding broadcast schemas.

The format of an Intellitext message is as follows:

<menu> <submenu_index> - <submenu> <data_index> : <data> <time_to_live>

The message comprises the following elements:

<menu>	the category this message refers to;
<submenu_index>	the (optional) sub-category index within the category;
<submenu>	the sub-category this message refers to;
<data_index>	the data index within the sub-category;
<data>	zero or more data items;
<time_to_live>	the (optional) lifetime of the data items.

The entire message must fit into a standard DL message, and thus has a total length of no more than 128 bytes. The message may contain as many data items as will fit within the space limitations.

All elements of the Intellitext message shall start and end with non-whitespace characters. Because Intellitext messages may be displayed by non-Intellitext capable receivers, the broadcaster may add whitespace between elements in order to present the message in human-readable form. Intellitext receivers will remove this whitespace (including any whitespace between elements and separating characters) during parsing of messages, such that each element starts and ends with a non-whitespace character.

The <menu> element is descriptive text which identifies the broad category that the message applies to. There may be several different <menu> elements broadcast, but only one per message. The <menu> element consists of one to 16 characters, and shall not include the hyphen ("-") or square bracket ("[" or "]") characters.

The <submenu_index> element is an identifier which allows the receiver to order the <submenu> elements within a category. It is an optional element. If the <submenu_index> element is omitted, the receiver defined default behaviour shall apply. The <submenu_index> element consists of the "[" character, a positive integer in the range 0 to 255 coded as 1 to 3 numeric characters, followed by the "]" character.

The <menu> and optional <submenu_index> elements are separated from the <submenu> element by a hyphen character ("-").

The <submenu> element is descriptive text which identifies the sub-category within the <menu> category that the message applies to. There may be several different <submenu> elements broadcast, but only one per message. The <submenu> element consists of one to 16 characters, and shall not include the colon (":") or square bracket ("[" or "]") characters.

The <data_index> element is an identifier which allows the receiver to order the <data> element or elements contained in this message within the sub-category. The <data_index> element consists of the "[" character, a positive integer in the range 0 to 255 coded as 1 to 3 numeric characters, followed by the "]" character.

The <data_index> element is separated from the <data> element by a colon character (":").

The <data> element is text which provides information within the sub-category. Each <data> element may contain zero or more data items, the limit being set by the size of the DL message. The <data> element consists of one or more data items. A data item consists of one or more characters, at least one of which must be non-whitespace, and shall not contain the semi-colon character (";"). Additional data items after the first are separated by semi-colon characters (";").

When multiple data items have been transmitted in a <data> element, the receiver will order them alphabetically with respect to each other. If a broadcaster-defined ordering is required, the data items should be broadcast in separate messages so that each has a unique <data_index>.

The <time_to_live> element controls the lifetime of the message within the receiver. It is an optional element. If the <time_to_live> element is omitted, the receiver defined default value shall apply. The <time_to_live> element consists of one, two or three period characters (".") and may only take the following values:

".": 24 hours;

"...": 12 hours;

"....": 1 hour.

The <time_to_live> element always appears at the end of the message. No trailing characters are permitted (this will lead to the characters comprising the <time_to_live> element being treated as part of the final <data> element). If the final data element ends with one or more period characters ("."), then this will be interpreted as a <time_to_live> element. If this behaviour is not desired, a space character (" ") should be appended to the <data> element. The <time_to_live> applies to all the data items within the message.

6 Updating and deleting messages

6.1 Updating messages

To update the <data> element of a message, a new message with the same <menu>, <submenu> and <data_index> is transmitted with the updated <data> element. The new <data> element may have a different number of data items as desired.

To update the <time_to_live> element of a message, a new message with the same <menu>, <submenu> and <data_index> is transmitted with the updated <time_to_live> element.

The <data> element and <time_to_live> element may both be updated with a single message.

The lifetime of a message may be extended by repeating the message. Receivers shall re-calculate the lifetime of the message each time it is received.

6.2 Deleting messages

To delete a message, a new message with the same <menu>, <submenu> and <data_index> is transmitted with a <data> element with zero data items (i.e. the <data> element is a colon character (":")).

NOTE: There are no guarantees that all receivers that were tuned to a service when the original message was broadcast will still be tuned to that service when a delete command is transmitted. It is therefore strongly recommended that the <time_to_live> element should always be used.

Messages will be automatically deleted by receivers when their lifetime expires.

7 Receiver behaviour

Intellitext messages are a special form of DL messages, but not all transmitted DL messages will be Intellitext messages. Therefore Intellitext capable receivers need to determine whether a received DL message is an Intellitext message or not in order to process the received message appropriately.

Intellitext messages are parsed and stored. The stored messages shall be updated and deleted to ensure that the data is appropriately maintained.

7.1 Validation

Receivers shall examine the received DL messages to determine whether the specified format for Intellitext messages is present. If the received DL message is an Intellitext message then it shall be parsed and the elements shall be stored in non-volatile memory in the receiver.

7.2 Parsing

The elements that make up the message shall be tokenised by dividing the message into its component parts. It is necessary to determine if the <time_to_live> element is present or not before trimming whitespace.

The whitespace which may have been added for presentation purposes between the elements of a message shall be trimmed by the parser. Whitespace may exist within each element and shall not be trimmed.

EXAMPLE:

The character strings below, will both produce the <menu> element "Prem·Lge" after parsing:

"Prem·Lge-" ".....Prem·Lge....."

where a space character is represented by ".".

7.3 Automatic deletion of messages

Each Intellitext message shall be saved to non-volatile storage along with its lifetime. Each message shall be deleted once its specified lifetime has passed. Intellitext messages received without a <time_to_live> element shall be given a default lifetime; the suggested values are one day, one week, or such other value as determined by the manufacturer.

Once all storage space has been filled, existing messages shall be deleted on an oldest-first basis. The maximum number of messages that may be stored by a given receiver will depend on a number of factors, and is receiver-specific.

7.4 Segregation of data

Intellitext messages transmitted by a given service shall not be aggregated with messages from any other service. This means that no service has influence over data transmitted by another service.

Annex A (informative): Example Intellitext strings

A.1 Example 1

The following DL messages are transmitted:

DL1 "Football - Prem Lge Table[1]: 1. Chelsea 27 pts; 2. Spurs 18 pts; 3. Charlton 16 pts"

DL2 "You're listening to Talksport with special studio guest Ledley King"

DL3 "Football - Prem Lge Table[3]: 7. Arsenal 12 pts; 8. Wigan 11 pts; 9. West Ham 10 pts"

DL4 "News – Headlines[1]: Aliens land in Kings Langley"

DL5 "Football - Prem Lge Table[2]: 4. Man Utd 14 pts; 5. Man City 14 pts; 6. Bolton 14 pts"

DL6 "News – Headlines[2]: Screaming Lord Sutch becomes new Conservative party leader"

DL7 "Football – Results[2]: Spurs 2, Man Utd 1"

DL8 "Football – Results[1]: Arsenal 0, Wigan 3"

DL9 "Football – Results[3]: West Ham 2, Sunderland 3"

Based on these nine DL messages, information will be presented to the user in the following format:

Football

Prem Lge Table

1. Chelsea 27 pts
2. Spurs 18 pts
3. Charlton 16 pts
4. Man Utd 14 pts
5. Man City 14 pts
6. Bolton 14 pts
7. Arsenal 12 pts
8. Wigan 11 pts
9. West Ham 10 pts

Results

Arsenal 0, Wigan 3

Spurs 2, Man Utd 1

West Ham 2, Sunderland 3

News

Headlines

Aliens land in Kings Langley

Screaming Lord Sutch becomes new Conservative party leader

A.2 Example 2

The following DL messages are transmitted:

DL1 "Football - Prem Latest[1]: Arsenal 1 - Wigan 1"

DL2 "Football - Prem Latest[2]: Bolton 0 – West Ham 0"

DL3 "Football - Prem Latest[3]: Spurs 1 - Charlton 2"

DL4 "Football - Prem Latest[1]: Arsenal 1 - Wigan 2"

DL5 "Football - Prem Latest[3]: Spurs 2 - Charlton 2"

DL6 "Football - Prem Latest[3]: Spurs 3 - Charlton 2"

The user will see the following sets of information:

1) after 3 messages received:

Football

Prem Latest

Arsenal 1 – Wigan 1

Bolton 0 – West Ham 0

Spurs 1 – Charlton 2

2) after 5 messages received:

Football

Prem Latest

Arsenal 1 – Wigan 2

Bolton 0 – West Ham 0

Spurs 2 – Charlton 2

3) after 6 messages received:

Football

Prem Latest

Arsenal 1 – Wigan 2

Bolton 0 – West Ham 0

Spurs 3 – Charlton 2

A.3 Example 3

The following DL messages are transmitted, and received at the given times:

12:00 DL1 "News – Latest[1]: Queen to give away lots of cash..."

12:05 DL2 "News – Latest[2]: Raving Loony Monster Party win election..."

12:25 DL3 "News – Economics[1]: Petrol companies make bumper profits"

12:45 DL4 "News – Latest[1]: Queen might give away lots of cash..."

13:10 DL5 "News – Latest[2]: "

The user will see the following sets of information:

1) after 2 messages received, at 12:15:

News

Latest

Queen to give away lots of cash

Raving Loony Monster Party win election

2) after 3 messages received, at 12:30:

News

Latest

Queen to give away lots of cash

Raving Loony Monster Party win election

Economics

Petrol companies make bumper profits

3) after 4 messages received, at 12:50:

News

Latest

Queen might give away lots of cash

Raving Loony Monster Party win election

Economics

Petrol companies make bumper profits

4) after 5 messages received, at 13:15:

News

Latest

Queen might give away lots of cash

Economics

Petrol companies make bumper profits

5) at 13:40:

News

Latest

Queen might give away lots of cash

Economics

Petrol companies make bumper profits

6) at 13:50:

News

Economics

Petrol companies make bumper profits

A.4 Example 4

The following messages are all invalid:

Cricket-England[4]: ; ; ; ;

Multiple data items, all empty

-Thai[2]:3 Lime leaves

No menu name

Prehistoric Creatures-Insects[2]:Beetles

Menu name is too long

Tennis-[3]:Lindsay Davenport

No sub-menu name

Football-Prem Lge:Spurs 3 - Charlton 2

No data index

Football-Scottish League Div 1[0]:Rangers 0 - Celtic 0

Sub-menu name is too long

Annex B (informative): Intellitext 1.0 and backwards compatibility

Intellitext 1.0 is an earlier version of the Intellitext application that may still be in use by some broadcasters. It is recommended that receivers that implement Intellitext 1.1 should also support Intellitext 1.0.

B.1 Differences between Intellitext 1.1 and 1.0

The format of an Intellitext 1.0 message is similar to an Intellitext 1.1 message, as defined in clause 5. However, the definition of the <menu>, <data_index> and <data> elements are different and there is no <time_to_live> element.

<menu> <submenu_index> - <submenu> <data_index> : <data>

The message comprises the following elements:

<menu>	the category this message refers to;
<submenu_index>	the (optional) sub-category index within the category;
<submenu>	the sub-category this message refers to;
<data_index>	the (optional) data index within the sub-category;
<data>	one or more data items.

The <menu> element is as defined for Intellitext 1.1 *except that* the <menu> element consists of two addition symbol characters ("++") followed by one to 16 characters.

The <submenu_index> element is as defined for Intellitext 1.1.

The <submenu> element is as defined for Intellitext 1.1.

The <data_index> element is as defined for Intellitext 1.1 *except that* it is an optional element.

The <data> element is as defined for Intellitext 1.1 *except that* there shall be at least one data item present.

B.2 Receiver behaviour

A receiver designed to process both Intellitext 1.1 and 1.0 messages requires additional validation and parsing rules to determine whether a received DL message is an Intellitext 1.1 message, an Intellitext 1.0 message or not in order to process the received message appropriately.

B.2.1 Validation

Receivers shall examine the received DL messages to determine whether the specified format for Intellitext 1.0 messages is present. All Intellitext 1.0 messages start with two addition symbol characters ("++"). If the received DL message is an Intellitext 1.0 message then it shall be parsed and the elements shall be stored in non-volatile memory in the receiver.

B.2.2 Parsing

The elements that make up the message shall be tokenised by dividing the message into its component parts.

The whitespace which may have been added for presentation purposes between the elements of a message shall be trimmed by the parser. Whitespace may exist within each element and shall not be trimmed.

If the <data_index> element was omitted by the broadcaster, unindexed data items should be stored alphabetically, and shall be displayed before any indexed items. Data items that start with numerical characters should be ordered before those beginning with alphabetical characters.

A message that contains no data items (or empty data items) is incorrectly formatted (there is no broadcaster delete facility in Intellitext 1.0); such messages are therefore invalid and shall not be processed as an Intellitext 1.0 message.

All received messages shall be assigned the default lifetime (there is no <time_to_live> element in Intellitext 1.0).

B.2.3 Automatic deletion of messages

Each Intellitext 1.0 message shall be saved to non-volatile storage with its lifetime. Each message shall be deleted once its specified lifetime has passed.

Once all storage space has been filled, existing messages shall be deleted on an oldest-first basis. The maximum number of messages that may be stored by a given receiver will depend on a number of factors, and is receiver-specific.

B.2.4 Segregation of data

Intellitext 1.0 messages transmitted by a given service shall not be aggregated with messages from any other service. This means that no service has influence over data transmitted by another service.

B.3 Examples

The following are valid Intellitext 1.0 messages:

"++Football - Prem Lge Table[1]: 1. Chelsea 27 pts; 2. Spurs 18 pts; 3. Charlton 16 pts"

"++Football - Prem Lge Table[3]: 7. Arsenal 12 pts; 8. Wigan 11 pts; 9. West Ham 10 pts"

"++News - Headlines: Aliens land in Kings Langley"

"++Football - Results: Spurs 2, Man Utd 1"

"++Football - Prem Lge Table[2]: 4. Man U 14 pts; 5. Man City 14 pts; 6. Bolton 14 pts"

This will produce the structure:

Football

Prem Lge Table

1. Chelsea 27 pts
2. Spurs 18 pts
3. Charlton 16 pts
4. Man U 14 pts
5. Man City 14 pts
6. Bolton 14 pts
7. Arsenal 12 pts
8. Wigan 11 pts
9. West Ham 10 pts

Results

Spurs 2, Man Utd 1

News

Headlines

Aliens land in Kings Langley

History

Document history		
V1.1.1	October 2007	Publication