

DIN EN 62298-3

ICS 33.170; 35.240.99

**TeleWeb-Anwendung –
Teil 3: Superteletext-Profil (IEC 62298-3:2005);
Deutsche Fassung EN 62298-3:2005**

Teleweb application –
Part 3: Superteletext profile (IEC 62298-3:2005);
German version EN 62298-3:2005

Application Teleweb –
Partie 3: Profil supertélétexte (CEI 62298-3:2005);
Version allemande EN 62298-3:2005

Gesamtumfang 103 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Beginn der Gültigkeit

Die von CENELEC am 2005-06-01 angenommene EN 62298-3 gilt als Norm ab 2005-11-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN IEC 62298-3:2003-10.

Für die vorliegende Norm ist das nationale Arbeitsgremium K 742 „Audio-, Video- und Multimediasysteme, -geräte und -komponenten“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE zuständig.

Die enthaltene IEC-Publikation wurde vom TC 100 „Audio, video and multimedia systems and equipment“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem auf der IEC-Website unter „<http://webstore.iec.ch>“ mit den Daten zu dieser Publikation angegebenen Datum (maintenance result date) unverändert bleiben soll. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Die Reihe EN 62298 mit dem allgemeinen Titel „TeleWeb-Anwendung“ besteht aus folgenden Teilen:

- Teil 1: Allgemeine Beschreibung
- Teil 2: Ausgabeverfahren
- Teil 3: Superteletext-Profil
- Teil 4: Interaktive Profile

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

TeleWeb-Anwendung
Teil 3: Superteletext-Profil
(IEC 62298-3:2005)

Teleweb application
Part 3: Superteletext profile
(IEC 62298-3:2005)

Application Teleweb
Partie 3: Profil supertélétexte
(CEI 62298-3:2005)

Diese Europäische Norm wurde von CENELEC am 2005-06-01 angenommen. Die CENELEC-Mitglieder sind gehalten, die CEN/CENELEC-Geschäftsordnung zu erfüllen, in der die Bedingungen festgelegt sind, unter denen dieser Europäischen Norm ohne jede Änderung der Status einer nationalen Norm zu geben ist.

Auf dem letzten Stand befindliche Listen dieser nationalen Normen mit ihren bibliographischen Angaben sind beim Zentralsekretariat oder bei jedem CENELEC-Mitglied auf Anfrage erhältlich.

Diese Europäische Norm besteht in drei offiziellen Fassungen (Deutsch, Englisch, Französisch). Eine Fassung in einer anderen Sprache, die von einem CENELEC-Mitglied in eigener Verantwortung durch Übersetzung in seine Landessprache gemacht und dem Zentralsekretariat mitgeteilt worden ist, hat den gleichen Status wie die offiziellen Fassungen.

CENELEC-Mitglieder sind die nationalen elektrotechnischen Komitees von Belgien, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Lettland, Litauen, Luxemburg, Malta, den Niederlanden, Norwegen, Österreich, Polen, Portugal, Schweden, der Schweiz, der Slowakei, Slowenien, Spanien, der Tschechischen Republik, Ungarn, dem Vereinigten Königreich und Zypern.

CENELEC

Europäisches Komitee für Elektrotechnische Normung
European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique

Zentralsekretariat: rue de Stassart 35, B-1050 Brüssel

Vorwort

Der Text des Schriftstücks 100/924/FDIS, zukünftige 1. Ausgabe von IEC 62298-3, ausgearbeitet von dem IEC TC 100 „Audio, video and multimedia systems and equipment“, wurde der IEC-CENELEC Parallelen Abstimmung unterworfen und von CENELEC am 2005-06-01 als EN 62298-3 angenommen.

Nachstehende Daten wurden festgelegt:

- spätestes Datum, zu dem die EN auf nationaler Ebene durch Veröffentlichung einer identischen nationalen Norm oder durch Anerkennung übernommen werden muss (dop): 2006-03-01
- spätestes Datum, zu dem nationale Normen, die der EN entgegenstehen, zurückgezogen werden müssen (dow): 2008-06-01

Der Anhang ZA wurde von CENELEC hinzugefügt.

Anerkennungsnotiz

Der Text der Internationalen Norm IEC 62298-3:2005 wurde von CENELEC ohne irgendeine Abänderung als Europäische Norm angenommen.

Inhalt

	Seite
Vorwort	2
Einleitung	6
1 Anwendungsbereich	7
2 Normative Verweisungen	7
3 Begriffe und Abkürzungen	7
3.1 Begriffe	7
3.2 Abkürzungen	8
4 Anzeige	8
4.1 Darstellung der Farbe	9
4.2 Textdarstellung	10
4.3 Platzierung des Textes	12
4.4 Darstellung eines Bildes	12
4.5 Anzeigen im Bildseitenverhältnis 4:3 und 16:9	13
5 Eindeutige Quelladressen (URL, en: uniform resource locator)	13
5.1 Datei-Benennung	14
5.2 Zugriff über die Seitenzahl	15
5.3 Bezugsschema für TeleWeb-Datei	15
5.4 Schema für Teletextseiten-Verweise	18
5.5 Schema für NexTView-Verweise	19
5.6 Spezialfunktionen-URLs	19
6 Text und Hypertext	19
6.1 TeleWeb-HTML-Dateiformat	20
6.2 Syntax von TeleWeb-HTML-Tags	21
6.3 Elemente der Dokumentenstruktur	23
6.4 Tags des Header-Abschnittes	26
6.5 Paragraph formatieren	28
6.6 Zeichenformatierung	31
6.7 Hypertext-Links	35
6.8 Listen	39
6.9 Tabellen	43
6.10 Bilder	48
6.11 Lauftext	51
6.12 Außergewöhnliche ignorierte Tags	53
7 TeleWeb-Vorgabenstil	53
8 Bilddateien	53
8.1 GIF	53
8.2 JPEG	53

	Seite
9 Kennung für den Inhalt	54
9.1 Vordefinierte Themen und Codierung der Kennung	54
9.2 Altersbeschränkung	54
10 Spezielle Daten	54
10.1 Dienst-Kennzeichnungsgrafik	54
10.2 Homepage	55
10.3 Vorgabeseite	55
10.4 Profilerweiterungsseite	55
10.5 ZLIB-Verzeichnistabellen-Dateien	55
11 Dienstbezogene Attribute	55
11.1 Name	56
11.2 Information	56
11.3 Language	56
11.4 Transmission Schedule	56
12 Dateibezogene Attribute	57
12.1 Allgemeine Dateiattribute	59
12.2 Dateiinhalte-Attribute	60
12.3 Zeit- und datumsbezogene Attribute	61
12.4 Speicherbezogene Attribute	62
12.5 Attribute spezieller Funktionen	62
13 Short- und Full-TeleWeb-Dienst	63
14 Individualadressierung – Gruppenadressierung	63
Anhang A (informativ) HTML-Kompatibilität	65
Anhang B (normativ) Vorgabe-Farbpaletten-Spezifikation	68
Anhang C (normativ) Tabelle der vordefinierten Themen	70
Anhang D (informativ) TeleWeb DTD, Dokument-Typ-Definition	77
Anhang E (informativ) Vorgabe-Formatvorlage CSS2 für TeleWeb	88
Anhang F (normativ) Schriftmaße	91
Anhang G (informativ) TeleWeb-SuperteleText-Bezugsdecoder	97
Literaturhinweise	100
Anhang ZA (normativ) Normative Verweisungen auf internationale Publikationen mit ihren entsprechenden europäischen Publikationen	101
Bilder	
Bild 1 – Zeichenmaße	11
Bild 2 – Ausrichten von transparentem Inhalt über einem Videobild	13
Bild 3 – Allgemeine Inhalte einer HTML-Datei für TeleWeb-Verwendung	21
Bild 4 – COORDS-Attribute (SHAPE = RECT)	38
Bild 5 – COORDS-Attribute (SHAPE = CIRCLE)	38
Bild 6 – COORDS-Attribute (SHAPE = POLY)	39
Bild G.1 – TeleWeb-Logo	98
Bild G.2 – Blockschalbild TeleWeb-Bezugsdecoder	99

Tabellen

Tabelle 1 – Vorgabe-Farbpalette	10
Tabelle 2 – Schriftgrößen	11
Tabelle 3 – Datei-Benennung	14
Tabelle 4 – EPG-Parameter	17
Tabelle 5 – Die Profilparameter	17
Tabelle 6 – Spezialfunktionen-URLs	19
Tabelle 7 – Liste der dienstbezogenen Attribute	56
Tabelle 8 – Liste der dateibezogenen Attribute	58
Tabelle 9 – Bewertungsschema	60
Tabelle 10 – Filter-Attribute für nexTView-Verweisungen	64
Tabelle A.1 – Nicht unterstützte HTML V3.2-Tags	65
Tabelle A.2 – Nicht unterstützte HTML V3.2-Attribute	66
Tabelle A.3 – Unterstützte browserspezifische Tags	66
Tabelle A.4 – Unterstützte TeleWeb-spezifische Tags	67
Tabelle B.1 – Vorgabe-Farbpaletten-Spezifikation	68
Tabelle C.1 – Vordefinierte Themen	70

Einleitung

Zweck des TeleWeb ist es, nach Art des World Wide Web codierte Inhalte zum Fernsehgerät in den Wohnraum zu übertragen, um dem Zuschauer ein gesteigertes Fernseherlebnis zu bieten. Ein TeleWeb-Dienst sendet Dateien mit Text und Grafiken hoher Auflösung an geeignete Decoder. Die gesendeten Daten können mit Ereignissen in den begleitenden Fernsehprogrammen eng verknüpft sein, oder sie sind mehr allgemeiner Natur und emulieren einen herkömmlichen aber hochauflösenden Superteletext. Es werden verschiedene Profile definiert.

Es besteht die Absicht, fernsehbasierte Decoder kostenbewusst ohne Rückgriff auf die üblicherweise mit Personalcomputern verbundene Technologie zu implementieren. Zum Teil wird dies durch Begrenzen der Anzahl der verschiedenen Arten von Multimediadaten erreicht, die innerhalb eines Dienstes verwendet werden können. Durch umsichtige Gestaltung der Benutzerschnittstelle ist der Decoderhersteller in der Lage, für den Zugang zu TeleWeb-Diensten leicht zu bedienende Geräte anzubieten, für dessen Bedienung der Verbraucher keine Computerkenntnisse benötigt. Darüber hinaus können die Decoderhersteller ihre Produkte kundenfreundlicher gestalten, um sich so von ihren Konkurrenten zu unterscheiden.

In diesem Schriftstück wird das TeleWeb-Superteletextprofil festgelegt. Es konzentriert sich auf die Darstellungsschicht, speziell die Implementierung von HTML. Darüber hinaus werden die grafischen Anforderungen wie Farben und Zeichensätze und die verwendeten Inhaltsformate definiert.

1 Anwendungsbereich

In diesem Teil von IEC 62298 wird das TeleWeb-Superteletext-Profil festgelegt, das auf geeigneten Decodern die Anzeige von Text und Grafiken im Stil des Web ermöglicht. Ein „TeleWeb“-Dienst umfasst Multimedia-Dateien, deren Format und Attribute in dieser Spezifikation definiert werden. Diese Spezifikation konzentriert sich auf die Darstellungsschicht und speziell auf die Implementierung von TeleWeb-HTML. Darüber hinaus werden die grafischen Anforderungen wie Farbe und Zeichensätze und die verwendeten Inhaltsformate definiert. Für allgemeine Informationen und Informationen über die Transportschicht wird auf IEC 62298-1 und IEC 62298-2 verwiesen.

2 Normative Verweisungen

Die folgenden zitierten Dokumente sind für die Anwendung dieses Dokuments erforderlich. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

IEC 62297-1, *Triggering messages for broadcast applications*.

ISO 639-2, *Codes for the representation of names and languages – Part 2: Alpha-3 code*.

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

ISO 8859-1:1998, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*.

ETSI TR 101231, *Television systems – Register of Country and Network Identification (CNI), Video Programming System (VPS) codes and Application codes for Teletext based systems*.

ETSI EN 300231, *Fernseh-Systeme – Festlegung des Systems zur programmbezogenen Steuerung von Videorecordern für den Heimgebrauch (PDC)*.

ETSI EN 300468, *Digitaler Fernsehrundfunk (DVB) – Festlegung der Serviceinformation (SI) für DVB-Systeme*.

ETSI EN 300706, *Erweiterte Fernsehtext-Spezifikation*.

ETSI EN 300707, *Elektronischer Programmführer (EPG) – Protokoll für einen TV-Führer bei Verwendung elektronischer Datenübertragung*.

3 Begriffe und Abkürzungen

3.1 Begriffe

Für die Anwendung dieser Norm gelten die folgenden Begriffe.

3.1.1

Taste

Teil der Benutzerschnittstelle, der es dem Zuschauer ermöglicht, eine Seite auszuwählen oder ein Ereignis usw. auszulösen. Die Taste braucht nicht notwendigerweise als physikalische Taste an einer Fernbedienungseinheit vorhanden zu sein.

3.1.2

CDATA

Zeichendaten in einem HTML-Schriftstück. Zeichen-Entitäten und HTML-Auszeichnungen werden nicht erkannt.

3.1.3**bedingter Zugang****CA**

eine Einrichtung, mit der der Zugang des Benutzers zu Dienstkomponenten beschränkt werden kann

3.1.4**PCDATA**

geparste Zeichendaten in einem HTML-Schriftstück. In den Daten werden sowohl Zeichen-Entitäten (numerische und benannte Entitäten) als auch HTML-Auszeichnungen erkannt.

3.1.5**ganze Zahl mit Vorzeichen**

ein positiver oder negativer ganzzahliger Wert in Dezimalnotation. Der ersten Ziffer muss ein obligatorisches Plus (+) oder Minus (–) vorangestellt werden. Es darf zwischen dem Symbol und der ersten Ziffer keine Leerstelle bestehen.

3.1.6**Textfolge**

eine Folge von anzeigbaren Zeichen des lateinischen Alphabetes Nr. 1

3.1.7**ganze Zahl ohne Vorzeichen**

ein ganzzahliger Wert in Dezimalnotation ohne vorangestelltem Plus (+)- oder Minus (–)-Symbol

3.2 Abkürzungen

ASCII	American Standard Code for Information Interchange	Amerikanischer Normcode für Informationsaustausch
CA	Conditional Access	Bedingter Zugang
CDATA	Character Data	Zeichendaten
CRC	Cyclic Redundancy Check	Zyklische Redundanzprüfung
DTD	Document Type Definition	Definition des Schriftstücktyps
DVB	Digital Video Broadcasting	Digitaler Fernseh Rundfunk
ETS	European Telecommunication Standard	Europäische Telekommunikationsnorm
GIF	Graphics Interchange Format	Austauschformat für Grafiken
HTML	Hyper Text Mark-up Language	Hypertext-Auszeichnungs-Sprache
JPEG	Joint Picture Experts Group	Vereinigte Expertengruppe für Bilder
LSB	Least Significant Bit	Niedrigstwertiges Bit
MJD	Modified Julian Date	Modifiziertes julianisches Datum
MSB	Most Significant Bit	Höchstwertiges Bit
PCDATA	Parsed Character Data between tags	Geparste Zeichendaten zwischen Tags
RFC	Request for Comments	Anforderung für Kommentare
URL	Uniform Resource Locator	Eindeutige Quelladresse
UTC	Universal Time Co-ordinated	Koordinierte Weltzeit
WWW	World Wide Web	Weltweites Netz (Internet)

4 Anzeige

Dieser Abschnitt definiert die Mindestanforderungen an eine TeleWeb-Anzeige. Die Anforderungen gelten sowohl für Editierstationen als auch für Decoder.

4.1 Darstellung der Farbe

4.1.1 Allgemeine Anforderungen

Ein Decoder muss in der Lage sein, jedes Pixel von Text und Grafik, Hintergrundbild- und Hintergrundfarben-Ebenen in verschiedenen Farben aus einer Palette von mindestens 188 Farben anzuzeigen. Vollständige Transparenz und einstufige Halbtransparenz für das Videobild müssen auch unterstützt werden.

Ein Decoder darf entweder mit einem True-Colour-Speicher oder einem indizierten Farbsystem implementiert werden. Um den zweiten Decodertyp anzupassen und eine Palette für GIF-Bilder zu definieren, die selbst keine Palette festlegen, wird in 4.1.5 eine einzelne Palette mit 188 Farben definiert. Text und Grafiken, Hintergrundbild- und Hintergrundfarben-Ebenen müssen mindestens Farben unterstützen können, die zu diesen Farben subjektiv äquivalent sind. Inhalte können unter Verwendung von anderen Farben erzeugt werden, den Verfassern sollte aber bewusst sein, dass manche Decoder nicht in der Lage sind, sie genau wiederzugeben, und sie dürfen dann eine solche Farbe in der am besten dazu passenden aus der vordefinierten Farbpalette abbilden.

Für das Verfassen müssen Texte/Grafiken, Hintergrundbild- und Hintergrundfarben-Ebenen jederzeit mindestens 188 Farben auf dem Bildschirm unterstützen. Praktisch dürfte ein Decoder in der Lage sein, mindestens 256 Farben anzuzeigen und dem Gerätehersteller mindestens 68 Farben für die Benutzerschnittstelle zu überlassen.

Um eine optimale Anzeige zu erhalten, sollte der Inhalt unter Verwendung der TeleWeb-Vorgabepalette erzeugt werden. Werden die verwendeten Farben nicht alle von der vordefinierten Farbpalette genommen, ist in dieser Spezifikation die Reaktion eines Decoders nicht definiert. Unter diesen Umständen kann es erforderlich sein, Farbdithering oder Anpassungstechniken anzuwenden. Dabei kann sich die Wiedergabe der Decoder unterscheiden.

4.1.2 Farbauflösung

Jede Farbe muss durch Rot-, Grün- und Blau-(RGB) Komponenten oder eine Farbkonstante definiert werden (siehe 6.2.5.1).

Der redaktionelle Inhalt muss Farben als 24-Bit-Werte definieren, d. h. 8 Bit für jede Komponente in der Reihenfolge R, G, B.

Der Decoder muss eine Farbauflösung von mindestens 4 Bit je Komponente (mindestens 12 Bit insgesamt) haben.

4.1.3 Gamma

Für den Decoder wird vorausgesetzt, dass alle durch den redaktionellen Inhalt definierten und aufgerufenen RGB-Werte für das Auge Gamma-vorentzerrt wurden.

4.1.4 Transparenz

Für Decoder wird gefordert, dass sie mindestens drei Transparenzpegel – undurchsichtig, halbtransparent und völlig transparent – implementieren.

Die Farbpalette ist für 30 % Halbtransparenz optimiert. Kann der Decoder den Wert von 30 % Halbtransparenz nicht implementieren, muss er ihn durch den nächsten implementierbaren Wert der Halbtransparenz ersetzen. Liegt der codierte Wert der Transparenz im Bereich von 10 % bis 90 %, darf er weder als 0 % noch als 100 % angenähert werden. So dürfen 9 % als 0 % angenähert werden, während 10 % durch einen Wert im Bereich von 10 % bis 90 % wie 30 % dargestellt werden müssen. Entsprechend dürfen 91 % als 100 % angenähert werden.

4.1.5 Farbpalette

Der vordefinierte Satz von 188 Farben wird in Tabelle 1 und Anhang B gezeigt. Die gewählten Farben besitzen eine wahrnehmungsmäßige Gleichverteilung über den Farbraum. Dem Dienstanbieter ist freigestellt, ob er Inhalte erzeugt, die nur diesen Farbensatz verwenden.

Ein Decoder muss mindestens diese Farben bis zu ihrer Auflösungsgrenze unterstützen. Es ist dann Aufgabe des Decoders, die im redaktionellen Inhalt definierten Farben mit der besten Annäherung innerhalb des vordefinierten Farbensatzes abzubilden.

Tabelle 1 – Vorgabe-Farbpalette

Transparenz- pegel	Gesamtzahl der Einträge	Quantisierungspegel für Rot	Quantisierungspegel für Grün	Quantisierungspegel für Blau
0 % (völlig undurchsichtig)	135	5 Pegel 0, 63, 127, 191, 255	9 Pegel 0, 31, 63, 95, 127, 159, 191, 223, 255	3 Pegel 0, 127, 255
0 % (völlig undurchsichtig)	4	4 zusätzliche Grauskala-Farben R = G = B = 42,85, 170, 212		
30 %	48	4 Pegel 0, 85, 170, 255	6 Pegel 0, 51, 102, 153, 204, 255	2 Pegel 0, 255
100 % (völlig durchsichtig)	1	–	–	–
durch Decoder definierbar	68			
ANMERKUNG Die Pegel sind als Dezimalwerte im Bereich von 0 bis 255 angegeben.				

Um diese Farben für die Wiedergabe eines GIF-Bildes benutzen zu können, von dem die Farbpaletten-Information fehlt, ist es erforderlich, nach Anhang B jeder Farbe einen eindeutigen Wert zuzuordnen.

4.2 Textdarstellung

Text wird unter Verwendung eines Proportionalschrift-Zeichensatzes und eines Zeichensatzes mit festem Zeichenabstand wiedergegeben, jeder in fünf Größen und zwei Schriftarten – einfach und fett. Die Schriftarten Kursiv und Fett-Kursiv sind wahlfrei. Die Zeichensätze werden in dieser Norm nicht definiert. Um die Kompatibilität zwischen den Geräten sicherzustellen, wird anstelle der Maße für jedes Zeichen für alle vorstehend erwähnten Größen die Zelle definiert.

Ein Dienstanbieter, der übereinstimmende Anzeigen sicherstellen will, muss die Inhalte unter Verwendung der gleichen Maße wie bei den implementierten Decodern erzeugen. Der Inhalt muss bei abgeschalteter Unterschneidung erzeugt werden.

4.2.1 Erforderliche Größen

Tabelle 2 zeigt die unterstützten Größen der Schriftarten für proportionalen und festen Zeichenabstand und die Zuordnung zu den Größenattributen des HTML-Schriftsatz-Tags. Die Schrift- und Zeichenmaße einschließlich der Breite (Vorschub) aller Zeichenzellen für Schriften mit proportionalem und festem Zeichenabstand sind für die verschiedenen Schriftgrößen in Anhang F angegeben.

Tabelle 2 – Schriftgrößen

Größe Pixel	HTML-Schrift-Tag
22	Größe = 1 oder 2
24	Größe = 3
27	Größe = 4
31	Größe = 5
36	Größe = 6 oder 7

4.2.2 Schriftarten Fett und Kursiv

Wird die Schriftart Fett nicht unterstützt, kann Fett durch nochmaliges Schreiben des Zeichens mit einem Offset von einem Pixel rechts von der normalen Position emuliert werden. Fett sollte auf kleine Teile des Textes, z. B. Überschriften, beschränkt werden. Es sollte nicht für den gesamten Text verwendet werden. Die emulierte Schriftart Fett kann für bestimmte Zeichen besonders bei kleinen Schriftgrößen nicht mehr akzeptabel sein (z. B. können die doppelten Anführungszeichen oder Zeichen mit Umlaut kritisch werden).

Für die Proportionalchrift darf die Breite (Vorschub) einer Zeichenzelle, die ein Zeichen Fett enthält, wie in Bild 1 gezeigt, mit maximal ein Pixel pro Zeichen vergrößert werden. Inhalte müssen unter Berücksichtigung dieser zusätzlichen Pixel erzeugt werden. Für Schrift mit festem Zeichenabstand ist die Breite aller Zeichen für eine gegebene Schriftgröße für alle Arten die gleiche.

Die Implementierung der wahlfreien Schriftarten Kursiv und Kursiv und Fett bleibt offen. Die entsprechende Breite der Zeichenzelle für die Arten Normal bzw. Fett muss jedoch immer die gleiche sein.

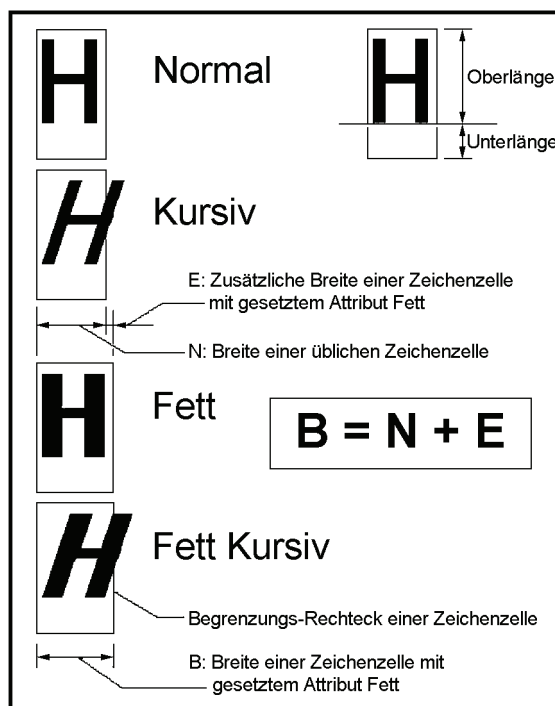


Bild 1 – Zeichenmaße

4.2.3 Zeichenvorrat

Decoder müssen den vollständigen ISO-Zeichensatz Lateinisches Alphabet Nr. 1 nach ISO 8859-1 und das EURO-Zeichen (Unicode 0x20AC) unterstützen. Das EURO-Zeichen kann als benannte Zeichenentität (&euro) oder als numerische Darstellung (#x20AC oder #8364) benutzt werden. Unterscheidet sich die Zeichencodierung der HTML-Dateien vom ISO-Zeichensatz Lateinisches Alphabet Nr. 1 oder das Zeichen ist

in ISO-Zeichensatz Lateinisches Alphabet Nr. 1 nicht definiert und ist kein EURO- Zeichen, dann muss der Decoder ein leeres Quadrat von der Größe eines großen „V“ als Ersatzzeichen anzeigen.

ANMERKUNG Die Zeichen 0x00 bis 0x1F sind im ISO-Zeichensatz Lateinisches Alphabet Nr. 1 nicht definiert, deshalb wird stattdessen ein Ersatzzeichen angezeigt.

4.2.4 Vorgabeattribute

Fehlt irgendeine Schrift, Größe, Vordergrundfarbe oder Hintergrundinformation innerhalb des redaktionellen Inhaltes, kann der Decoder seine eigenen Vorgabebedingungen einsetzen. Es wird empfohlen, dass der Decoder die in Anhang E definierten Vorgabe-Formatvorlagen (Style Sheets) beachtet.

4.3 Platzierung des Textes

4.3.1 Textbreite

Um sicherzustellen, dass Text unabhängig von der Qualität der Zeichenwiedergabe auf verschiedenen Decodern und Geräten der Verfasser identisch fließt, wird ein einfacher Algorithmus definiert, um die vertikale Platzierung und den Zeilenumbruch festzulegen. Der Fluss wird als identisch angesehen, wenn Zeilen und Wörter an derselben Zeichenposition umbrechen. Die Berechnung ermöglicht den Autoren des Inhaltes, zur Zeit des Abfassens genügend Platz für ihre Zeichenfolgen zur Verfügung zu stellen. Dies stellt keine Anforderungen an die einzelnen eingesetzten Wiedergabesysteme. Die Zeichen können Bitmap- oder Vektor-Zeichensätze, geglättet oder nicht geglättet usw. sein. Die Berechnung muss sowohl für die Geräte der Verfasser als auch für Decoder angewendet werden.

4.3.2 Berechnung der Textbreiten

Die Berechnung der Breite wird unter Berücksichtigung der begrenzenden Rechtecke der einzelnen Zeichen so durchgeführt, wie sie mit der in Bild 1 gezeigten Bemaßung der Schrift definiert sind. Für Proportionalchrift muss eine durch Schriftart Fett verursachte besondere Breite berücksichtigt werden. Die Breite einer Textfolge ist die Summe der Breite der Zeichen. Diese Berechnungen werden mit Pixelauflösung durchgeführt.

4.3.3 Zeilenumbrüche

Liefert die in 4.3.1 definierte Textbreiten-Berechnung ein Ergebnis, das den verfügbaren Platz übersteigt, muss das erste Wort nach einer Leerstelle (aber keiner nicht umbrechenden Leerstelle (0xA0)), nach einem Bindestrich (0x2D) oder nach einem weichen Bindestrich (0xAD), der nicht vollständig auf die Zeile passt, auf der Zeile darunter wiedergegeben werden. Dies beinhaltet, dass das Gerät Trennungsregeln nicht zu kennen oder anzuwenden braucht.

4.3.4 Vertikaler Zeilenabstand

Die Grundlinie einer Schrift wird durch ihre Ober- und Unterlängenmaße bestimmt (siehe Bild 1). Alle Zeichen sind vertikal auf ihre Grundlinie ausgerichtet. Der vertikale, für eine Zeile erforderliche, Abstand ist die Summe der größten Oberlänge und der größten Unterlänge in dieser Zeile plus dem Zwischenraum (externer Durchschuss). Der Zwischenraum (externer Durchschuss) zwischen Zeilen ist für alle Schriftarten und Größen auf –1 gesetzt.

4.4 Darstellung eines Bildes

4.4.1 Format

Ein Decoder muss, abgesehen von der „Klartext“-Erweiterung, die vollständige GIF-Spezifikation [GIF] implementiert haben. Eine übertragene Datei darf „Klartext“ enthalten. Zusätzlich wird das sequentielle JPEG-Basisbildformat unterstützt [HTML.3.2][HTML4.0][XHTML1.0].

4.4.2 Animation

Animation und Schleifendurchlauf von GIF-Bildern, wie in [GIF] bzw. [GIF about] beschrieben, werden unterstützt.

ANMERKUNG Für Animation oder Schleifendurchlauf wird keine Mindestzeit für die Anzeige eines Frame angegeben. Dies hängt von der Verarbeitungs-Leistung und der Komplexität und Größe der Bilder ab. Verschiedene Decoder können verschiedene Reaktionsgeschwindigkeiten haben. Die Größe eines bewegten GIF-Bildes kann durch eine Anwendungsregel beschränkt werden.

4.4.3 Benutzen in der Hintergrund-Bildebene

Ein Bild, das kleiner ist als die Hintergrund-Bildebene, sollte gekachelt werden (zeilen- und spaltenweises Wiederholen), um den verfügbaren Bereich zu füllen.

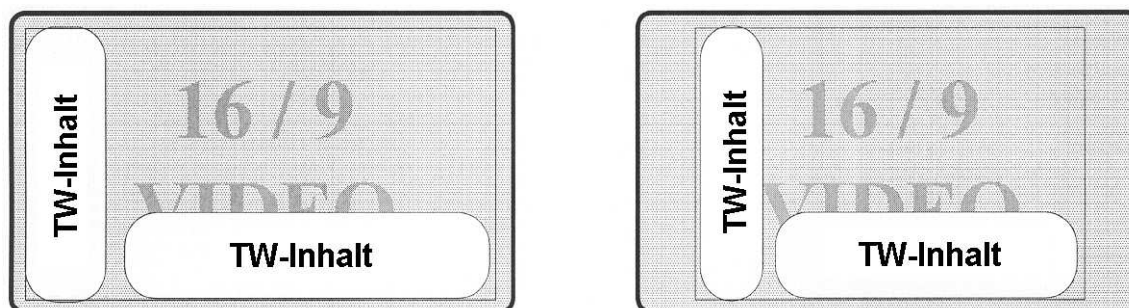
Für die Wiedergabe von Animation in der Hintergrund-Bildebene ist kein Decoder erforderlich. Sind mehrere Frames in der vorgesehenen Datei vorhanden, sollte der erste Frame ununterbrochen angezeigt werden.

4.5 Anzeigen im Bildseitenverhältnis 4:3 und 16:9

Inhalte sollten für eine 4:3-Anzeige erstellt werden. Bei Decodern mit einem Bildschirm-Seitenverhältnis von 16:9 kann die Wiedergabe des TeleWeb-Inhaltes so gewählt werden, dass sein 4:3-Seitenverhältnis beibehalten wird. Dadurch sollte ein Kreis in einem Bild auch als Kreis erscheinen.

Das zur Anzeige des TeleWeb-Inhaltes benutzte Bildseitenverhältnis liegt im Ermessen der Hersteller des Decoders. Die TeleWeb-Spezifikationen empfehlen die Anzeige der Inhaltspixel in 4×3, lassen dem Hersteller aber andere Wahlmöglichkeiten offen. Die Anzeige des Inhaltes mit einem Bildseitenverhältnis von 16:9 scheint akzeptabel zu sein (IFA 2001).

Die vorstehenden Ausführungen sind gültig, solange es zu dem Videobild keine Transparenz gibt. In diesem Fall sollte der Inhalt im Bildseitenverhältnis des angezeigten Videobildes angezeigt werden, um die Ausrichtung zum Videobild zu gewährleisten. Bei der Anzeige von 4:3-TeleWeb-Inhalten im oberen Teil eines 16:9-Videobildes können unschöne Nebeneffekte auftreten (siehe Bild 2). Der Rundfunkbetreiber kann in Kenntnis dieses Verhaltens die Zeichnungen und Bilder mit derartigem transparenten Inhalt in Abstimmung mit dem Rundfunk-Videobild vorkompensieren (z. B. senden eines Ovals, damit es als Kreis angezeigt wird).



Dem zugrunde liegenden Video korrekt zugeordneter 4:3-TeleWeb-Inhalt.

Der gleiche TeleWeb-Inhalt, der dem zugrunde liegenden Video falsch zugeordnet ist.

Bild 2 – Ausrichten von transparentem Inhalt über einem Videobild

5 Eindeutige Quelladressen (URL, en: uniform resource locator)

Eine eindeutige Quelladresse (URL) ist eine kompakte wortgetreue Bezeichnung für den Zugriff auf Information innerhalb des TeleWeb-Systems. Die URLs innerhalb des TeleWeb-Systems müssen, wie in [URI] beschrieben, mit der Syntax für Internet-URLs übereinstimmen, außer es ist anders angegeben. Es ist zu beachten, dass der URL-Zeichensatz auf die untere Hälfte des ISO-Zeichensatzes Lateinisches Alphabet Nr. 1

nach ISO 8859-1 beschränkt ist. Für die Kompatibilität mit anderen Browsern müssen URLs Zeichencodes in der oberen Hälfte codieren.

Eine URL besteht aus einem Schema und einem Schema-spezifischen Teil. Alle Zeichen in einer URL gehören zu dem ISO-Zeichensatz Lateinisches Alphabet Nr. 1 nach ISO 8859-1.

Schema: Schema-spezifischer Teil

In einer vollständigen URL folgt auf das Schema ein Doppelpunkt („:“) und die Folge des Schema-spezifischen Teils. Superteletextprofil-Decoder müssen in diesem Abschnitt nicht aufgeführte Schemata ignorieren, falls in TeleWeb-Dokumenten solche Schemata vorkommen.

Der Schemaname ist eine Zeichenfolge. Die einzigen in einem Schemanamen zulässigen Zeichen sind nur die Kleinbuchstaben „a“ bis „z“, die Ziffern „0“ bis „9“ und die Zeichen Plus, Bindestrich und Punkt, d. h., „+“, „-“ und „.“. Für robusten Betrieb muss die URLs interpretierende Software (z. B. der TeleWeb-Decoder) die Zeichen alphabetischer Großbuchstaben auf ihren entsprechenden Kleinbuchstaben umsetzen.

Der Schema-spezifische Teil ist eine Folge von sicheren Zeichen. Sichere Zeichen haben, wie in ISO 8859-1 beschrieben, eine druckbare grafische Darstellung in ISO-Zeichensatz Lateinisches Alphabet Nr. 1. Darüber hinaus ist der Satz von sicheren Zeichen beschränkt, um die reservierte Verwendung von einigen Satzzeichen zu berücksichtigen.

Alphabetische Zeichen stehen an den Positionen 41hex bis 5A, 61hex bis 7Ahex, C0hex bis D6hex, D8hex bis F6hex und F8hex bis FFhex. Deshalb dürfen nur alphanumerische Zeichen, Zeichen der Folge „\$-_.+!*’()“, und Schema-spezifische Schlüsselzeichen der Folge „;/?:@=&“ innerhalb einer URL uncodiert verwendet werden. Andererseits darf jedes Datenzeichen codiert werden. Codierte Zeichen müssen von einer die URLs interpretierende Software erkannt und decodiert werden. Ein Datenzeichen ist ein in einem Schema nicht als Satzzeichen benutztes Zeichen.

Datenzeichen-Codierung ist das Abbilden eines 8-Bit-Wertes auf eine Folge von 3 Zeichen. Das Prozentzeichen „%“ zeigt den Beginn des Code. Die nächsten beiden Zeichen sind die Hexadezimalwerte der zu codierenden Zeichen. Zum Beispiel wird eine Leerstelle als **%20** codiert, da eine Leerstelle den dezimalen Code 32 hat, der 20 hexadezimal entspricht. Ein Prozent-Datenzeichen wird als **%25** codiert, da das Prozentzeichen an Position 37 dezimal (oder 25 hexadezimal) steht. Eine unvollständige Codierungssequenz z. B. **%F=** hat keine festgelegte Bedeutung.

5.1 Datei-Benennung

Jeder Datei muss ein Namen-Attribut (12.1.2) zugeordnet werden. Dieses Attribut, eine Textfolge, muss sowohl Namen-Element als auch Typelement (Erweiterung) enthalten, z. B. „file1.html“. Dabei ist „file1“ das Namen-Element und „html“ das Typelement. Dieser Name sollte in allen Hypertext-Links auf die entsprechende Datei verweisen. Das Dateityp-Element für HTML (Text)-Dateien muss entweder „htm“ oder „html“ sein. In Tabelle 3 wird die Datei-Benennung für die Benennung der bei TeleWeb verarbeiteten Dateitypen festgelegt.

Tabelle 3 – Datei-Benennung

Inhalt-Typ	Dateinamen-Erweiterung	MIME-Typ
HTML	.htm oder .html	Text/html
GIF	.gif	Bild/gif
JPEG	.jpg oder jpeg	Bild/jpeg
ZLIB-Verzeichnis	.dic	Anwendung/x-zlib-dic

Ein Typ-Attribut (siehe 12.1.1) muss in der Datei enthalten sein und muss den Dateityp anzeigen, z. B. für HTML „Text/html“. Wird ein Typ-Attribut weggelassen oder fehlt es, sollte der Decoder versuchen, den Dateityp durch Überprüfen des Namen-Attributes zu bestimmen. Bei Widersprüchen muss das Typ-Attribut Vorrang haben. Bei Dateinamen wird nicht zwischen Groß- und Kleinschreibung unterschieden.

5.2 Zugriff über die Seitenzahl

Dieser Abschnitt beschreibt, wie über Seitenzahlen direkt auf die Seiten zugegriffen werden kann.

Der Rumpf solcher Dateinamen besteht nur aus Ziffern, z. B. 123.htm, 4545.html, 911.htm und 4444.html.

Ein Gerätehersteller muss einen Mechanismus vorsehen, mit dem diese Dateien durch Zahleneingabe direkt angewählt werden können. Zum Beispiel kann eine Datei mit dem Namen „123.html“ durch tippen der Zahlen „1“, „2“ und „3“ an der Fernbedienung direkt erreicht werden. Die Gestaltung der Benutzerschnittstelle liegt im Ermessen der Gerätehersteller. Sie können numerischen Zugriff über eine Spezialtaste, einen Menüpunkt oder irgendeine andere zur Benutzerschnittstelle passende Technik vorsehen.

Die numerisch benannten Dateien müssen in das Wurzelverzeichnis des TeleWeb-Inhaltes übertragen werden. Numerische Dateinamen in Unterverzeichnissen sind nicht über diese Zahleneingabe an der Benutzerschnittstelle zugänglich.

Alle Zahlen innerhalb einer TeleWeb-Übertragung müssen die Präfix-Bedingungen erfüllen, d. h. eine Zahl sollte niemals Präfix einer anderen Zahl sein. Zum Beispiel darf 1234.html nicht in demselben TeleWeb-Dienst vorkommen wie 1234567.html.

Die Länge der Ziffernfolge ist nur durch die maximale Länge des Dateinamens begrenzt.

5.3 Bezugsschema für TeleWeb-Datei

TeleWeb-Datei-Bezugs-URLs können absolut oder relativ sein. Eine absolute URL gibt den Dienstenamen des Inhaltenanbieters an, während eine relative URL dies nicht kann. Eine absolute URL muss einen vollständigen Pfad zu einer Ressource angeben, während eine relative URL einen Teilpfad angeben kann.

5.3.1 Absolute URLs

Eine absolute TeleWeb-URL hat folgende Form:

tw: // *Dienstname* / *Pfadname* [*?Anfrage*] [*# Fragment*]

Dabei werden Großbuchstaben und Kleinbuchstaben als gleichwertig angesehen und Teile innerhalb der Klammern dürfen weggelassen werden.

Nachstehend werden die Komponenten spezifiziert:

tw	ist der Schemaname, der den Schema-spezifischen Teil als TeleWeb-URL kennzeichnet
// <i>Dienstname</i>	kennzeichnet den Namen des Dienstes (siehe 11.1)
/ <i>Pfadname</i>	Der Pfad zu der Information, d. h. in etwa der Name der Datei. Schrägstriche innerhalb des Pfadnamens kennzeichnen hierarchische Abschnitte des Namens, erfordern aber keine spezielle Datenstruktur innerhalb der TeleWeb-Komponenten, obwohl eine Analogie zu einem Dateisystem möglich ist. Folglich ist der Schrägstrich („/“) ein reserviertes Zeichen und wenn erforderlich, muss ein Datenschrägstrich als %2F codiert werden. Darüber hinaus sind die hierarchischen Abschnittsnamen „.“ und „..“ („Punkt“ und „Punkt-Punkt“) reserviert und Daten mit diesen Namen dürfen, wenn erforderlich, mit %2E und %2E%2E angegeben werden. Der dem Dateityp-Element vorausgehende Trennpunkt gehört eindeutig zu den Daten, da er selbst kein hierarchisches Element ist. Deshalb darf ein Trennpunkt nicht codiert werden, obwohl er als Daten angesehen werden darf, wenn er eine URL bestimmt.
	Sobald die hierarchischen Abschnitte gekennzeichnet sind, werden alle „.-Abschnitte unterdrückt, dann werden alle Abschnitt-, „.-Abschnittpaare unterdrückt. Schließlich werden alle Abschnitte neu kombiniert und durch Schrägstriche getrennt, um den Ressourcenamen (Dateinamen) zu ergeben (siehe auch 5.3.4).

*Fragment* Das Fragment bezeichnet innerhalb des aufgerufenen Schriftstückes den Anker, der zum Anzeigen des verknüpften Schriftstückes angesteuert werden muss. Der Anker in dem verknüpften Schriftstück wird durch den entsprechenden Namen (für <A>-Elemente) oder ID-Attribute (für aktivierbare Bereiche; en: image maps) definiert. Der angezeigte Anker muss auf geeignete Art hervorgehoben werden.

Ist das Fragment leer oder ungültig (z. B. in dem Ziel-Schriftstück nicht vorhanden), muss der erste verfügbare Anker in der oberen linken Ecke hervorgehoben werden.

? *Anfrage* Die Anfrage definiert zusätzliche Parameter für die URL (siehe 5.3.3).

5.3.2 Relative URLs

Eine relative TeleWeb-Datei-Bezugs-URL hat die folgende Form:

[/] Pfadname [?Anfrage] [# Fragment]

Darin ist die Bedeutung der Komponenten die gleiche wie im Abschnitt über absolute URLs (5.3.1).

Beginnt die relative URL mit einem Schrägstrich, dann wird der Pfadname innerhalb des im Kontext mit dem Link benutzten aktuellen Dienstes als absolut genommen (d. h. die relative URL bezieht sich auf dessen Basisverzeichnis). Beginnt die relative URL nicht mit einem Schrägstrich, dann bezieht sie sich auf den mitgenommenen Pfad des im Kontext mit dem Link benutzten Dienstes. Ist der Pfadname der URL nicht leer, dann ersetzt er den letzten hierarchischen Abschnitt in dem übernommenen Pfadnamen. Der resultierende Pfadname wird entsprechend dem Abschnitt über absolute URLs ausgewertet.

Ankernamen und Anfrageteile werden niemals übernommen.

5.3.3 URL-Anfrage

Der Anfrageteil einer TeleWeb-URL gibt zusätzliche Parameter für die URL an. Parameter sind entweder Schlüsselworte oder Schlüsselwort-Wertepaare. Parameter werden durch Semikola getrennt. Schlüsselworte werden durch Gleichheitszeichen von Werten getrennt. Deshalb sind „;“ und „=“ im Anfrageteil reserviert. Die Reihenfolge und die Anwesenheit von Schlüsselwort-Wertepaaren ist willkürlich. Die zu erkennenden Schlüsselwort-Wertepaare werden in dem folgenden Abschnitt definiert. In einer Anfrage müssen zwei beliebige Parameter von verschiedenem Typ sein.

5.3.3.1 Die EPG-Parameter

Diese Parameter nach Tabelle 4 können zur Realisierung eines TeleWeb-EPG benutzt werden.

Tabelle 4 – EPG-Parameter

Parameter	Syntax	Format	Beispiel
Sendernetz-Betreiber	<code>cni=CNI_Code</code>	4-stelliger hexadezimaler Wert. Der Code der Land- und Sendernetz-Kennung (CNI) für die Rundfunk-Betreiber des beigefügten Programms. Wird ein Dreizeichencode benutzt, muss ihm eine 0 vorangestellt werden.	<code>cni = C380</code>
PIL	<code>pil=PIL_Code</code>	5-stelliger hexadezimaler Wert (20 bit). Der PDC PIL für das Programm. Ist das Sendernetz konform zu VPS, dann enthält dieses Feld das VPS-Label für dieses Programm. Werden weder VPS noch PDC unterstützt, muss dieses Feld Timersteuerungscodes nach ETSI TS 300231 enthalten.	<code>pi = FFFFE</code>
Startzeit	<code>sta=Start_Zeit</code>	Definiert die Start-Datum/Zeit des beigefügten Programms. Das Zeitformat entspricht ISO 8601, ausgenommen, es wird UTC vorausgesetzt. Die Verwendung der Form <code>yyyymmddThhmmss</code> wird empfohlen; der Großbuchstabe „T“ trennt das Datum von der Zeit. Es ist möglich die Zeit-Zeichenfolge durch Verringern der Auflösung zu kürzen. Z. B. ist <code>yyyymmddThhmm</code> (Sekunden nicht angegeben) gültig.	<code>sta = 20001103T2030</code>
Dauer	<code>dur=Zeit</code>	Definiert die Dauer des beigefügten Programms in Minuten.	<code>dur = 90</code>

Wird nur der CNI-Parameter definiert, sollte der Decoder zu dem Kanal gehen, auf dem CNI gesendet wird und hier bleiben, auch wenn die TeleWeb-Anwendung geschlossen ist.

5.3.3.2 Der Profilparameter

Dieser Parameter nach Tabelle 5 sichert die Kompatibilität zum Erweitern auf künftige Dienste mit höherem Profil.

Tabelle 5 – Die Profilparameter

Parameter	Syntax	Format	Beispiel
Profil	<code>Pro=Profil</code>	Der Profilparameter zeigt das Profil der Seite, auf die verwiesen wurde. Dieser Parameter sollte nicht definiert werden, wenn der Link sich auf eine Superteletextseite bezieht. Für einen Superteletextdecoder zeigt das Vorhandensein der Profilparameter an, dass der Link sich auf eine Seite mit höherem Profil bezieht. Als Ergebnis zeigt der Decoder die Profilerweiterungsseite nach 10.4 an. Das Format der Parameterwerte wird in den Spezifikations-Schriftstücken für künftige Profile definiert. Es muss beachtet werden, dass dieser Parameter nur benötigt wird, wenn die Grenze des Profilbereichs von einem tieferen zu einem höheren Profil überschritten wird.	<code>pro = ...</code>
ANMERKUNG Der Parameterwert des Beispiels kann irgendein Text sein, ausgenommen „“.			

Dieser Abschnitt ist nur informativ und bestimmt nicht die erforderlichen Dienst- und Pfadnamen und Anfrage-teile.

5.3.4 Beispiele

```
<BASE HREF=„tw://Basis/heute/Nachrichten/lokal“>
<A NAME=top></A>
```

... viel Text ...

```
<A NAME=ex-1 HREF=„tw://Sport/Football.htm“>Letzte Footballnachrichten</A>
<A NAME=ex-2 HREF=„Sport/Football.htm“> Heutiges Footballprogramm</A>
<A NAME=ex-3a HREF=„morgen/Programm.htm#Football“>morgiges Footballprogramm</A>
<A NAME=ex-3b HREF=„.../morgen/Programm.htm#Football“>morgiges Footballprogramm</A>
<A NAME=ex-4 HREF=„#top“>Rückkehr zum Anfang des Schriftstückes</A>
```

ANMERKUNG Der Zusammenhang des Schriftstückes wird durch das Element BASE gebildet zu tw://Basis/heute/Nachrichten/lokal. Der Anker EX-1 ist ein absoluter Verweis auf den Dienst „Sport“, Seite „Football.htm“. Der Anker EX-2 ist ein relativer Verweis äquivalent zu tw://Basis/heute/Nachrichten/Sport/Football.htm. Die Anker EX-3a und EX-3b sind beide äquivalent zu tw://Basis/morgen/Programm.htm#Football, dem Anker Football auf der Seite Programm.htm. Der Anker EX-4 ist ein Verweis auf den Anker TOP in dem aktuellen Schriftstück.

5.4 Schema für Teletextseiten-Verweise

Durch einen Hyperlink kann unter Verwendung der URL-Syntax auf eine Seite im Teletextdienst verwiesen werden.

ttx: // cni / Seitenzahl [/ Seiten_Subcode]

Darin ist die Folge unabhängig von der Groß- und Kleinschreibung und:

ttx	Kennzeichnet den Schema-spezifischen Teil als eine Teletextseiten-URL.
// cni	Der Code der Land- und Sendernetz-Kennung (CNI) für die Rundfunkbetreiber wird in ETSI TS 101231 definiert. Wird ein Dreizeichen-VPS-CNI-Code benutzt, muss ihm eine 0 vorangestellt werden. Der Wert 0000 muss als Fernsehkanal des TeleWeb-Dienstes interpretiert werden, in dem die URL benutzt wird.
/ Seitenzahl	Ein Wert aus drei Hexadezimalzeichen im Bereich 100 bis 8 FF, der das Magazin, Seitenzehner- bzw. Seiteneiner-Werte einer Teletextseite, wie in EN 300706 definiert, darstellt.
/ Seiten_Subcode	Ein Wert aus vier Hexadezimalzeichen im Bereich 0000 bis 3F7F, der die S4-, S3-, S2- bzw. S1-Werte eines Teletextseiten-Subcode, wie in EN 300706 definiert, darstellt. Der Wert 3 F7F muss anzeigen, dass kein spezieller Subcodewert definiert wurde. S2 hat einen gültigen Bereich von 0 bis 7 und S4 hat einen gültigen Bereich von 0 bis 3. Das Einbeziehen eines page_subcode-Elementes ist freigestellt.

Die Verarbeitung von Teletextlinks ist obligatorisch, auch wenn das Gerät mit einem Teletext-Decoder ausgerüstet ist. Dieser Punkt der Spezifikation wird durch den folgenden Absatz ergänzt und besondere Anforderungen an den Decoder hinzugefügt.

Ein Decoderhersteller sollte den Benutzern geeignete Mittel zur Verfügung stellen, durch einen Teletextlink von der Teletext-Anwendung zu der TeleWeb-Seite zurückzukehren, von der sie ursprünglich die TeleWeb-Anwendung verlassen haben. In der Zwischenzeit können die Benutzer Teletextseiten mit den Funktionen durchblättern, die ihnen durch die Teletextanwendung zur Verfügung stehen. Haben sie das Durchblättern der Teletextseiten abgeschlossen, müssen Benutzer in der Lage sein, zu der TeleWeb-Seite zurückzukehren, die sie anfänglich verlassen haben.

Es ist Sache des Decoderherstellers, ob dies mit einer besonderen Taste, Menü oder irgendeinem anderen Detail der Benutzer-Schnittstelle erreicht wird, solange es für den Benutzer eindeutig ist, wie er diese benutzen muss.

Für die Handhabung von „nicht verfügbaren“ Teletextseiten ist auch der Decoderhersteller zuständig. Eine mögliche Lösung könnte sein, die Farbe des Cursors in die Farbe der nicht verfügbaren TeleWeb-Seiten zu ändern.

Beim Arbeiten mit einem Link zu einer Teletextseite sollte ein Decoder die neueste Version der verfügbaren Seite anzeigen. Die Handhabung von durchlaufenden Folgen von Seiten liegt im Ermessen des Decoderherstellers.

5.5 Schema für NexTVView-Verweise

Eine NexTVView-Auflistung kann unter Verwendung der URL-Syntax mit einem Hyperlink aufgerufen werden.

nextview: // cni ? Filterkriterien

Dabei wird in der Folge nicht zwischen Groß- und Kleinschreibung unterschieden.

nextview	kennzeichnet die nachstehende Folge als eine nexTVView-URL
// cni	Der Code der Land- und Sendernetz-Kennung (CNI) für die Rundfunkbetreiber wird in ETSI TS 101231 definiert. Wird ein Dreizeichen-VPS-CNI-Code benutzt, muss ihm eine 0 vorangestellt werden. Der Wert 0000 muss als Fernsehkanal des TeleWeb-Dienstes interpretiert werden, dessen URL benutzt wird. Der CNI-Code kennzeichnet eindeutig den nexTVView-Dienst, weil es meistens einen nexTVView-Dienst pro Kanal gibt.
? Filterkriterien	Eine Liste der Schlüsselwort-Werte-Paare. Paare werden durch Semikola getrennt. Schlüsselwörter werden durch Gleichheitszeichen von Werten getrennt. Die Reihenfolge und die Anwesenheit von Parametern ist willkürlich. Die Schlüsselwort-Werte-Paare, die erkannt werden müssen, sind in Tabelle 10 aufgelistet. Die Interpretation der Schlüsselwort-Werte-Paare ist in EN 300707 angegeben. Der Decoder muss entsprechend den definierten Filterkriterien eine Liste erstellen. Sind keine Filterkriterien definiert, ist es Aufgabe des Decoders, die Eingabe der NexTV-Anwendung festzulegen.

Von TeleWeb-Decodern wird nicht gefordert, dass sie NexTVView unterstützen. Wird NexTVView nicht unterstützt, werden NexTVView-Links vom Decoder ignoriert.

5.6 Spezialfunktionen-URLs

Zum Anpassen der eingebetteten Browserbedienung werden drei Spezialfunktionen-URLs (siehe Tabelle 6) definiert.

Tabelle 6 – Spezialfunktionen-URLs

URL	Beschreibung
Funktion: forward	Zeigt die nächste Seite im Ablauf an
Funktion: back	Zeigt die vorhergehende Seite im Ablauf an
Funktion: home	Zeigt die Home-Page an

Ist eine der Seiten nicht verfügbar, wird keine Anzeige durchgeführt.

6 Text und Hypertext

Der Textinhalt eines TeleWeb-Dienstes beruht auf HTML [HTML3.2] [HTML4.0]. In diesem Abschnitt werden die unterstützten HTML-Tags und Attribute beschrieben.

Andere in diesem Abschnitt nicht aufgelistete HTML-Tags und Attribute müssen von den Decodern, die diesen Spezifikationen entsprechen, ignoriert werden. Sie sind in Anhang A zusammengefasst. Maßgebend ist jedoch die DTD (Document Type Definition) in Anhang D. Ein Decoder sollte nur die nicht unterstützten Tags und Attribute ignorieren und sollte doch den von solchen Tags eingeschlossenen gültigen Text, Tags oder Attribute ausführen. In den übertragenen Dateien können nicht unterstützte Tags und Attribute auftreten.

Als Hilfe für eine gleichmäßige Anzeige bei allen Decodern enthält die Definition der Tags in diesem Abschnitt obligatorische anzeigebezogene Gesichtspunkte, die in der HTML-Spezifikation [HTML3.2] [HTML4.0] nicht vorkommen. Zusätzliche Wiedergabeinformationen sind in den in Anhang E definierten TeleWeb-Vorgabe-Style Sheets (Formatvorlagen) zu finden.

6.1 TeleWeb-HTML-Dateiformat

Die Verarbeitung einer TeleWeb-HTML-Datei muss auf die Konstrukte in der TeleWeb-DTD begrenzt sein, einschließlich der benannten und numerischen Spezifikations-Zeichenfolgen (en: Character Entity References). Ausgenommen von der Interpretation sind SGML-(meta-)Konstrukte wie Auszeichnungs-Deklarationen (z. B. `<! ... >`), Verarbeitungsanweisungen (z. B. `<? ... >`, einschließlich XML-Deklarationen und Tags) und markierte Bereiche (z. B. `<[CDATA[ ... ]]>`). Diese ausgeschlossenen Elemente dürfen ignoriert oder als Daten behandelt werden. Ausgeschlossen sind auch Function Character References (z. B. `&#SPACE`, `&#RS`, `&#RE` usw.) und die Codierung von Funktionszeichen als Character Entity References. Alle Character Entity References sollten als benannte Datenzeichen interpretiert werden.

6.1.1 Tags und Attribute

Ein TeleWeb-HTML-Tag hat die allgemeine Form: `<Tag_Name Attribut_1 Attribut_2 ... .>`, dabei kennzeichnet **Tag_Name** das Tag und **Attribut_1, Attribut_2 ...** sind eine Anzahl von modifizierenden Attributen, einschließlich Null. Die spitzen Klammern `< >` sind obligatorisch. In den Teilen der HTML-Tagdefinition, die wahlfrei sind, werden z. B. Text, Attribute oder Endtags in {Klammern} gesetzt.

Für manche Tags gibt es ein komplementäres abschließendes Endtag in der Form `</Tag_Name>`. Dieser Tag-Typ wird als Container bezeichnet.

Im Allgemeinen haben innerhalb des Containers auftretende Tags keine Wirkung nach außen. Tags ohne komplementäre Endtags werden „leere Tags“ genannt.

Als Werte für Tagattribute eingegebene Textfolgen, z. B. `NAME = Textliche_Folge`, müssen in Anführungszeichen eingeschlossen werden, wenn die Folge Leerstellen oder Spezialzeichen enthält, z. B. `NAME = „Textliche Folge“`.

In der TeleWeb-DTD nicht angegebene Tags sollten ignoriert werden.

Innerhalb des Containers nicht angegebene Tags sollten innerhalb des Containers ignoriert werden.

6.1.2 Text

Jeder Text außerhalb eines Tags aber innerhalb des Rumpf-Abschnittes (siehe 6.3.4) ist als Text anzusehen, der in dem Inhaltsbereich angezeigt werden muss. Das Aussehen dieses Textes hängt von vorausgehenden Tags ab.

6.1.2.1 Leerstellen und Tabulatoren

Ausgenommen innerhalb vorformatierter Elemente (PRE) müssen Decoder alle Folgen von Leerstellen (0x20), Tabulatoren (0x09), Zeilenvorschub (0x0A) und Absatzzeichen (0x0D) innerhalb CDATA und PCDATA als einzelne Leerstelle behandeln.

Innerhalb von Attributwerten müssen führende und abschließende Leerstellen ignoriert werden.

Innerhalb vorformatierter Elemente muss ein Tabulator durch 1 bis 8 Leerstellen ersetzt werden, so, dass die letzte eingesetzte Leerstelle das *n*-te Zeichen auf der Zeile sein muss, dabei ist *n* eine ganze Zahl, multipliziert mit 8.

Innerhalb vorformatierter Elemente müssen alle Leerzeichen als Daten behandelt werden (d. h. jedes Leerzeichen muss am Ausgang erscheinen, es darf keine Vermischung entstehen). Ein Zeilenvorschub oder ein Absatzzeichen oder ein Paar von Absatzzeichen und ein Zeilenvorschub (was davon am längsten ist) müssen einen Zeilenvorschub signalisieren.

6.1.3 Dateistruktur

Jede TeleWeb-HTML-Datei darf höchstens ein Schriftstück enthalten. Das kleinste Schriftstück ist eine leere Datei.

Der allgemeine Inhalt einer HTML-Datei für TeleWeb-Anwendungen wird in Bild 3 gezeigt.

```

<!DOCTYPE HTML-TW PUBLIC „Profile 1.0“>
<HTML>
  <HEAD>
    :
    :
    :
  </HEAD>
  <BODY>
    :
    :
    :
  </BODY>
</HTML>

```

(Header (Kopf)-Abschnitt)

(Body (Rumpf)-Abschnitt)

Bild 3 – Allgemeine Inhalte einer HTML-Datei für TeleWeb-Verwendung

Das Tag <DOCTYPE> bezeichnet die Version der benutzten TeleWeb-HTML. Die korrekte Syntax für nach dieser Spezifikation konzipierte Dienste wird in 6.3.1 angegeben. Die Verwendung dieses Tags ist für nach dieser Spezifikation konzipierte Dienste nicht obligatorisch.

Die Tags <HTML> und </HTML> schließen Header- und Body-Abschnitte (Rumpfabschnitte) ein. Die Verwendung dieser Tags ist nicht obligatorisch. Ein Decoder sollte das Ende der Datei äquivalent zu einem </HTML>-Tag interpretieren.

Die Tags <HEAD> und </HEAD> umschließen den Header-Abschnitt. Die Verwendung dieser Tags ist nicht obligatorisch. Der Header-Abschnitt enthält auf das ganze Schriftstück anwendbare Daten. Dies kann den Titel des Schriftstückes einschließen. Ein Decoder muss jeden Text direkt innerhalb des Header-Abschnittes ignorieren. Zur Klarstellung: Text direkt innerhalb des Header-Abschnittes (d. h. PCDATA) kommt in der obersten Ebene vor, d. h. er ist nicht innerhalb des Header-Elementes (z. B. TITLE) eingeschlossen. Dadurch wird Titeltext nicht ignoriert, aber alle dem Titel-Starttag unmittelbar vorausgehende oder dem Titel-Endtag unmittelbar folgende Textzeichen müssen ignoriert werden.

Die Tags <BODY> und </BODY> umschließen den Body-Abschnitt. Die Verwendung dieser Tags ist nicht obligatorisch. Der Body-Abschnitt enthält in dem Inhaltsbereich anzuzeigende Text- und Hypertext-Links. Fehlt ein BODY-Tag, muss das erste Vorkommen eines in der DTD als innerhalb des Body-Abschnittes vorhanden definierten Tag durch einen Decoder als der Anfang des Body-Abschnittes interpretiert werden.

6.2 Syntax von TeleWeb-HTML-Tags

In diesem Abschnitt beschriebene TeleWeb-HTML-Tags und Attribute werden obligatorisch von jedem TeleWeb-Browser unterstützt, obwohl die Verwendung dieser Tags und Attribute in einem TeleWeb-HTML-Schriftstück freigestellt sein kann.

6.2.1 Richtlinien für Inthalteanbieter: Konformität mit XHTML

Wo zutreffend, sollten die Syntaxbeschränkungen in der XHTML-Spezifikation [XHTML 1.0] angewendet werden.

Schriftstücke sollten gut gestaltet sein. Die Verschachtelung sollte korrekt sein.

Element- und Attributnamen sollten in Kleinbuchstaben ausgeführt sein. Konforme Browser müssen jedoch Elemente und Attributnamen in Groß- und Kleinbuchstaben gleichwertig akzeptieren.

Für nicht leere Elemente sollten Endtags einbezogen werden. Konforme Browser müssen jedoch wegzulassende Endtags korrekt behandeln.

Attributwerte sollten immer angeführt werden. Konforme Browser müssen jedoch nicht angeführte numerische und alphanumerische Werte korrekt behandeln.

Attributnamen sollten immer vorhanden sein. Konforme Browser müssen jedoch unbenannte Attributwerte, bei denen der Wert mit dem Namen identisch ist und die keinen anderen Wert haben, korrekt akzeptieren, z. B. compact.

Leere Tags sollten mit /> enden. Konforme Browser sollten jedoch dieses Verhalten weder fordern noch sollten sie dadurch unterbrochen werden.

Leerstellen am Anfang oder Ende eines Attributwertes sollten als nicht signifikant angesehen werden; außerdem sollte eine Folge von Leerstellenzeichen innerhalb eines Attributwertes als äquivalent zu einem einzigen Leerstellenzeichen angesehen werden.

In Schriftstücken sollten keine CDATA-Abschnitte eingesetzt werden, die nicht unterstützt werden.

Zur Kennzeichnung von Fragmenten sollten die ID-Attribute gegenüber den abgelehnten NAME-Attributen bevorzugt werden. Für Rückwärtskompatibilität muss jedoch in Elementen, die ein Name-Attribut deklarieren, NAME mit identischem Wert benutzt werden.

6.2.2 Zeichen

Für alle Tags dürfen nur Zeichen aus dem ASCII-Zeichensatz verwendet werden.

Character Entity References werden immer als Daten behandelt. Diese Forderung ist mit der Anwendung von SGML unvereinbar, sie wurde aber durchgesetzt, um die Implementierung zu vereinfachen und das Verhalten von bestehenden Browsern anzupassen.

Es können Großbuchstaben, Kleinbuchstaben und jede Kombination davon benutzt werden.

Schlüsselworte innerhalb von Tags (z. B. style, align) müssen vollständig und dürfen nur in der festgelegten Schreibweise geschrieben werden.

ANMERKUNG Für bestimmte Schlüsselworte wird die amerikanische Schreibweise benutzt z. B. colour, centre.

6.2.3 Leerstellen

Die Leerstelle ist jede Kombination von Leerraum- (0x20), Tabulator- (0x09), Return- (0x0D) und Zeilenvorschub- (0x0A) Zeichen.

Der Tag- (oder Element-)name muss unmittelbar auf das Öffnungssymbol des Tag („<“) ohne dazwischenliegende Leerstelle folgen. Attributnamen muss eine Leerstelle vorangestellt werden und ggf. folgen. Attributwerten muss eine Leerstelle vorangestellt werden und ggf. folgen.

Zum Beispiel: <P ALIN = LEFT> ist äquivalent zu <P ALIN=LEFT>, aber <P> ist nicht zulässig.

6.2.4 SHORTTAG

Das Merkmal SHORTTAG wird nicht unterstützt. Die einzige Form der unterstützten Attribut-Minimierung gibt es nur dort, wo der legitime Attributwert mit dem Attributnamen identisch ist, z. B. COMPACT = COMPACT, darf nur als COMPACT geschrieben werden. Beispiele nicht unterstützter Notation sind <html<head><title/hello</body></></html>.

6.2.5 Attribute

Alle Attribute sind, wenn nicht anders festgelegt, wahlfrei. Es darf maximal ein Attribut von jedem erlaubten Typ pro Tag geben.

Attribute können innerhalb eines Tag in jeder beliebigen Reihenfolge aufgelistet werden.

In den Tags können nicht unterstützte Attribute auftauchen. Diese Attribute sollten aber ignoriert werden. Bei unbekannten Attributwerten sollten die Attribute ignoriert werden.

6.2.5.1 Farb-Attribute

Farben können auf zwei Arten bezeichnet werden:

- 1) „#RRGGBB“
Dabei ist RR, GG und BB die bezüglich Groß- und Kleinschreibung unempfindliche hexadezimale Darstellung der Werte im Bereich 00 bis FF, die die Amplitude der roten, grünen bzw. blauen Komponenten definieren. Beispiel: <BODY BGCOLOR = „#1E47DA“>
- 2) *standard_color*
Dabei ist *standard_color* eine der benannten nachstehend definierten Farben. Bei den Namen der Farben wird nicht zwischen Groß- und Kleinschreibung unterschieden. Beispiel: <BODY BCOLOR = „Yellow“>

Benannte Farben:

BLACK = „#000000“ GRAY = „#808080“ SILVER = „#C0C0C0“ WHITE = „#FFFFFF“ MAROON = „#800000“ RED = „#FF0000“ PURPLE = „#800080“ FUCHSIA = „#FF00FF“ GREEN = „#008000“ LIME = „#00FF00“ OLIVE = „#808000“ YELLOW = „#FFFF00“ NAVY = „#000080“ BLUE = „#0000FF“ TEAL = „#008080“ AQUA = „#00FFFF“

ANMERKUNG 1 Die Norm-HTML-Werte werden von der TeleWeb-Vorgabe-Farbpalette nicht unterstützt. Decoder, die nur diese Palette verwenden, sollten den hexadezimalen Wert 7F anstelle des Wertes 80 in den oben genannten RR-GG-BB-Werten einsetzen.

ANMERKUNG 2 Für robusten Betrieb sollte der Farbname GREY äquivalent zur Normfarbe GRAY behandelt werden.

ANMERKUNG 3 Die vordefinierte Farbe SILVER wird von der TeleWeb-Vorgabe-Farbpalette nicht unterstützt. Decoder, die nur diese Palette verwenden, sollten stattdessen die Farbe #D4D4D4 (Farbindex 138, siehe Anhang B) verwenden.

6.3 Elemente der Dokumentenstruktur

6.3.1 <!DOCTYPE>-Tag für Definition des Schriftstücktyps

Funktion: Definiert die zum Verfassen des Schriftstückes benutzte HTML-Version.

Format: <!DOCTYPE {Versionsnummer}>

Attribute: Keine

Anwendung: Für mit dieser Spezifikation kompatible Dateien muss das Tag eingegeben werden als:

<!DOCTYPE HTML-TW PUBLIC „Profile 1“> mit Leerstelle zwischen den einzelnen Zeichenfolgen.

Wenn vorhanden, muss diese Zeile die erste Zeile einer Datei sein. Sie darf in jeder Datei nicht mehr als einmal vorkommen. Ist dieses Tag vorhanden, sollte der Browser alle Tags streichen, die vom TeleWeb-Supertext-Profil nicht unterstützt werden.

6.3.2 <HTML>-Tag für Hypertext-Auszeichnungssprache

Funktion: Enthält ein HTML-Schriftstück.

Format: <HTML> ... </HTML> (Anfangs- und Endtag sind wahlfrei)

Attribute: Keine

Anwendung: Die Verwendung dieses Tags ist wahlfrei. Ist es vorhanden, wird vom Decoder keine besondere Antwort erwartet. Es darf in jeder Datei nicht mehr als einmal auftreten. Der auf das Endtag folgende Text muss ignoriert werden.

6.3.3 <HEAD>-Tag für Schriftstück-Header (Kopf)

Funktion: Enthält den Header-Abschnitt des Schriftstückes.

Format: <HEAD> ... </HEAD> (Anfangs- und Endtag sind wahlfrei)

Attribute: Keine

Anwendung: Die Verwendung dieses Tag ist wahlfrei. Ist es vorhanden, wird vom Decoder keine besondere Antwort erwartet. Es darf in jeder Datei nicht mehr als einmal auftreten. Ein Decoder muss jeden Text unmittelbar innerhalb des Header-Abschnitts ignorieren. Innerhalb der Titel-Tags enthaltener Text wird nicht ignoriert, da das Element erwartet wird und den eigentlichen Text enthält.

6.3.4 <BODY>-Tag für den Body (Rumpf) des Schriftstückes

Funktion: Enthält den Body-Abschnitt des Schriftstückes. Text innerhalb des Body-Abschnittes muss im Inhaltsbereich angezeigt werden.

Format: <BODY {Attribut_1} {Attribut_2} ... > ... </BODY> (Anfangs- und Endtag sind wahlfrei)

Attribute: ALINK, BACKGROUND, BGCOLOR, BOTTOMMARGIN, LEFTMARGIN, LINK, RIGHTMARGIN, TEXT, TOPMARGIN, VLINK, TRANSPARENCY

Anwendung: Die Verwendung dieses Tag ist wahlfrei. Ist es vorhanden, wird vom Decoder keine besondere Antwort erwartet. Es darf in jeder Datei nicht mehr als einmal auftreten. Der Inhalt des Body-Bereiches kann aus einer nicht gegliederten Sammlung irgendwelcher Tags bestehen, die beginnend mit 6.5 bis zum Ende von Abschnitt 6 definiert werden.

ALINK-Attribut

Funktion: Definiert die Vordergrundfarbe für einen Hypertext-Link, die in dem Augenblick aktiviert wird, während er auf die neue anzuzeigende Seite wartet.

Format: ALINK = *Farbe* Das Format der *Farbe* wird in 6.2.5.1 definiert.

Anwendung: Im Decoder kann die Farbe ALINK als Empfehlung behandelt werden und kann anzeigen, dass ein auf irgendeine Art gewählter Link aktiviert wurde.

BACKGROUND-Attribut

Funktion: Definiert die URL einer in der Hintergrund-Bildebene anzuzeigenden Bilddatei.

Format: BACKGROUND = TeleWeb *URL* Das Format von *URL* wird in 5.3 definiert.

Anwendung: Ist dieses Attribut nicht vorhanden, wird die Hintergrundfarbe in Abstimmung mit dem Transparenzwert angezeigt (siehe BGCOLOR). Animation von Hintergrundbildern wird nicht unterstützt.

BGCOLOR-Attribut

Funktion: Definiert die in der Hintergrundfarbenebene anzuzeigende Farbe.

Format: BGCOLOR = *Farbe* Das Format der *Farbe* wird in 6.2.5.1 definiert.

Anwendung: Mit der festgelegten Farbe muss der innerhalb des Inhaltsbereiches sichtbare Teil der Hintergrundfarbenebene vollständig gefüllt werden. Ist dieses Attribut nicht definiert, dann muss der Decoder die Hintergrundfarbe auswählen.

BOTTOMMARGIN-Attribut

Funktion: Dieses Attribut bestimmt den unteren Rand für den gesamten Rumpf der Seite in Pixel und hebt den Vorgaberand auf. Wenn auf Null (0) gesetzt, ist der untere Rand die untere Kante des Fensters, in dem der Inhalt gezeigt wird.

Format: BOTTOMMARGIN = *vorzeichenlose ganze Zahl*

LEFTMARGIN-Attribut

Funktion: Dieses Attribut bestimmt den linken Rand der Seite in Pixel und hebt den Vorgaberand auf. Wenn auf Null (0) gesetzt, ist der linke Rand die linke Kante des Fensters, in dem der Inhalt gezeigt wird.

Format: LEFTMARGIN = *vorzeichenlose ganze Zahl*

LINK-Attribut

Funktion: Definiert die Vordergrundfarbe für einen Hypertext-Link, der nicht besucht wurde und nicht aktiviert ist.

Format: LINK = *Farbe* Das Format der *Farbe* wird in 6.2.5.1 definiert.

Anwendung: Wenn dieses Attribut nicht definiert ist, dann muss der Decoder die Farbe auswählen.

RIGHTMARGIN-Attribut

Funktion: Dieses Attribut bestimmt den rechten Rand der Seite in Pixel und hebt den Vorgaberand auf. Wenn auf Null (0) gesetzt, ist der rechte Rand die rechte Kante des Fensters, in dem der Inhalt gezeigt wird.

Format: RIGHTMARGIN = *vorzeichenlose ganze Zahl*

TEXT-Attribut

Funktion: Definiert die Vorgabe-Vordergrundfarbe für den Text des Schriftstückes.

Format: TEXT = *Farbe* Das Format der *Farbe* wird in 6.2.5.1 definiert.

Anwendung: Ist dieses Attribut nicht definiert, dann muss der Decoder die Farbe auswählen.

TOPMARGIN-Attribut

Funktion: Dieses Attribut bestimmt den oberen Rand für den gesamten Rumpf der Seite in Pixel und hebt den Vorgaberand auf. Wenn auf Null (0) gesetzt, ist der obere Rand die obere Kante des Fensters, in dem der Inhalt gezeigt wird.

Format: TOPMARGIN = *vorzeichenlose ganze Zahl*

TRANSPARENCY-Attribut

Funktion: Definiert die Undurchsichtigkeit der Hintergrundfarbe vor dem darunterliegenden Videobild.

Format: TRANSPARENCY = *vorzeichenlose ganze Zahl*

Anwendung: Ist dieses Attribut nicht definiert, wird vollständige Undurchsichtigkeit angenommen.

VLINK-Attribut

Funktion: Definiert die bevorzugte Vordergrundfarbe für einen Hypertextlink, der besucht wurde.

Format: VLINK = *Farbe* Das Format der *Farbe* wird in 6.2.5.1 definiert.

Anwendung: Ist dieses Attribut nicht definiert, dann muss der Decoder die Farbe auswählen. Um die Kontrolle des Herstellers über „look and feel“ zu steigern, ist dieser Wert nur eine Empfehlung.

6.3.5 <! ... >-Tag für Kommentar

Funktion: Erlaubt redaktionellen, innerhalb einer HTML-Datei einzubettenden Kommentar.

Format: <![Text des Kommentars]..>

Attribute: Keine

Anwendung: Das Benutzen dieses Tag ist wahlfrei. Außer in der ersten Zeile kann es überall innerhalb einer Datei benutzt werden, wenn ein Tag <DOCTYPE> enthalten ist. Um künftige Rückwärtskompatibilität sicherzustellen, müssen alle Texte innerhalb von Tags von Decodern, die nach dieser Spezifikation gebaut sind, ignoriert werden.

Kommentare dürfen nicht verschachtelt werden.

6.4 Tags des Header-Abschnittes

Zwischen <HEAD> und </HEAD> sollten nur die in diesem Abschnitt definierten Tags benutzt werden. Wenn keine Tags <HEAD> vorhanden sind, sollte das erste Vorkommen eines dieser Elemente den Anfang des Header-Abschnittes kennzeichnen.

6.4.1 <LINK>-Tag für Linkstruktur

Funktion: Kennzeichnet die Beziehung des aktuellen Schriftstückes zu anderen Schriftstücken.

Format: <LINK {Attribut_1} {Attribut_2}... >

Attribute: REL, HREF, REV, TITLE

Anwendung: Dieses Tag darf nur innerhalb des Header-Abschnittes benutzt werden. Es kann dort mehrfach vorkommen. Ein Decoder darf dieses Tag wählen oder ignorieren.

Beispiele: <LINK REL = NEXT HREF = page_2.html>, dies beinhaltet, dass die Datei *page_2.html* die nächste in der Folge ist.

HREF-Attribut

Funktion: Definiert die Adresse der verknüpften Resource.

Formate: HREF = *TeleWeb-URL*, siehe 5.3
HREF = *Teletextseite-URL*, siehe 5.4
HREF = *nexTView-URL*, siehe 5.5

REL-Attribut

Funktion: Definiert die Beziehung (relationship) der verknüpften Ressource zu dem aktuellen Schriftstück. relationship kann irgendeine Folge sein.

Formate: REL = *Beziehung*

Beispiele für Beziehung: NEXT, PARENT, PREVIOUS, SAME, TOP, CONTENTS, INDEX, GLOSSARY, COPYRIGHT, HELP, SEARCH, ...

REV-Attribut

Funktion: Definiert eine umgekehrte Beziehung (relationship). Ein Link von Schriftstück A zu Schriftstück B mit REV = *Beziehung* drückt die gleiche Beziehung aus wie ein Link von B nach A mit REL = *Beziehung*. *Beziehung* kann irgendeine Folge sein.

Formate: REV = *Beziehung*

Beispiele für Beziehung: NEXT, PARENT, PREVIOUS, SAME, TOP, CONTENTS, INDEX, GLOSSARY, COPYRIGHT, HELP, SEARCH, ...

TITLE-Attribut

Funktion: Liefert einen erläuternden Titel für die verknüpfte Ressource.

Formate: TITLE = Textfolge

Das Attribut TITLE darf von einem Decoder ignoriert werden.

6.4.2 <TITLE>-Tag für Titel

Funktion: Enthält den Titel eines Schriftstückes.

Format: <TITLE> *Text des Titels* </TITLE>

Attribute: Keine

Anwendung: Dieses Tag muss benutzt werden. Ist es vorhanden, wird von einem Decoder keine besondere Antwort erwartet. Es darf in jeder Datei nicht mehr als einmal vorkommen. Der Inhalt des TITLE-Elementes sind PCDATA. Dies bedeutet, dass keine weiteren Auszeichnungen (en: mark-up) erlaubt sind.

Dennoch werden Zeichen-Entitäten (benannte Zeichen) unterstützt. Wenn ein Decoder das Darstellen eines Titels wählt, muss dieser außerhalb des Inhaltsbereiches angezeigt werden.

6.4.3 <BASE>-Tag

Funktion: Definiert eine Basis-URL zum Auflösen von relativen URLs. Ist das Tag vorhanden, muss es vor irgendeinem HEAD-Abschnitt vorkommen.

Das Anfangstag ist **notwendig** und das Endtag ist **verboten**.

Format: <BASE HREF = *URL*>

Attribute: Obligatorisches HREF-Element

Anwendung: Das Einbeziehen dieses Tag ist wahlfrei. Es liefert die Basis für das Zurückverweisen relativer URLs.

HREF-Attribut

Funktion: Definiert die Adresse der Basis.

Format: HREF = *URL*, siehe Abschnitt 5.

6.5 Paragraph formatieren

6.5.1 <P>-Tag für Paragraph

Funktion: Der eingeschlossene Text wird als Paragraph eingeordnet. Am Ende und am Anfang eines Paragraphs werden besondere Leerräume eingesetzt (siehe Abschnitt 7).

Format: <P {Attribut } > ... </P> (Das Endtag ist wahlfrei)

Attribute: ALIGN

ALIGN-Attribut

Funktion: Definiert die horizontale Anordnung des Paragraphs relativ zu den aktuellen Rändern des Schriftstückes. Die Anordnung ist nur innerhalb des Paragraphen gültig, in der sie definiert wurde.

Format: ALIGN = *Position*

- Position:**
- LEFT: Der Paragraph muss linksbündig wiedergegeben werden.
 - RIGHT: Der Paragraph muss rechtsbündig wiedergegeben werden.
 - JUSTIFY: Der Paragraph muss durch Anpassen der Leerzeichen zwischen den Wörtern links- und rechtsbündig wiedergegeben werden.
 - CENTER: Der Paragraph muss zentriert wiedergegeben werden.

Anwendung: Ist kein Attribut ALIGN vorhanden, gilt das Vorgabeattribut LEFT.

6.5.2
-Tag für Zeilenumbruch

Funktion: Führt einen Zeilenumbruch aus. Das Attribut CLEAR erlaubt, den Text relativ zum Bild zu positionieren.

Format: <BR {Attribut}>

Attribut: CLEAR

Anwendung: Der Text unmittelbar nach dem Tag muss auf der folgenden Zeile beginnen. Bestehende Anpassbedingungen bleiben erhalten. Fehlt das Attribut CLEAR, muss CLEAR = NONE angenommen werden.

CLEAR-Attribut

Funktion: Definiert, wie der Text relativ zum Bild positioniert werden sollte.

Format: CLEAR = *Position*

- Position:**
- LEFT: Löscht Fließtext an links ausgerichteten Bildern bis zum nächsten linken Seitenrand.
 - RIGHT: Löscht Fließtext an rechts ausgerichteten Bildern bis zum nächsten rechten Seitenrand.
 - ALL: Löscht den Text zwischen beiden Seitenrändern.
 - NONE: Führt ein CR durch und nicht mehr.

6.5.3 <Hn>-Tags für Überschriften, <H1> bis <H6>

Funktion: Diese Tags implementieren sechs Schriftgrößen für Überschriften von Schriftstücken; <H1> ist die größte und <H6> die kleinste.

Format: <Hn {Attribut}> ... </Hn>, dabei ist n eine ganze Zahl im Bereich von 1 bis einschließlich 6. Das schließende Tag </Hn> ist obligatorisch.

Attribute: ALIGN

Anwendung: Fehlt das Attribut ALIGN, muss der Text LEFT positioniert werden. Dies kann durch ein eingeschlossenen Tag <DIV> oder <CENTER> aufgehoben werden. Verschachtelung ist nicht zulässig.

ALIGN-Attribut

Funktion: Definiert die horizontale Ausrichtung des eingeschlossenen Textes relativ zu den aktuellen Rändern. Die Ausrichtung ist nur für die Überschriften gültig, in denen sie definiert wurde.

Format: ALIGN = *Position*

Position:	LEFT:	Die Überschrift muss linksbündig wiedergegeben werden.
	RIGHT:	Die Überschrift muss rechtsbündig wiedergegeben werden.
	CENTER:	Die Überschrift muss zentriert werden.
	JUSTIFY:	Die Überschrift muss durch Anpassen der Leerzeichen zwischen den Wörtern links- und rechtsbündig wiedergegeben werden.

6.5.4 <DIV>-Tag für Einteilungen

Funktion: Ermöglicht, ein Schriftstück als Hierarchie von Bereichen zu strukturieren.

Format: <DIV {Attribut_1}> ... </DIV>

Attribute: ALIGN

ALIGN-Attribut

Funktion: Definiert die horizontale Ausrichtung des eingeschlossenen Textes relativ zu den aktuellen Rändern. Die Ausrichtung ist nur für die Einteilung gültig, in der sie definiert wurde.

Format: ALIGN = *Position*

Position:	LEFT:	Der Text muss linksbündig wiedergegeben werden.
	RIGHT:	Der Text muss rechtsbündig wiedergegeben werden.
	CENTER:	Der Text muss zentriert werden.
	JUSTIFY:	Der Text muss durch Anpassen der Leerzeichen zwischen den Wörtern links- und rechtsbündig wiedergegeben werden.

6.5.5 <CENTER>-Tag für zentrierten Text

Funktion: Der eingeschlossene Text wird zentriert angezeigt. (Äquivalent zu dem Tag <DIV ALIGN = CENTER>.)

Format: <CENTER> ... </CENTER>

Attribute: Keine

Anwendung: Der eingeschlossene Text wird relativ zum aktuellen linken und rechten Rand zentriert angeordnet.

6.5.6 <ADDRESS>-Tag für Adresse

Funktion: Der eingeschlossene Text ist (üblicherweise) ein Name, Adresse oder andere Kontaktinformation.