



# Annotationsschicht: Definitionen und Termverwendungsinstanzen

Dokumentation

Projekt  
**Hypertextualisierung auf  
textgrammatischer Grundlage**  
([www.hytex.info](http://www.hytex.info))

Eva Anna Lenz  
Michael Beißwenger  
Sandra Wellinghoff

2004 (revidiert: 2006)

- 1 Motivation und Ziele
- 2 Vorgehensweise
- 3 Technische Umsetzung

Anhang:

- A1 Die kommentierte DTD
- A2 Die Termkandidatenliste  
der Pilotphase

## 1. Motivation und Ziele

Im HyTex-Projekt werden sowohl Definitionen von Fachtermini als auch deren Verwendungen (Termverwendungsinstanzen) im Text ausgezeichnet. Dafür wurde eine DTD entwickelt, die es erlaubt, beides unabhängig voneinander zu annotieren. Sie ist also dazu geeignet, Dokumente zu validieren, in denen nur Termverwendungsinstanzen oder nur Definitionen oder beides annotiert sind.

Bei den Definitionen werden Definiens, Definiendum und Definitor ausgezeichnet. Im Falle einer Glossardefinition werden nur Definiens und Definiendum gekennzeichnet, sofern sie nicht durch ein Interpunktionszeichen voneinander getrennt werden. In diesem Fall wird das Interpunktionszeichen als Separator annotiert. Zudem wird der Typ der Definition (Setzung, Selbstzuschreibung, Direktive, Fremdzuschreibung) erfasst. Zu jeder Termverwendungsinstanz werden die Grundform (z. B. „referentieller Link“) und die Normalform (z. B. „Link, referentieller“) mit angegeben, um später die automatische Verarbeitung zu erleichtern. Bei der Annotation definitorischer Textsegmente wird weiterhin Wert darauf gelegt, Einheiten herausgreifen zu können, die mindestens syntaktisch und nach Möglichkeit auch weitgehend kohäsiv geschlossen sind. Kontext, der über die Definition im engeren Sinne (also das Definiendum und die Teile des Definiens) hinausreicht, für das Verständnis der Definition aber unabdingbar ist (z. B. im Falle von Anaphern in Definiendums- oder Definiens-Position), muss daher in vielen Fällen mit annotiert werden. Auch diesem Umstand wurde bei der Konzeption der verwendeten Dokumentgrammatik Rechnung getragen.

Die der Annotation und Typisierung definitorischer Textsegmente zu Grunde liegenden Überlegungen sind im Projektbericht *Annotation definitorischer Textsegmente und „terminologiesensitives Linking“* (Beißwenger 2004) beschrieben. Eine Textversion der in der ersten Pilotphase annotierten Termkandidatenliste findet sich im Anhang (A2).

Die Annotation der Definitionen und Termverwendungsinstanzen dient folgenden Zielen:

- Der Erzeugung von Links von Termverwendungsinstanzen zum Glossar.
- Der Generierung von Links von Termverwendungsinstanzen zu einer Definition dieses Terminus im Vortext, falls eine solche vorhanden ist. Dadurch hat auch ein selektiv lesender Benutzer Zugriff auf genau diejenige Definition des Terminus, die der Autor seiner Verwendung des Terminus zu Grunde legt. Wenn mehrere Definitionen im Vortext vorhanden sind, wird auf der Grundlage des Typs und der Position der Definitionen im Text (automatisch) entschieden, welche Definition dem Nutzer angezeigt wird. Die Grundlage für diese Entscheidung bildet ein Inventar an Regeln zur Gewichtung „konkurrierender Definitionen“ (siehe Beißwenger 2004).
- Der Erzeugung von Links von Glossareinträgen zu den zum jeweiligen terminologischen Ausdruck in den Korpusdokumenten enthaltenen Definitionen.

## 2. Vorgehensweise

Die Annotation der Definitionen und Termverwendungsinstanzen geschieht in zwei Schritten:

1. Manuelle Annotation der Definitionen. Dabei wird gleichzeitig die Termkandidatenliste aktualisiert: Alle Termini, die als Definiendum einer Definition annotiert werden, werden der Termkandidatenliste hinzugefügt, falls diese sie nicht bereits enthält.
2. Validierung gegen die DTD.

3. Auf der Grundlage der Termkandidatenliste wird die Annotation der Termverwendungsinstanzen automatisch durch das Perl-Skript `tvi-tagger.pl` durchgeführt.<sup>1</sup>

Die Termkandidatenliste ist eine Liste aller in den Pilottexten definierten Termini in der Grundform zusammen mit ihrer jeweiligen Normalform und allen flektierten Formen. Sie liegt in einem XML-Format vor und kann gegen eine DTD validiert werden. Die Aktualisierung der Termkandidatenliste im ersten Schritt ist wichtig, da sie nicht nur hier im dritten Schritt verwendet wird, sondern zudem als Grundlage für die Modellierung des terminologischen Wortnetzes (*TermNet*, vgl. Beißwenger, Storrer & Runte 2004) dient. In Schritt drei können alle Termini in den Texten anhand der flektierten Formen in der Termkandidatenliste automatisch gefunden und annotiert werden. Wird die Termkandidatenliste nachträglich geändert, so kann das Perl-Skript erneut auf die Datei angewendet werden, wobei ggf. auch Löschungen durchgeführt werden.

Die DTD lässt prinzipiell andere Vorgehensweisen zu, z. B. könnte Schritt drei auch vor Schritt 1 erfolgen.

### 3. Technische Umsetzung

Als Dokumentgrammatik wurde eine DTD entwickelt. Prinzipiell hätte die Verwendung von XML Schema es erlaubt, verschiedene Inhaltsmodelle weiter einzuschränken, z. B. hätte man mit XML Schema ausdrücken können, dass `<defComplex>` mindestens zwei Instanzen des Elements `<definiensAlt>` enthalten muss. Dies ist aufgrund der Restriktionen bei gemischten Inhaltsmodellen mit DTDs nicht möglich. Wir haben uns dennoch für die Verwendung einer DTD entschieden, damit eine größere Bandbreite von Annotationseditoren verwendet werden kann. Die Restriktionen können mittels eines dafür entwickelten Schematron-Skripts überprüft werden.<sup>2</sup>

Die Anforderung, mit der DTD Termverwendungsinstanzen und Definitionen getrennt annotieren zu können, hat zur Liberalisierung des Inhaltsmodells des Elements `<definiendum>` geführt (s. Kommentar in der DTD).

### Erwähnte Literatur

- Beißwenger, Michael (2004): Annotation definitorischer Textsegmente und „terminologiesensitives Linking“. Arbeitsbericht. Online unter <http://www.hrz.uni-dortmund.de/~hytex/hytex/Publikationen/deflink.pdf>.
- Beißwenger, Michael, Storrer, Angelika & Runte, Maren (2004): Modellierung eines Terminologienetzes für das automatische Linking auf der Grundlage von WordNet. In: Kunze, Claudia; Lemnitzer, Lothar; Wagner, Andreas (Hrsg.): Anwendungen des deutschen Wortnetzes in Theorie und Praxis. Beiträge des GermaNet-Workshops Tübingen, Oktober 2003 (LDV-Forum – Zeitschrift für Computerlinguistik und Sprachtechnologie 19. 1/2), 113-125.

---

<sup>1</sup> Aufruf in der Demo: `make tiffi`

<sup>2</sup> Aufruf in der Demo: `make checkDefs`

## Anhang A 1: Die kommentierte DTD

```
<!--  
Projekt HyTex, DFG-Forschergruppe "Texttechnologische Informationsmodellie-  
rung"  
http://www.hytex.info/
```

```
=====  
Kommentierte DTD zur Annotation von definitorischen  
Textsegmenten und Termverwendungsinstanzen  
(v. 2.06)  
=====  
-->
```

```
<!--  
Die DTD ist so konzipiert, dass Termverwendungsinstanzen und  
Definitionen in verschiedenen Annotationsprozessen getrennt  
voneinander annotiert werden koennen. Die DTD kann auch zur Validierung  
nur eines der beiden verwendet werden.
```

An vielen Stellen war daher und aufgrund der Beschraenkungen von DTDs (insbesondere im Fall gemischter Inhaltsmodelle) notwendig, die Inhaltsmodelle sehr allgemein zu fassen. Wo inhaltliche Beschraenkungen notwendig sind, die nicht in der DTD ausgedrueckt werden koennen, wurde dies dokumentiert.

Version 1.0: November 2003  
Version 2.0: Juni 2004  
Version 2.01: November 2004  
Version 2.02: April 2005  
Version 2.03: Mai 2005  
Version 2.04: August 2005  
Version 2.05: August 2005  
Version 2.06: Oktober 2005

Aenderungen in version 2.06:

- \* Kommentare zu den Aenderungen ergaenzte
- \* Elemente `hypernym`, `meronym`, `holonym` und `defIndicator` gelöscht, da sie nicht benötigt wurden

Aenderungen in Version 2.04:

- \* Fehler im Element `glossDef` behoben

Aenderungen in Version 2.04:

- \* Elemente `hypernym`, `meronym`, `holonym` hinzugefügt

Aenderungen in Version 2.03:

\* Element g l o s s D e f hinzugefügt

Aenderungen in Version 2.02:

\* Element d e f I n d i c a t o r hinzugefügt

Aenderungen in Version 2.01:

Anpassung der Struktur von Version 2.0 an die Erfordernisse der Studie  
"Auffindung definitorischer Textsegmente - manuell vs. automatisch-  
patternbasiert":

\* Aenderung der Auftretensrestriktion des Attributs t y p e zu den Elemen-  
ten d e f und d e f C o m p l e x auf IMPLIED (weil in der Vergleichsstudie  
zunaechst nicht benoetigt)

\* Hinzufuegung der Elemente d f n und d f n S e g m e n t (um Definitoren -  
ggf. auch diskontinuierlich - auszeichnen zu koennen)

\* Aenderung Auftretensrestriktionen der Attribute b a s e F o r m und n o r  
m a l F o r m zum Element t e r m auf IMPLIED (weil in der Vergleichsstudie  
keine automatische Annotation von Termverwendungsinstanzen vorgesehen ist)

Aenderungen in Version 2.0:

\* Hinzufuegung des Elements d e f S e g m e n t

\* Abschaffung des Elements d e f C l u s t e r (da dieses in d e f S e g m  
e n t mit abgebildet werden kann)

Kommentar von

Michael Beißwenger (November 2003 / Juni 2004 / November 2004)

Eva Anna Lenz (November 2003 / Mai 2005 / August 2005)

Sandra Wellinghoff (Januar 2006)

-->

<!--

=====

ELEMENT: Wurzelement d e f i n i t i o n s

=====

-->

<!ELEMENT definitions (#PCDATA | defSegment | glossDef | term)\*>

<!-- Das Wurzelement d e f i n i t i o n s beschreibt ein Dokument, in  
welchem Termverwendungsinstanzen (TVIs) und/oder definitorische Textsegmen-  
te asgezeichnet wurden. Durch das sehr allgemein gehaltene Inhaltsmodell  
kann die Struktur entweder nur fuer die Auszeichnung definitorischer Text-  
segmente oder nur fuer die (automatische) Auszeichnung von TVIs oder fuer  
beides (und fuer die anschliessende Validierung der Instanzen) verwendet  
werden.

-->

<!--

```

=====
ELEMENT: d e f S e g m e n t
=====
-->

<!ELEMENT defSegment (#PCDATA | def | defComplex | term)*>

<!-- Das Element d e f S e g m e n t beschreibt solche Textsegmente, in de-
nen mindestens eine Definition oder ein Definitionskomplex enthalten ist. d
e f S e g m e n t kann jedoch gegenueber Instanzen der Elemente d e f (fuer
Definitionen) bzw. d e f C o m p l e x zudem noch textuellen Kontext umfas-
sen. Impetus bei der Beschreibung einzelner Textteile als d e f S e g m e n
t s ist es, fuer die HyTex-Anwendung (Demo) textuelle Einheiten herausgrei-
fen zu koennen, die folgende Kriterien erfuellen:
(a) Sie sind mindestens syntaktisch abgeschlossen (d.h.: repraesentieren
entweder einen ganzen Satz oder ein Syntagma, das auch ohne Kontext autonom
verstaendlich ist und somit fuer die Konzeptualisierung des darin definier-
ten Terminus hilfreich sein kann),
(b) Sie sind idealitaer auch kohaesiv geschlossen (d.h.: in Faellen, in
welchen die definiens- oder die definidendum-Position mit einer Anapher be-
setzt ist, umfasst das betreffende d e f S e g m e n t nach Moeglichkeit
auch noch den zur Kohaerenzbildung noetigen Kontext - ein d e f S e g m e n
t sollte jedoch (pi mal Daumen) maximal drei Saetze umfassen, damit der in
der Anwendung angebotene Textausschnitt kompakt bleibt.
Das Element d e f S e g m e n t wurde der Struktur im Juni 2004 hinzuge-
fuegt und bildet das Container-Element fuer alle Definitionen und Definiti-
onskomplexe.
-->

<!--
=====
ELEMENT: d e f
=====
-->

<!ELEMENT def (#PCDATA | term | definiendum | definiens | definiensSegment
| dfn | dfnSegment)*>
<!ATTLIST def
    type (Setzung | Selbstzuschreibung | Direktive | Fremdzuschreibung)
#IMPLIED
>

<!-- Das Element d e f beschreibt einzelne definatorische Textsegmente.
Einzelnen Instanzen des elements d e f wird jeweils (obligatorisch) ein t y
p e zugewiesen, dessen zulaessiger Wertebereich die Bezeichnungen der von
MB beschriebenen Sprachhandlungstypen umfasst. Die t y p e zugewiesenen

```

Werte sind zentral fuer die spaetere automatische Gewichtung von Definitionen im Falle von "Definitionen-Konkurrenz" in ein- und demselben Text. Aufgrund oder Variabilitaet bei der Vertextung von Definitionen ist das Inhaltsmodell von `d e f` recht allgemein gehalten: Sowohl ist die Reihenfolge des Auftretens der Kindelemente `d e f i n i e n d u m` und `d e f i n i e n s` beliebig, als auch kann das `d e f i n i e n s` diskontinuierlich auftreten (in Form zweier oder mehrerer Vorkommnisse des Elements `d e f i n i e n s S e g m e n t`). Weiterhin kann das Element `d e f` auch PCDATA beinhalten (naemlich solche Textteile, die nicht zur eigentlichen `d e f`-Struktur gehoeren, aber die z.B. zwischen zwei Vorkommnissen von `d e f i n i e n s S e g m e n t` stehen).

Durch Einfuehrung des Elements `d f n` ist es ab v. 2.01 auch moeglich, Definitoren auszuzeichnen. Da Definitoren ebenfalls diskontinuierlich auftreten koennen, gibt es zu diesem Element die Variante `d f n S e g m e n t` (analog zu handhaben wie die Variante `d e f i n i e n s S e g m e n t` zum Element `d e f i n i e n s`).

Beispiele fuer die Variation bei der Vertextung von Definitionen:

Es sind vier Strukturtypen von definitorisches Textsegmenten moeglich, die sich in der Abfolge von Definiens und Definiendum unterscheiden:

[[[ Legende: `*...*` markiert in den nachfolgenden Beispielen das definiendum, `#...#` markiert das Definiens bzw. DefiniensSegmente]]]

(1) definiendum vor definiens

Bsp.: `*Links*` sind `#computerverwaltete Zuordnungen zwischen Ankern#`.

(2) definiendum vor diskontinuierlichem (d.h.: durch anderen Text durchbrochenem) definiens

Bsp.: Unter `*Annotationen*` werden in der Hypertextliteratur `#Anmerkungen und Notizen#` verstanden, `#die ein Hypertextnutzer waehrend des Rezeptionsvorgangs zu den Inhalten eines Moduls anbringt#`.

(3) definiens vor definiendum

Bsp.: `#Computerverwaltete Zuordnungen zwischen Ankern#` bezeichnet man als `*Links*`.

(4) diskontinuierliches definiens umschliesst das definiendum:

Bsp.: In der Hypertextliteratur werden solche `#Anmerkungen und Notizen#` als `*Annotationen*` bezeichnet, `#die ein Hypertextnutzer waehrend des Rezeptionsvorgangs zu den Inhalten eines Moduls anbringt#`.

Beschaerungen fuer das Element `d e f`:

def enthaelt

- (1) mindestens ein definiendum-Element,
- (2) entweder genau ein definiens-Element oder mindestens zwei definienssegment-Elemente,
- (3) entweder genau ein dfn-Element oder mindestens zwei dfnsegment-Elemente.

Prinzipiell kann das Element def zwei Vorkommnisse des Kindelements definiendum beinhalten, naemlich in solchen Faellen, in welchen Benennungsalternativen angegeben werden; Beispiel:

#Filter, die ueber etadaten zu Modulen und Links operieren#, nenne ich im Weiteren \*metadatenorientierte Filter\*, kurz: \*Metafilter\*.

-->

<!--

=====

ELEMENTE: defComplex

=====

-->

<!ELEMENT defComplex (#PCDATA | term | definiendum | dfn | dfnSegment | definiensAlt)\*>

<!ATTLIST defComplex

type (Setzung | Selbstzuschreibung | Direktive | Fremdzuschreibung)

#IMPLIED

>

<!-- Erlaeuterung zum Element defComplex: Ein Definitions-Komplex liegt dann vor, wenn in eine syntaktische Struktur zwei alternative Definiertes zu ein und demselben Terminus eingebettet sind und der Terminus (als Definiendum) dabei nur einmal genannt wird.

Beispiel:

Hypertext-Module werden in der Werkstattsprache des WWW oft als "Seiten" bezeichnet, wobei "\*Seite\*" einmal #ein am Bildschirm sichtbares Objekt# (Seite-Screen) ein anderes Mal #die von einem WWW-Server verwaltete Dateneinheit# (Seite-Datei) benennt.

Beschraenkungen fuer das Element defComplex:

defComplex enthaelt

- (1) mindestens ein definiendum-Element,
- (2) mindestens zwei definiensAlt-Elemente sowie
- (3) entweder genau ein dfn-Element oder mindestens zwei dfnsegment-Elemente.

-->

<!--

=====

ELEMENT: d e f i n i e n d u m

=====

-->

<!ELEMENT definiendum (#PCDATA | term)\*>

<!ATTLIST definiendum

baseForm CDATA #IMPLIED>

<!-- Eine Instanz des Elements d e f i n i e n d u m enthaelt nach dem au-  
tomatischen

Annotierungsprozess der Termverwendungsinstanzen normalerweise

genau ein t e r m-Element. In diesem Fall wird das Attribut

b a s e F o r m von d e f i n i e n d u m nicht ausgewertet, sondern nur

das von

t e r m. In Faellen, in denen das Definiendum jedoch keinen

Terminus aus der Termkandidatenliste enthaelt, da dieser

im Definitionstext nur implizit vorkommt, enthaelt d e f i n i e n d u m

nach der

automatischen Annotierung kein t e r m-Element. Fuer diesen Fall muss

das Attribut b a s e F o r m von d e f i n i e n d u m gesetzt werden, um

auch diese

Termverwendungen spaeter automatisch auffinden zu koennen.

Beispiel:

<defSegment>

Hyperlinks koennen nach ihrem Zielpunkt kategorisiert werden, je nach

dem, ob <def type="Fremdzuschreibung"> <definiens>eine Stelle in der

gleichen informationellen Einheit</definiens> &ndash;&gt;

(<definiendum baseForm="intrahypertextueller

Link">intrahypertextuell</definiendum>)

</def>, <def type="Fremdzuschreibung"> <definiens>eine Informationseinheit

in der

gleichen Hypertextbasis</definiens>

(<definiendum baseForm="interhypertextueller

Link">interhypertextuell</definiendum>)

</def> oder <def type="Fremdzuschreibung"> <definiens>in einer anderen

Hypertextbasis</definiens>

(<definiendum baseForm="extrahypertextueller

Link">extrahypertextuell</definiendum>)

</def> referenziert wird (Kuhlen 1991: 107f.; Kennedy/Musciano 1999:

213;215ff.).

</defSegment>

Beschränkungen fuer das Element `definiendum`:

\* enthaelt null bis ein `term`-Elemente

-->

<!--

=====

ELEMENT: `definiens`

=====

-->

<!ELEMENT `definiens` (#PCDATA | `term`)\*>

<!--

Im `definiens` wird das `definiendum` definiert.

Es kann kontinuierlich oder diskontinuierlich auftreten.

Ist es dem `definiendum` vor- oder nachgestellt, wird das Element `definiens` zur Auszeichnung verwendet. Tritt es dagegen diskontinuierlich auf, wird das Element `definiensSegment` verwendet.

Beispiel für ein kontinuierliches `definiens`:

(1) `definiendum` vor `definiens`

Bsp.: `*Links*` sind `#computerverwaltete` Zuordnungen zwischen `Ankern#`.

(2) `definiendum` nach `definiens`

Bsp.: `#Computerverwaltete` Zuordnungen zwischen `Ankern#` heißen `*Links*`.

-->

<!--

=====

ELEMENT: `definiensSegment`

=====

-->

<!ELEMENT `definiensSegment` (#PCDATA | `term`)\*>

<!--

Das `definiensSegment` wird verwendet, wenn das `definiens` diskontinuierlich auftritt.

Beispiel für ein diskontinuierliches `definiens` (das `definiens` wird durch "verstanden" unterbrochen):

Unter `*Annotationen*` werden in der Hypertextliteratur `#Anmerkungen` und `Notizen#` verstanden, `#die` ein Hypertextnutzer während des Rezeptionsvorgangs zu den Inhalten eines Moduls anbringt\*.

-->

<!--

=====

ELEMENT: d e f i n i e n s A l t

=====

-->

<!ELEMENT definiensAlt (#PCDATA | term)\* >

<!--

Werden in einer Definition zu einem definiendum verschiedene - also alternative - definiens angeboten, werden sie jeweils mit dem Element d e f i n i e n s A l t ausgezeichnet.

-->

<!--

=====

ELEMENT: d f n

=====

-->

<!ELEMENT dfn (#PCDATA) >

<!--

Mit diesem Element wird der definator in einer Definition ausgezeichnet: Er etabliert die Zuordnungsbeziehung zwischen definiens und definiendum.

-->

<!--

=====

ELEMENT: d f n S e g m e n t

=====

-->

<!ELEMENT dfnSegment (#PCDATA) >

<!--

Tritt der definator wie im folgenden Beispiel diskontinuierlich auf, werden seine Bestandteile mit d f n S e g m e n t ausgezeichnet.

Beispiel (definator = bezeichnen als; unterbrochen durch "ich Links"):

Im Folgenden bezeichne ich Links als computerverwaltete Einheiten zwischen Ankern.

-->

```

<!--
=====
ELEMENT: g l o s s D e f
=====
-->

<!ELEMENT glossDef (#PCDATA | definiendum | separator | definiens)*>
<!ATTLIST glossDef
    type (Setzung | Selbstzuschreibung | Direktive | Fremdzuschreibung)
#IMPLIED
>
<!-- Das Element g l o s s D e f zeichnet sog. Glossardefinitionen aus und
wird anstelle von d e f verwendet,
wenn die Definition nicht in den Text eingebettet ist, sondern sich typo-
graphisch/strukturell vom Kontext optisch abhebt. Zudem ist häufig kein de-
finitor nachweisbar. In manchen Fällen werden definiendum und definiens
durch Interpunktionszeichen getrennt, die dann als separator ausgezeichnet
werden.
-->

<!--
=====
ELEMENT: s e p a r a t o r
=====
-->
<!ELEMENT separator (#PCDATA)>

<!--
In Glossardefinitionen (g l o s s D e f) wird der s e p a r a t o r verwen-
det, um die Interpunktionsmittel auszuzeichnen, die anstelle eines defini-
tors das definiendum vom definiens separieren.
-->

<!--
=====
ELEMENT: t e r m
=====
-->

<!ELEMENT term (#PCDATA)>
<!ATTLIST term
    baseForm CDATA #IMPLIED
    normalForm CDATA #IMPLIED>

```

## Anhang A 2: Die Termkandidatenliste der Pilotphase

1. 1:1-Beziehung
2. 1:1-Link
3. 1:m-Beziehung
4. 1:n-Link
5. 1-Anker-Link
6. actuate
7. aktionsauslösender Link
8. am Link teilnehmen
9. Anker
10. Annotation
11. arc
12. arcrole
13. association
14. association role type
15. Assoziation
16. assoziative Verknüpfung
17. ausgehende Kante
18. Auszeichnung
19. Authoring Tool
20. Autorenwerkzeug
21. Backtracking
22. bidirektionaler Link
23. Bookmarks
24. bread crumbs
25. Brotkrumen
26. Browser
27. Cascading Style Sheets
28. Chronik
29. content management
30. cross-media-Publishing
31. Datenfilter
32. datenorientierter Filter
33. deduktiver Link
34. Document Object Identifier
35. DOI
36. Dokument-Instanz
37. Dokumenttyp-Definition
38. DTD
39. dynamischer Pfad
40. einfacher Link
41. einfaches Dokumenten-Modul
42. eingebetteter Link
43. eingehende Kante
44. Elementtyp
45. End-Ressource
46. entfernte Ressource
47. erweiterter Link
48. E-Text
49. Expert Hypertext
50. eXtensible Markup Language

51. eXtensible Style Language
52. extensional definierter Link
53. extrahypertextueller Link
54. extra-referenzieller Link
55. Filter
56. Filterung
57. Filterwerkzeug
58. forced march
59. frame-link
60. from
61. geführte Tour
62. getypter Link
63. globaler Anker
64. guided tour
65. history
66. href
67. HTML
68. Hyperdokument
69. Hyperlink
70. Hypertext
71. HyperText Markup Language
72. Hypertextbasis
73. Hypertextdokument
74. Hypertextsystem
75. Hypertrail
76. ikonischer Link
77. immanenter Link
78. Index
79. individueller locator
80. informationelle Einheit
81. Informationseinheit
82. Informationsmodellierung
83. inhaltliche Kontextualisierungshilfe
84. Inhalts-Link
85. intensional definierter Link
86. interhypertextueller Link
87. intrahypertextueller Link
88. invisible link
89. Kante
90. Katalog
91. kausaler Link
92. Komponente
93. konforme XLink-Anwendung
94. konklusiver Link
95. Kontextualisierungshilfe
96. konzeptionelle Ebene
97. konzeptioneller Link
98. konzessiver Link
99. label
100. leeres Element
101. Lesezeichen
102. Link

103. Link mit eingebetteter Anzeige
104. Link mit ersetzender Anzeige
105. Link mit paralleler Anzeige
106. link typing
107. Linkanzeiger
108. Link-Bank
109. Link-Datenbank
110. Link-Etikett
111. Link-Explikation
112. Linking-Muster
113. Link-Kennzeichnung
114. locator
115. lokale Ressource
116. lokaler Anker
117. Medienobjekt
118. Metadaten
119. metadatenorientierter Filter
120. Meta-DTD
121. Metafilter
122. Metasuchmaschine
123. Modul
124. multidirektionaler Link
125. Multiple-Media-Publishing
126. multiple-media-publishing
127. n:1-Beziehung
128. n:m-Beziehung
129. n:m-Link
130. navigatorische Kontextualisierungshilfe
131. nicht traversierbarer Link
132. nicht valide Instanz
133. objektiver Link
134. Pfad
135. Portal
136. pragmatischer Link
137. Präsentations- und Interaktionsebene
138. progressiver Link
139. Ranking
140. RDF
141. referenzielle Verknüpfung
142. referenzieller Link
143. Relation
144. resource
145. Resource Description Framework
146. Ressource
147. retrospektive Hilfe
148. Retrospektive Hilfe zur Navigation
149. rhetorisch-illokutiver Link
150. role
151. Rücksprung
152. script-link
153. Seite
154. semantic attribute

155. semantic web
156. semantische Rolle
157. semantischer Link
158. semantisches Attribut
159. semantisches Web
160. separat angeordneter Link
161. SGML
162. SGML-Deklaration
163. show
164. Sicht
165. Sichten-Modul
166. Site
167. Speicherebene
168. Start-Ressource
169. statischer Pfad
170. Struktur-Link
171. subjektiver Link
172. symbolischer Link
173. syntagmatische Verknüpfung
174. syntaktischer Link
175. Tag
176. Template
177. temporärer Link
178. thematischer Link
179. theme
180. Themenbereich
181. Third-Party-Kante
182. title
183. to
184. Topic
185. topic
186. Topic Maps
187. topic occurrence
188. Topik-Anker
189. trail
190. Traversierung
191. Traversierungsattribut
192. type
193. typisierte Verknüpfung
194. typisierter Link
195. Typisierung
196. Typisierung von Links
197. Überblickshilfe
198. unidirektionaler Link
199. Uniform Resource Identifier Reference
200. URI-Referenz
201. valide Instanz
202. Verhaltensattribut
203. verknüpfender Link
204. Verknüpfung
205. Verknüpfungsanzeiger
206. Verknüpfungszielpunkt

- 207. Waisen
- 208. Web
- 209. wohlgeformtes Dokument
- 210. World Wide Web
- 211. WWW
- 212. XLink
- 213. XLink-Anwendung
- 214. XLink-konformes XML-Element
- 215. XLink-Link
- 216. XLink-Link-Element
- 217. XML
- 218. XML Linking Language
- 219. XML Path Language
- 220. XML Pointer Language
- 221. XML Topic Maps
- 222. XPath
- 223. XPointer
- 224. XSL
- 225. XTM
- 226. zusammengesetztes Modul