

DIPLOMARBEIT

# *Link Description Font.*

Ein grafisches System  
zur Beschreibung von Hypertextlinks im WWW.

eingereicht von:

Robert Zinner  
Rosenstrasse 14  
07749 Jena

[robert.zinner@izake.de](mailto:robert.zinner@izake.de)  
Matrikel-Nr.: 961820

betreut durch:

Prof. Dr. Jens Geelhaar

Prüfer:

Prof. Dr. phil. Walter Bauer-Wabnegg

an der:

Bauhaus-Universität Weimar  
Studiengang Mediengestaltung



# Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbständig angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher noch keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Jena, den 30. Oktober 2002

Robert Zinner



# *Gliederung.*

*Wohin ich gehe, weiß ich erst, wenn ich da bin.*

Zum Gegenstand der Arbeit	11
Die Problematik der Arbeit	13

*Link ist nicht gleich Link.*

Hypertext und Link – Begriffsdefinitionen	19
Erzeugung und Präsentation des Links im WWW	23

*Nach dem Klick ist vor dem Klick.*

Bestehende Ansätze	29
Das Beschreibungssystem	35
Die Beschreibungsmatrix	43

*Von der Beschreibung zum Zeichen und wieder zurück.*

Bestehende Interfacemodelle	47
Entwurf des grafischen Interfaces	53
Entwurf der Zeichenkodierung	57

## *Von Möglichkeiten und Machbarkeiten.*

- 63 Die optimale technische Lösung
- 65 Annäherungen an das Optimum

## *Es werde Schrift.*

- 71 Die Entwicklung der Zeichen
- 77 Die finalen Zeichen
- 83 Der Link Description Font
- 89 Festlegungen zum Umgang mit dem Link Description Font

## *Zum Schluß heißt nicht am Ende.*

- 93 Zusammenfassung und kritische Betrachtung
- 95 Ausblick

## *Teile des Ganzen.*

- 99 Tafeln der Einzelzeichen
- 115 Literatur
- 119 Inhalt der CD-ROM
- 119 Der Link Description Font im WWW



*Wohin ich gehe,  
weiß ich erst,  
wenn ich da bin.*

Das vorliegende Kapitel beschreibt den Gegenstand der Arbeit und erläutert die zugrunde liegende Problemstellung der Arbeit.



# Zum Gegenstand der Arbeit

Diese Arbeit beschäftigt sich mit Links im Hypertextsystem des World Wide Web. Dabei soll vor allem der spezielle Aspekt der Beschreibung von Links betrachtet werden.

Es wird ein grafisches System entworfen, das einerseits dem Autor eines Hypertextes die Möglichkeit gibt, die Links mittels bestimmter Zeichen zu beschreiben und dadurch andererseits dem Leser bei der Orientierung innerhalb eines Hypertextes hilft.

Das System soll vorrangig in komplexen wissenschaftlich-technischen Hypertexten als ein unterstützendes und hilfreiches Mittel sowohl für den Autor als auch für den Leser Verwendung finden.

Um dem Anspruch eines Systems gerecht zu werden, umfasst die Arbeit neben dem Entwurf der Zeichen auch eine zugrunde liegende Klassifizierung und Definition der Eigenschaften von Links.



# Die Problematik der Arbeit

Das World Wide Web (WWW), ein auf dem Internet basierender Informationsraum, ist eines der wichtigsten Informations- und Kommunikationsmedien unserer Zeit. Es vereinigt mittlerweile eine Informationsmenge in sich, die nicht mehr quantifizierbar ist und die von Tag zu Tag größer wird. Diese Tatsache allein unterscheidet das WWW jedoch noch nicht wesentlich von anderen Medien zur Informationsvermittlung und -speicherung; zum Beispiel den klassischen Printmedien. Die spezielle Eigenschaft, die dem WWW als Medium zu seiner Bedeutung und Einzigartigkeit verholfen hat, ist die Möglichkeit der direkten und beliebigen Verknüpfung von Informationen. Das Informationsmanagementsystem, das derartige Verknüpfungen von Informationen ermöglicht, ist die Idee des Hypertextes.

Hypertext ist zunächst einmal nichts anderes als Text, der Verknüpfungen zu anderen Informationen enthält. (Eine genaue Definition, soll im nächsten Kapitel vorgestellt werden.) Diese Verknüpfungen werden Links genannt. Links ermöglichen dem Autor einer Informationseinheit, diese mit anderen Informationseinheiten zu verbinden. Dadurch entstehen ganz neue Formen des Zugangs zu Informationen. Im Gegensatz zu traditionellen, sequentiellen Informationssystemen ermöglicht es der Hypertext, nichtsequentielle Informationssysteme aufzubauen, die sowohl den Autor eines solchen Informationssystems als auch den Leser vor neue Herausforderungen in Fragen der Informationsvermittlung und -aufnahme stellt.

Neben diesen grundlegenden Herausforderungen bezüglich der gedanklichen Verfolgung eines Hypertextes gibt es jedoch auch Aspekte der Präsentation, die die Rezeption eines Hypertextes zusätzlich erschweren. Einer dieser Aspekte ist die Tatsache, dass der Link in der Hypertextumgebung des World Wide Webs in der Regel für den Leser äußerst wortkarg ist. Der Link sagt in der Regel meist nur aus, dass es an dieser Stelle die Möglichkeit gibt woandershin zu springen.

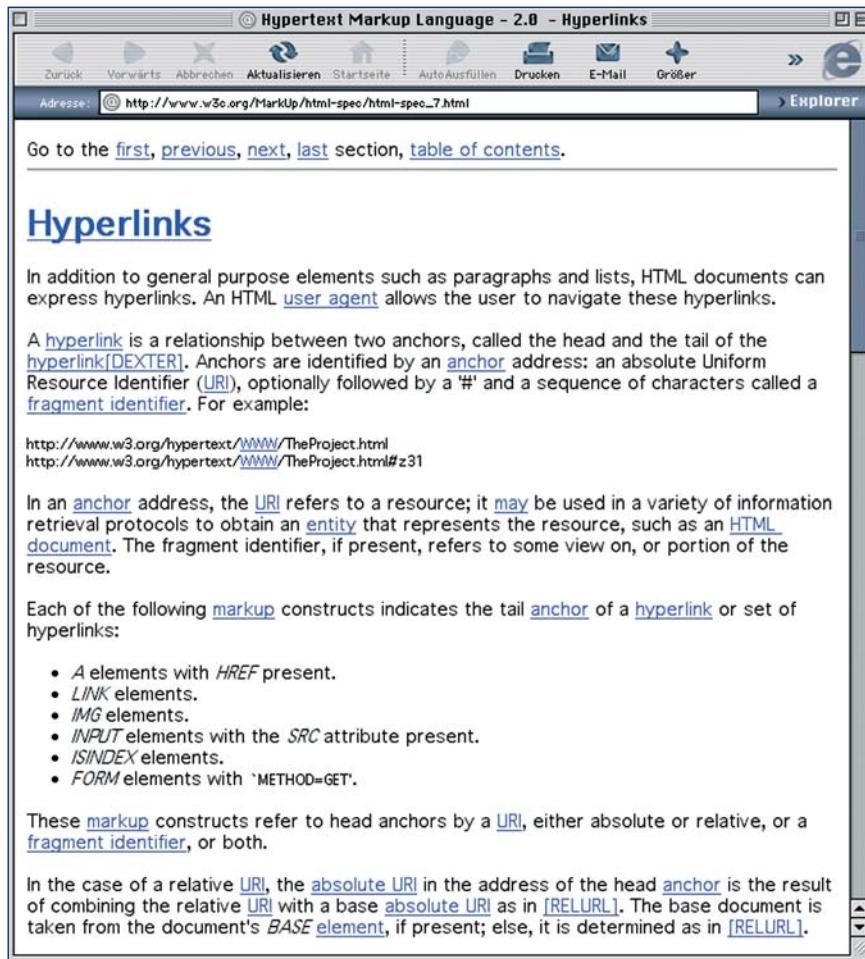


Abbildung 1:  
Screenshot einer Webseite des W3C über  
Hyperlinks.  
(01) Tim Berners-Lee, D. Connolly (1995)

Wohin wohin der Sprung führt, was das Aktivieren des Links bewirkt und was den Leser dort erwartet, welchen Zusammenhang das Ziel mit dem Ursprung des Links hat und von welcher Bedeutung das Ziel für den Leser ist, bleibt in der Regel unbeschrieben.

Der Leser erfährt also in der Regel erst etwas über den Zusammenhang von Quelle und Ziel eines Links, wenn er das Ziel erreicht hat.

Zur Veranschaulichung der Problematik sei hier auf Abbildung 1 verwiesen. Neben der für den Lesefluss störenden, augenfälligen Hervorhebung der Links steht der Leser bei einer derartigen Häufung vor der Frage, ob es tatsächlich Sinn macht dem einzelnen Link zu folgen oder nicht. Da die Darstellung des Links den Leser bei dieser Frage nicht unterstützt, bleiben nur die Möglichkeiten, entweder den Link zu ignorieren oder auf „gut Glück“ dem Link zu folgen, in der Hoffnung, dass die am Ziel befindliche Information zum Verständnis des Quelltextes unbedingt erforderlich ist. Handelt es sich bei dem Zieltext wiederum um einen Hypertext, besteht für den Leser die Gefahr, den Quelltext recht schnell aus den Augen zu verlieren. Der Quelltext kann also die Relevanz für den Leser einbüßen.

Geht man davon aus, dass der Autor des Quelltextes diesen mit der Intention einer Informationsvermittlung erstellt hat, dürfte ein derartiger Relevanzverlust nicht unbedingt im Interesse des Autors sein. Will der Autor jedoch die Beziehung zwischen Zieltext und Quelltext genauer beschreiben, um den Leser bei der relevanten Bewertung eines Links zu unterstützen, stehen ihm mit den Werkzeugen zur Linkerzeugung im WWW zunächst nur rudimentäre Mittel zur Verfügung. (Die Methoden zur Linkerzeugung im WWW werden im folgenden Kapitel genauer vorgestellt.) Ihm bleibt in diesem Fall nur eine textuelle Beschreibung des Links, die je nach Grad ihres Umfangs die Informationsvermittlung jedoch wiederum erschweren kann.

Es bedarf also einer Lösung, die einerseits dem Leser bei der Rezeption eines Hypertextes hilft und die andererseits dem Autor die Möglichkeit gibt, die Rezeption seines Hypertextes in gewisser Weise durch die Beschreibung und Bewertung von Links beeinflussen zu können.

Mit dieser Arbeit soll eine derartige Lösung entworfen werden.

Ziel ist also der Entwurf und die Umsetzung eines grafischen Systems zur Beschreibung von Links im World Wide Web.



# *Link ist nicht gleich Link.*

Im folgenden Kapitel werden einige begriffliche Definitionen getroffen und grundlegende Möglichkeiten der Erzeugung des Links im WWW vorgestellt.



# Hypertext und Link – Begriffsdefinitionen

Laut einer Definition des W3C – dem World Wide Web Consortium – ist Hypertext im Prinzip nichts anderes als Text, der Links zu anderen Texten enthält. Durch diese ist ein Hypertext nicht wie klassischer Lesetext zwingend linear aufgebaut, sondern kann vielfach verknüpft sein. Sind dem Text weitere Elemente wie Grafiken, Filme oder Sound beigelegt, spricht man von Hypermedia.

Beide Begriffe wurden erstmalig von Ted Nelson, einem Visionär auf dem Gebiet der Hypertextforschung gebraucht. Er erweiterte gedanklich die Möglichkeiten der Verknüpfung von Dokumenten und schuf für derartige Konstruktionen in seinem Aufsatz „*A File Structure for the Complex, the Changing and the Indeterminante*“ die Begriffe Hypertext und Hypermedia:

(02) T. H. Nelson (1965)

*„Let me introduce the word “hypertext” to mean a body of written or pictorial material interconnected in such a complex way that it could not conveniently be presented or represented on paper. It may contain summaries, or maps of its contents and their interrelations; it may contain annotations, additions and footnotes from scholars who have examined it.“*

(03) W3C (1992)

Dem Ansinnen des W3C entsprechend, das WWW und die ihm zugrunde liegenden Technologien für jedermann nutzbar und verfügbar zu halten, ist es wichtig zu betonen, dass Hypertext und Hypermedia hierbei als Konzepte verstanden werden und nicht als Produkte. Das heißt, Hypertext und Hypermedia verkörpern zunächst einmal die Idee eines Informationssystems, das von allen Autoren im WWW frei nutzbar ist und nicht bestimmten rechtlichen Restriktionen in seiner prinzipiellen Anwendung unterliegt.

In der Begriffsdefinition des W3C ist der Link eines Hypertextes, eine Beziehung zwischen Ankern (anchors) ein und der selben (internal) oder einer anderen (external) Datensammlung (database). Anker sind das Ziel oder die Quelle eines Links und stellen bestimmte Bereiche des Inhalts eines Knotens (node) dar.

Ein Knoten wiederum beschreibt eine zusammengehörige Informationseinheit; mehrere zusammengehörende Knoten werden in der Datensammlung vereint.

Normalerweise wird durch das Klicken mit der Maus auf einen Anker ein Link aktiviert und der Inhalt des Zielankers wird in einem Darstellungsfenster des Programms zum Lesen von Hypertext, dem Browser, dargestellt. Oftmals ist der Zielanker auch identisch mit dem Inhalt eines gesamten Knotens.

(04) W3C (1992)

Im Sinne eines besseren Verständnisses und einer engeren Begrenzung der Begrifflichkeiten in dieser Arbeit sollen hier jedoch einige spezielle Definitionen vorgeben werden:

**Hypertextseite/Seite:** Eine Hypertextseite ist eine Informationseinheit, die dem Leser als momentaner Inhalt eines Web-Browsers präsentiert wird. Sie kann größer als der Bildschirm sein, um mit einem Mal im Browser angezeigt werden zu können. In diesem Fall besteht für den Leser die Möglichkeit, durch Verschieben der Bildschirmsicht (scrollen) den verborgenen Inhalt zur Anzeige zu bringen. Die Hypertextseite oder ein bestimmter Abschnitt einer Hypertextseite ist das Ziel oder die Quelle eines Hypertextlinks. Eine im Browser darstellbare Informationseinheit, die weder Ziel noch Quelle eines Hypertextlinks ist, soll von dieser Definition einer Hypertextseite ausgeschlossen sein.

Als Ziel eines Hypertextlinks kann die Hypertextseite wiederum Hypertext enthalten, muss dies aber nicht.

**Hypertextdokument/Dokument:** Ein Hypertextdokument ist eine Sammlung mehrerer Hypertextseiten, die inhaltlich zusammengehören. Umfasst eine einzelne Hypertextseite den gesamten Inhalt an Informationen, sind Hypertextseite und Hypertextdokument identisch. Der inhaltliche Zusammenhang eines Hypertextdokumentes sowie seine inhaltliche Abgrenzung zu möglicherweise anderen verlinkten Seiten wird vom Autor des Dokumentes bestimmt.

**Hypertextautor/Autor:** Der Autor eines Hypertextdokumentes ist der Erschaffer der Inhalte des Dokumentes, sowie der Konstrukteur der Verknüpfungsstruktur des Dokumentes. Der Autor ist dadurch die entscheidende Instanz zur Wertung des Inhalts eines Textes durch den Leser. Das heißt, die Zuordnung einer Informationseinheit zu einem Autor ist für den Leser eines der wichtigsten Relevanzkriterien im Sinne einer Wertung des Zusammenhangs von Ziel- und Quellinformation. Der Autor muss nicht eine einzelne Person sein. Wenn mehrere Personen die Verfasser eines inhaltlich zusammengehörigen Dokumentes sind, werden sie im Sinne dieser Definition, gemeinsam als Autor des Dokumentes verstanden.

**Hypertextlink/Link:** Hypertextlinks sind Verweise innerhalb eines Hypertextes auf Hypertextseiten desselben Hypertextdokumentes (interner Link) oder Verweise auf andere Hypertextdokumente oder deren Hypertextseiten (externer Link). Verweise, die lediglich einer einfachen Aneinanderreihung von Informationseinheiten dienen, wie zum Beispiel Hinweise im Sinne von *weiter*, *zurück*, *hoch* oder *mehr*, sollen von dieser Definition eines Hypertextlinks ausgeschlossen werden. Ebenso sollen technische Verbindungen, wie zum Beispiel die Verbindung eines Bildes mit einer Seite (im Sinne einer Einbettung des Bildes in die Seite) oder die Verbindung einer Seite mit einem Style Sheet (einem Dokument, das die Art der Darstellung der Seite beschreibt), von dieser Definition ausgeschlossen werden.

Hypertext ist also Text, der Hypertextlinks enthält. Er wird in Form von Hypertextseiten und Hypertextdokumenten repräsentiert.

Der Einfachheit halber sollen im Verlauf der Arbeit nur noch die Begriffe Seite, Dokument, Autor und Link verwendet werden, jedoch entsprechend ihrer hier genannten Definitionen.



# Erzeugung und Präsentation des Links im WWW

(05) HCIL, University of Maryland

Als 1982 am Human Computer Interaction Lab (HCIL) der University of Maryland ein frühes Hypertextsystem auf IBM Computern entwickelt wurde, war einem der Entwickler, Ben Shneiderman, sicherlich noch nicht bewusst, dass seine Idee, Textelemente selbst als Linkmarker zu benutzen, eines Tages zum quasi Standard der Linkauszeichnung von Millionen Seiten des WWW werden würde.

Die später HyperTIES benannte Hypertextumgebung verwirklichte erstmalig das Konzept der *embedded menus*, der hervorgehobenen Links im Text selbst.

(06) Tim Berners-Lee, CERN (1989/90)

Tim Berners-Lee, der als „Gründer des WWW“ bekannt gewordene Hypertextvisionär, griff die Idee von HyperTIES auf und benutzte den Term *Hot Spots* für derartige Linkauszeichnungen in seinem als grundlegende Arbeit zur Entwicklung des WWW geltenden Aufsatz „*Information Management: A Proposal*“ aus dem Jahr 1989.

(07) Robert Cailliau, W3C (1995)

Zusammen mit der 1960 von Dough Engelbart als Eingabe und Bedienungsgerät seines Hypertextsystems *oNLine System (NLS)* vorgestellten Computermaus waren somit die entscheidenden User-Interface Komponenten des WWW geboren.

(08) D. Raggett, ...

Seit den ersten Tagen des WWW hat sich an der Erzeugung des Textlinks im Prinzip nichts geändert. Schon in der von Berners-Lee auf Grundlage der SGML (Standard Generalized Mark-up Language), einem internationalen Standard zur Beschreibung von strukturellen Elementen eines Textes, entwickelten ersten Version des „Werkzeuges“ zur Erstellung von Dokumenten für das WWW, der HTML (Hypertext Mark-up Language), fanden sich Elemente zur Auszeichnung eines Links, die auch heute noch ihren Bestand in aktuellsten Versionen von HTML haben.

Die Erzeugung eines normalen Hypertextlinks mittels HTML gestaltet sich auch heute noch, wie seit den Anfängen des WWW, recht einfach.

Diese Einfachheit der Erzeugung eines Hypertextes war im übrigen auch eine der entscheidenden Stärken der von Berners-Lee entwickelten HTML und einer der Gründe für die rasante Verbreitung und Entwicklung der Hypertextumgebung des WWW.

Der Hypertextlink in HTML-basierten Hypertexten des WWW kann nach wie vor durch einen einzigen Befehl erzeugt werden. Dieser Befehl, im Umgang mit HTML wird dabei in der Regel statt Befehl von so genannten *tags* gesprochen, ist der *anchor-tag*. Er dient zunächst der Auszeichnung der Quelle eines Verweises und der Definition des Ziels eines Verweises im Text. Derartig ausgezeichnete Textelemente werden beim Umgang mit der HTML als Anker bezeichnet. *Tags* wie der *anchor-tag* unterteilen sich dabei in einen so genannten *start-tag* und einen *end-tag*.

Dies sieht dann auf der Ebene der Erzeugung eines Ankers wie folgt aus:

```
<a> auszuzeichnendes Textelement </a>
```

Damit ein Anker als Hypertextlink funktioniert, müssen dem *anchor-tag* noch bestimmte Attribute und Werte mitgegeben werden. Das wichtigste Attribut ist dabei das *hypertext-reference-Attribut*. Ihm zugeordnet ist ein Wert, der meistens aus einer URL (uniform resource locator) besteht. Die URL kann beispielsweise aus der Adresse der zu verlinkenden Seite innerhalb des WWW bestehen. Damit stellt sich der *anchor-tag* im Sinne einer Linkerzeugung folgendermaßen dar:

```
<a href="Adresse"> Link zur Adresse</a>
```

Im Browser wird ein derartiger Link dann in der Regel, wenn dies vom Gestalter der Webseite nicht anders festgelegt wurde, in Form des unterstrichenen und blau hervorgehobenen Textelementes wiedergegeben (siehe auch Abbildung 1).

Bei der Verlinkung von Textelementen unterscheidet man absolute Links und relative Links. Absolute Links werden mit der vollständigen URL einer Seite angegeben (z.B. <http://www.dokumentname.com>). Relative Links geben nur einen Teil der URL einer Seite an und dienen der Verlinkung innerhalb der Struktur eines Dokumentes, vorausgesetzt alle zu verlinkenden Teile dieses Dokumentes befinden sich innerhalb einer Ordnerstruktur auf einem Server.

Liegen die Seite, von der der Link ausgeht, und die Zielseite innerhalb des gleichen Ordners auf einem Server, genügt beispielsweise die Angabe der Adresse in folgender Form:

```
<a href="andere_seite.html">Link zur anderen Seite</a>
```

Auch eine Verlinkung innerhalb einer Seite eines Dokumentes ist mit HTML möglich. Dazu muss mittels des *anchor-tags*, im Gegensatz zur oben beschriebenen Verlinkung von Seiten, jedoch auch das Ziel innerhalb des Textes auszeichnet werden. Hierzu muss das Ziel-Textelement benannt werden:

```
<a name="ziel">Ziel-Textelement</a>
```

Der Hypertextlink zu diesem Ziel-Textelement wird dann an anderer Stelle der Seite wie folgt angegeben:

```
<a href="#ziel">Link zum Ziel-Textelement</a>
```

Das Ziel-Textelement wird im Browser nicht besonders hervorgehoben, es bleibt unsichtbar. Es springt nach Betätigung des Links an die oberste Position im Darstellungsfenster des Browsers, vorausgesetzt, der Seiteninhalt nach dem Zielelement ist wenigstens so umfangreich, dass er das Darstellungsfenster des Browsers mit einem Mal ausfüllen kann.

Eine Kombination der Verlinkung mittels einer Adresse und die Verlinkung zu einem benannten Zielanker innerhalb eines Textes ist ebenfalls möglich. Damit besteht die Möglichkeit, von einer Quellseite direkt zu einem Abschnitt einer anderen Zielseite zu springen.

(09) M. E. Holzschlag (1999)

Der Linkbegriff in HTML ist weiter gefasst, als der in der oben stehenden Definition festgelegte Begriff des Hyperlinks. Als Links werden in HTML beispielsweise auch die Arten der Einbettung eines Bildes in eine Seite bezeichnet oder die Belegung eines Textelementes mit der Verknüpfung einer E-Mail-Adresse. Derartige Verknüpfungen sollen jedoch im Rahmen dieser Arbeit keine Rolle spielen.

Eine entscheidende Eigenschaft von Links, die mittels HTML erzeugt worden sind, ist die Tatsache, dass es nicht möglich ist, Links überlappend zu setzen. Diese Möglichkeit gibt es momentan nur in Hypertextdokumenten, die mittels XML (eXtensible Mark-up Language) beschrieben werden und die durch den im Sommer 2001 vom W3C verabschiedeten Standard XLink (XML Linking Language) miteinander verlinkt sind. Auf Grund der Tatsache, dass HTML nach wie vor die am meisten verbreitete Sprache zur Beschreibung von Hypertextdokumenten im WWW ist, soll hier lediglich diese neue Möglichkeit der Verlinkung von Dokumenten genannt werden. Auch im Hinblick auf die Tatsache, dass die in dieser Arbeit vorgestellte Variante der beschreibenden Auszeichnung von Links eine derartige Überlappung von Links ebenfalls zulässt.

(10) S. DeRose, ...

(11) S. DeRose, ...

# *Nach dem Klick ist vor dem Klick.*

Dieses Kapitel befasst sich mit der Problematik der Beschreibung von Links. Dazu werden zunächst einige bestehende Ansätze zu diesem Thema vorgestellt. Anschließend wird das anlässlich dieser Arbeit entworfene Beschreibungssystem erläutert.



## Bestehende Ansätze

(12) M. Thüring...

Um die Lesbarkeit eines Hypertextes zu erhöhen, kann der Autor eines Textes dem Leser bei der Konstruktion des Gedankenmodells helfen, indem er Faktoren, die diesen Prozess unterstützen, verstärkt und Informationen, die diesen Prozess behindern, abschwächt. Ein wesentlicher Aspekt der Erhöhung des Verständnisses der Zusammenhänge eines Hypertextes ist die Beschreibung der Eigenschaften der Verbindungen zwischen den Teilen des Hypertextes.

Um diese Forderungen an einen nutzerfreundlichen Hypertext zu erreichen, steht der Autor vor genau zwei Problemen. Dies ist zum ersten die Frage der exakten, verständlichen und eindeutigen Beschreibung der Links und zweitens die Frage der Einbettung dieser Bedeutungsbeschreibung in seinen Text. Hier soll es zunächst erst einmal um die Frage der Beschreibung der Links gehen.

Auf Möglichkeiten zur Lösung der zweiten Frage wird im folgenden Kapitel eingegangen.

Das Problem bei der Beschreibung eines Links liegt zunächst an den schier unendlich denkbaren Möglichkeiten von Linkeigenschaften. Hinzu kommt die Tatsache, dass es für eine derartige Einteilung von Links in bestimmte Eigenschaften und Eigenarten keinen verbindlichen Standard gibt, der dem Autor eines Hypertextes bei der Bewältigung seines Problems hilft. Ein derartiger Standard wäre für den Autor eines Hypertextes bei der Konstruktion seiner Verlinkungen nützlich und könnte dem Leser beim Verständnis des Aufbaus und der Struktur des Hypertextes und somit beim Verständnis des Inhalts des Hypertextes an sich behilflich sein.

Will der Autor ein derartiges System zur Beschreibung von Links in seinen Hypertext auf Grund des fehlenden Standards selbst integrieren, stellt ihn das vor eine komplexe Aufgabe. Diese erfordert von ihm neben der Erstellung der Inhalte des Hypertextes, noch zusätzliche Aufmerksamkeit und einen großen Zeitaufwand. Hinzu kommt, dass ein derartiges, vom Autor eines Hypertextes selbst erstelltes Sys-

tem, zunächst erst einmal nur Gültigkeit innerhalb seines Hypertextes hätte. Der Leser wiederum wäre bei der Rezeption eines einzigen Hypertextes sicherlich mit einem schlüssigen System der Linkbeschreibungen konfrontiert, würde sich aber bei der Rezeption unterschiedlicher Hypertexte von verschiedenen Autoren einer Vielzahl von Systemen, Bedeutungen und Begriffen gegenübergestellt sehen. Dies würde ihn überfordern und wäre für das Verständniss und eine Interpretation des Hypertextes im Sinne des Autors nicht förderlich. Bei der Frage nach einem System zur Beschreibung von Links handelt es sich um ein grundlegendes Problem des Mediums an sich. Ansätze zur Lösung dieses Problems auf übergeordneter Ebene können nur in generalisierter Form realisiert werden.

Um also den Autor eines Hypertextes nicht selbst mit der Problematik der Linkbeschreibung und -klassifizierung zu konfrontieren, haben sich schon zahlreiche Wissenschaftler auf dem Gebiet der Hypertextforschung dieses Problems in seiner grundlegenden Form angenommen. (Neben den im folgenden vorgestellten Beispielen befinden sich auf der CD zu dieser Arbeit weitere Quellen zu diesem Thema.) Viele dieser Arbeiten sehen ihre Bedeutung in der Entwicklung einer Hilfestellung für Autoren eines Hypertextes und in der Erhöhung des Verständnisses eines derartigen Textes für den Leser. Derartige Ansätze decken sich somit auch mit dem Grundgedanken dieser Arbeit. Es gibt jedoch auch zahlreiche Arbeiten, die eher im Sinne einer wissenschaftlichen Text- oder Datenanalyse zu werten sind und daher im Prinzip mit dem Anliegen der Verbesserung des Links als Autor-Leser-Schnittstelle, auch auf Grund ihrer Komplexität und Spezifizierung, nicht allzu viel gemein haben. Unter dem Gesichtspunkt der Verwertung innerhalb einer graphischen Benutzeroberfläche sind diese meist nicht dienlich.

So unterschiedlich wie die Ansätze der Untersuchungen zu diesem Thema sind, so unterschiedlich sind auch die Ergebnisse der jeweiligen Verfasser. Exemplarisch sollen daher im folgenden einige Theorien vorgestellt werden, um einen Ausschnitt aus der Bandbreite der Lösungsansätze im Umgang mit der Beschreibung von Links aufzuzeigen.

So beschreibt Tim Berners-Lee beispielsweise Linkeigenschaften die an die mathematische Mengenlehre erinnern: *A is part of B / B includes A; A Made B / B is made*

*by A; A uses B / B is used by A; A refers to B / B is referred by A.*

Im selben Text über „Link Types“ beschreibt er auch noch weitere sogenannte „magische“ Linktypen wie: *UseIndex, UseGlossary, Annotation, Next – Previous – Up, Embedded information, Person described by node A is author of node B, Person described by node A is interested in node B, Node A is in fact a previous version of node B, Node A is in fact a set of differences between B and its previous version.*

(13) Tim Berners-Lee (1990)

Der Standard des W3C zur HTML 4.01 aus dem Jahr 1999 enthält eine ebensolche Mischung aus semantischen, technischen und strukturellen Link-Typen zur Beschreibung eines Links: *Alternate, Stylesheet, Start, Next, Prev, Contents, Index, Glossary, Copyright, Chapter, Section, Subsection, Appendix, Help, Bookmark.*

(14) D. Raggett...

Die Vermischung von semantischen, technischen und strukturellen Eigenschaften in einem gleichwertigen Pool von Kategorien erschwert jedoch bei beiden Beispielen das Verständnis für die Bedeutungen, als auch ihre Verwendung. Hinzu kommt, dass die in HTML vorgesehenen *data types* zur Linkbeschreibung (*link types*) einerseits dem Leser nicht sichtbar gemacht werden und in ihrer Eigenschaft als Werte andererseits nicht kombinierbar sind. Dies sind sicher auch die Gründe, weshalb die *link types* in HTML-Dokumenten nur spärlich oder meist gar nicht zum Einsatz kommen.

Einen anderen Ansatz zur Beschreibung von Links in wissenschaftlichen Arbeiten formuliert Randall Trigg in seiner in diesem Zusammenhang oft zitierten Arbeit „*A Network-Based Approach to Text Handling for the Online Scientific Community*“ aus dem Jahr 1983. Er unterscheidet im Kontext wissenschaftlicher Arbeiten in so genannte *normal link types* und *commentary link types* und legt über 80 verschiedene Typen fest (siehe Abbildung 2).

(15) Randall Trigg (1983)

Triggs Ausführungen sind sehr umfassend und erlauben es, einen Link unter semantischen Gesichtspunkten recht exakt zu beschreiben. Mit seinen Festlegungen lassen sich sicherlich für fast jeden Link einer wissenschaftlichen Arbeit Eigenschaften zuordnen. Allerdings sind einige Eigenschaften nicht in jedem Punkt eindeutig. So liegt es im Ermessen des Autors, einen Link sowohl als Beispiel, Erläuterung oder auch Anwendung zu definieren. Der Erkenntnisgewinn für den Leser über den Inhalt des Ziels würde also deswegen zunächst der gleiche sein, als wäre für eine derartige

Normal link types			
Citation C-source C-pioneer C-credit C-leads C-epon  Background Future	Refutation Support  Methodology Data  Generalization/Specification Abstraction/Example Formalization/Application	Argument A-deduction A-induction A-analogy A-intuition  Solution  Summarization/Detail Alternate-view Rewrite	Simplification/Complication Explanation  Correction Update  Continuation
Commentary link types			
Comment Critics Supportive  <i>Environment</i> E-comment E-misrepresent E-vacuum E-ignored E-isupersede E-irefute E-isupport E-repeate	<i>Problem Posing</i> P-comment P-trivial P-unimportant P-impossible P-ill-posed P-solved P-ambitious  <i>Points</i> Pt-comment Pt-trivial Pt-unimportant Pt-irrelevant Pt-redherring Pt-contradict Pt-dubious	Pt-counter Pt-inlegant Pt-simplistic Pt-arbitrary Pt-unmotivated  <i>Arguments</i> A-comment A-invalid A-insuff A-immaterial A-mislead A-alternate A-strawman	<i>Data</i> D-comment D-inadequate D-dubious D-ignored D-irrelevant D-inapplicable D-misinterpreted  <i>Style</i> S-comment S-boring S-unimaginative S-incoherent S-arrogant S-rambling S-awkward

Abbildung 2:

Tabelle mit den von Randall Trigg vorgestellten Linktypen.

Seite nur ein beschreibender Begriff festgelegt. Diese Möglichkeit der Mehrdeutigkeit widerspricht in gewisser Weise auch dem Anliegen Triggs, Links möglichst exakt zu beschreiben und eigenmächtige Linkbedeutungen, die durch den Autor selbst festgelegt werden, im Sinne einer besseren Orientierung für den Benutzer auszuschließen.

Einen wiederum anderen Ansatz zur Beschreibung der Struktur von Links und der Konstruktion von Hypertexten verfolgt Marc Bernstein in seinem Aufsatz „*Patterns of Hypertext*“. Er untersucht Hypertexte auf Muster innerhalb ihrer Struktur und benennt insgesamt zehn Eigenschaften, nach denen Links bzw. die Verknüpfungen von Dokumenten aufgebaut sind: *Cycle - Joyce's Cycle - Douglas's Cycle - Web Ring - Contour, Counterpoint, MirrorWorld, Tangle, Sieve, Montage, Neighborhood, Split/Join, Missing Link, Navigational Feint*.

Einige der genannten Eigenschaften zeigen Bernsteins Nähe zur Hypertextliteratur und sind wohl auch eher prosaischer Bedeutung. Interessant an Bernsteins Aufsatz sind jedoch die Illustrationen zu einigen der benannten Eigenschaften, stellen sie

(16) Mark Bernstein (1998)

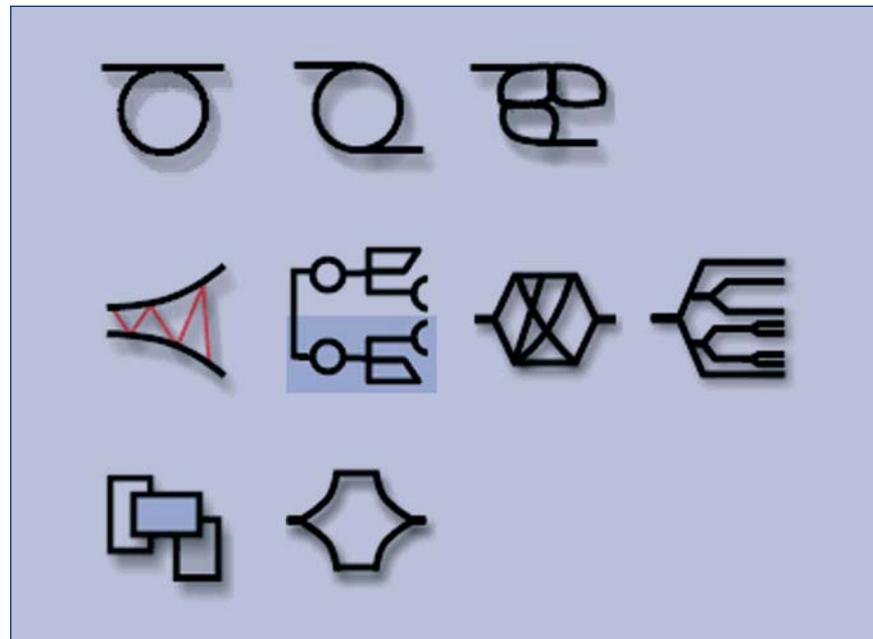


Abbildung 3:

Illustrationen aus Marc Bernsteins

Aufsatz „Patters of Hypertext“

oben: Cycle, Joyce's Cycle, Contour

mitte: Counterpoint, MirrorWorld,

Tangle, Sieve

unten: Montage, Split/Join

(von links nach rechts)

doch den Versuch dar, den abstrakten Sachverhalt der Verknüpfungsmechanismen von Hypertext grafisch zu beschreiben. Sie waren daher auch einer der Gedankenanstöße zu dieser Arbeit.

Bernsteins Arbeit zur Linkbeschreibung als Teil einer Hypertextkonstruktion unterscheidet sich grundlegend von Triggs Abhandlungen zu diesem Thema. Beide verwenden andere Kategorien und definieren die konkreten Eigenschaften des Links innerhalb dieser Kategorien. Sie sind innerhalb dieser Kategorien in sich schlüssig. Durch die Verwendung unterschiedlicher Kategorien sind die Arbeiten im Prinzip auch nicht vergleichbar. In einem Aspekt sind beide Arbeiten jedoch gleich: Sie beschreiben die Eigenschaften eines Links nicht umfassend.

Die Bedeutung einer durch einen Link definierten Verknüpfung von Textelementen besteht vor allem für den Leser aus einer Kombination von Eigenschaften technischer, semantischer und struktureller Kategorien. Ein wirklich generell gültiges System zur Linkbeschreibung sollte also alle derartigen Kategorien berücksichtigen, dabei jedoch immer noch verständlich und übersichtlich bleiben.

Die Frage nach dem „*Wohin?*“ eines Links sollte kurz, eindeutig und entsprechend präzise beantwortet werden, schließlich will der Rezipient eines Hypertextes diesen inhaltlich erfassen und nicht neben dem eigentlichen Text noch einen zweiten über die Bedeutung der eingesetzten Links lesen.

# Das Beschreibungssystem

Da in dieser Arbeit ein Zeichensystem zur Beschreibung von Links entworfen werden soll, ist es unumgänglich, gleichzeitig ein entsprechendes System zur Kategorisierung und Festlegung der Eigenschaften eines Links zu entwerfen, das die Basis für die gestalterische Umsetzung darstellt. Wie im voraus gegangenen Abschnitt erläutert, gibt es zu diesem Zweck bisher keinen verbindlichen Standard, auf den der Autor eines Hypertextes dabei zurückgreifen könnte. Auch sind andere zu dieser Thematik entwickelte Systeme zur Linkbeschreibung im Hinblick auf das zu entwerfende Zeichensystem wenig dienlich. Ihnen fehlt in der Regel eine gewünschte grafisch umsetzbare Systematik und Komplexität.

Die Eigenschaften eines Links können vielfältig sein. Um diese Vielfältigkeit innerhalb eines Beschreibungssystems durch Festlegungen einzuschränken, bedarf es einer Kombination aus bestimmten Kategorien und bestimmten Eigenschaften, die den Kategorien untergeordnet sind. Man kann sich ein derartiges System auch als eine Sammlung bestimmter Attribute und zugeordneter Werte vorstellen. Die Attribute können technischer, semantischer oder struktureller Natur sein. In einem derartigen System sollte es also möglich sein, den Link mittels einer Kombination von Werten aus verschiedenen Attributen zu beschreiben. Dabei sollte jedoch die zugrunde liegende Zuordnung von Attribut und Wert immer eindeutig bleiben. Mit anderen Worten, ordnet man einem Link bestimmte Werte zu, müssen diese eindeutig sein, sowohl im Sinne der Abgrenzung der Werte eines Attributes untereinander, als auch der Abgrenzung von Werten unterschiedlicher Attribute zueinander.

Um ein derartiges System sowohl für den Autor benutzbar zu halten, als auch in seiner Darstellung für den Leser schnell erfassbar und verständlich zu sein, sollte der Umfang des Systems nicht zu komplex sein. Der Aufbau des Systems mit seiner ihm zugrunde liegenden Ordnung von Attributen und Werten sollte sowohl dem Autor als auch dem Leser eindeutig, verständlich und einprägsam sein.

Da das System zur Beschreibung von Links letztendlich in Form von Zeichen in Verbindung mit einem Link präsentiert werden soll, muss auch die Korrespondenz zwischen Zeichen und System ebenso eindeutig, verständlich und leicht erlernbar sein.

An dieser Stelle sei nochmals erwähnt, dass das folgende System als Beschreibungssystem für Hypertextlinks vorrangig in wissenschaftlich-technischen Hypertexten gedacht ist. Es soll als ein unterstützendes und hilfreiches Mittel sowohl für den Autor als auch für den Leser derartiger komplexer Hypertexte gedacht sein. Verweise, die der Navigation innerhalb des Dokumentes dienen, oder aber auch Verweise, die eine bloße Aneinanderreihung von Informationseinheiten beschreiben, werden entsprechend der Definition eines Hypertextlinks bei diesem System nicht berücksichtigt.

Die hier nunmehr festgelegten Attribute nach denen ein Link beschrieben werden soll sind: **Ort**, **Inhalt** und **Art**.

Unter **Ort** versteht man die Eingrenzung der Lage eines Zielpunktes des Links bezüglich des Quelldokumentes. Das *Attribut des Ortes* soll jedoch nicht physikalisch die Lage eines Dokumentes auf einem Server bezeichnen, sondern die strukturelle Lage des Ziels innerhalb oder außerhalb des Dokumentes eines Autors (siehe dazu auch Relevanzkriterium in der Definition des Hypertextautors). Eine physikalische Definition der Lage macht als Information für den Leser keinen Sinn. Physikalisch betrachtet können die Teile eines Dokumentes auch auf verschiedenen Servern oder unter verschiedenen Adressierungen gespeichert sein, was den Leser jedoch im Prinzip nur beiläufig interessiert. Außerdem lässt sich die physikalische Lage eines Dokumentes, zumindest was die Adressierung anbelangt, aus der URL der Links ablesen.

Das *Attribut des Ortes* soll dem Leser anzeigen, ob er das aktuelle Dokument verlässt oder sich innerhalb des Dokumentes bewegt. Der Autor kann damit bestimmte Dimensionen seines Dokumentes beschreiben und sein Dokument von anderen Dokumenten inhaltlich abgrenzen.

Es sollen folgende Werte zum *Attribut des Ortes* festgelegt werden:

*Innerer Anker*

*Interne Seite*

*Externes Dokument*

Der Wert des *internen Ankers* verweist auf einen Sprung zu einem Abschnitt innerhalb der aktuell angezeigten Seite. Der Leser erkennt daran, dass er die Seite nicht verlässt und sich nur zwischen Text-Abschnitten ein und derselben Seite bewegt.

Der Wert der *internen Seite* verweist auf einen Sprung zu einer neuen Darstellungsseite desselben Dokumentes. Dies könnten zum Beispiel eine Kapitelseite, eine Glossarseite, eine einzelne Darstellungsseite oder die Erläuterungen zu einem Beispiel auf einer anderen Seite des Dokumentes sein. Die neue Seite kann entweder die alte Seite im Darstellungsfenster verdrängen oder es wird ein neues Fenster mit dem Inhalt der neuen Seite angezeigt. Auch der Sprung zu einem Abschnitt einer anderen Seite desselben Dokumentes wird mit dem Wert der *internen Seite* angezeigt.

Der Wert des *externen Dokumentes* verweist auf den Sprung zu einem anderen Dokument als dem gerade betrachteten. Es kann durchaus vom selben Autor verfasst sein, gehört aber nicht zum inhaltlichen Zusammenhang des Ausgangsdokumentes. Der Wert des *externen Dokumentes* schließt den Sprung zur Startseite, zu einzelnen Seiten des Dokumentes und Abschnitten in einzelnen Seiten ein.

Bei der Betrachtung der Werte zum *Attribut des Ortes* fällt auf, dass sie unterschiedliche Schärfegrade der Beschreibung beinhalten. Der Wert *innerer Anker* ist sehr eindeutig und klar definiert. Der Wert des *externen Dokumentes* ist da wesentlich unpräziser. Diese zunehmende Unschärfe macht durchaus Sinn, wenn man den Zusammenhang des Inhalts der Quelle und des Ziels als inhaltlich strukturelle Entfernung ansieht. Mit anderen Worten: Je unschärfer der Wert desto weiter ist in der Regel die inhaltliche Entfernung vom gerade gelesenen Text.

Das zweite Attribut, das den Inhalt in seinen semantischen Eigenschaften beschreiben soll, ist das *Attribut des Inhalts*. Eine Inhaltsbeschreibung eines Textes kann

sehr komplex sein, wie bereits die im vorherigen Abschnitt erwähnte Arbeit von Randall Trigg zeigt. Im Sinne einer Bewertung der inhaltlichen Zusammenhänge von Quelle und Ziel durch den Leser und einer Abgrenzung dieser durch den Autor lassen sich jedoch drei Hauptwerte finden. Diese sind zwar bei weitem nicht so präzise wie die von Trigg vorgestellten, aber als Information für den Leser sind sie essentiell und am wichtigsten.

Es sollen daher für das *Attribut des Inhalts* folgende Werte festgelegt werden:

*Ausführung*

*Bestätigung*

*Assoziation*

Der Wert *Ausführung* definiert Seiten, die normale, inhaltlich weiterführende Bestandteile des Textes sind. Dies können zum Beispiel ein weiteres Kapitel, Anmerkungen, Beispiele, Darstellungen und Tabellen sein, die der inhaltlichen Zusammengehörigkeit des Dokumentes entsprechen. Will der Leser sich die inhaltliche Bedeutung und Aussage eines Dokumentes erarbeiten, gehören dazu alle Dokumentteile, die dem Wert *Ausführung* entsprechen. Der Autor markiert mit dem Wert *Ausführung* quasi seine eigene gedankliche Arbeit am Dokument.

Der Wert *Ausführung* gilt in der Regel nur für Seiten, die sich innerhalb eines Dokumentes befinden.

Der Wert *Bestätigung* kann auch im Sinne einer Referenz verstanden werden. Der Leser erfährt dadurch, dass ein entsprechender Link ihn zu einem Text führt, der den gerade angezeigten Text inhaltlich bestätigt, die inhaltliche Quelle des Textes markiert oder repräsentiert. Mit anderen Worten, ein Link mit dem Wert *Bestätigung* kann den Leser beispielsweise zu einer Literaturliste mit Quellenangaben führen oder aber auch zur Quelle eines Zitates selbst. Der Wert *Bestätigung* kann somit sowohl innerhalb, als auch außerhalb eines Dokumentes liegen. Im Gegensatz zum Wert *Ausführung* sind Texte mit dem Wert *Bestätigung* zur inhaltlichen Erfassung eines Dokumentes nicht unbedingt erforderlich.

Der Wert *Assoziation* markiert die weitestgehende inhaltliche Entfernung vom Quelltext. Er markiert Seiten, die es dem Leser ermöglichen sollen, aus dem vorlie-

genden Text heraus weiterführenden oder völlig freien Gedankenverknüpfungen zu folgen. Ein Link mit dem Wert *Assoziation* könnte also beispielsweise innerhalb einer Arbeit über Hypertext der Verweis auf eine Ergebnisseite einer Suchmaschine sein, die Links zum Suchbegriff „Hypertext“ enthält. Es könnte auch ein unkommentierter Link zu einer Portalseite über Hypertext sein, sofern diese nicht Quelle der Arbeit ist und somit eher dem Wert *Bestätigung* entspricht. Der Wert *Assoziation* führt in der Regel zu Seiten, die außerhalb eines Dokumentes liegen, muss dies aber nicht.

Auch zwischen den Werten zum *Attribut des Inhalts* besteht eine zunehmende Unschärfe, im Sinne einer inhaltlichen Relevanz bezüglich des Ausgangstextes oder Dokumentes. Die Unschärfe nimmt vom Wert *Ausführung*, über den Wert *Bestätigung* bis hin zum Wert *Assoziation* zu.

Das dritte Attribut ist technischer Natur und beschreibt die Art der Präsentation und Kodierung der Information, die sich wiederum auf Art und Aufwand der Rezeption der Information durch den Leser auswirkt. Es beinhaltet Werte, die dem Leser darüber Aufschluss geben, wie und mit welchem Aufwand er sich die verlinkten Informationseinheiten erschließen muss. Dieses Attribut ist daher wichtig für den Leser im Sinne der Planbarkeit seiner Arbeit mit einem Hypertext bzw. dessen Verlinkungen.

Für das *Attribut der Art* sollen folgende Werte festgelegt werden:

- Linear*
- Verbunden*
- Applikativ*

Der Wert *Linear* steht für die lineare Präsentation einer Informationseinheit. Das bedeutet für die Präsentation, dass die Information in der Regel in einem Fenster des Browsers dargestellt wird. Dies könnten zum Beispiel ein Text, ein einzelnes Bild oder eine Tabelle sein. Auch eine Hypertextseite oder der Abschnitt einer Hypertextseite sind lineare Informationen, wenn sie in sich eine Informationseinheit darstellen. Ebenso linear können Filme und Soundfiles sein.

Der Wert *Verbunden* steht für Informationen, die sich nicht mit einem Mal erschließen lassen, sondern nur durch die Gesamtrezeption verbundener Informationseinheiten. Beispielsweise ist ein Link zur Startseite eines Hypertextdokumentes, das als Summe die zu verknüpfende Gesamtinformation darstellt, eine Verlinkung mit dem Wert *Verbunden*. Auch ein Link zu einer Linkliste ist in der Regel eine derartige Verknüpfung, da die Liste an sich ja nur den Zugang zu weiteren Informationen repräsentiert und in sich noch nicht die Information selbst darstellt. Denkbar wäre hinter diesem Wert auch ein interaktiver Film.

Der Wert *Applikativ* fasst Informationen, die dem Wert *Verbunden* entsprechen, noch weiter. Hinter einem derartigen Link können sich komplexe mediale Systeme, Applikationen oder Programme verbergen. Beispielsweise könnten dies interaktive Flash oder Director Applikationen sein (Applikationen, die mittels der Programme Director oder Flash der Firma Macromedia erstellt wurden) oder ein interaktives Lernsystem.

Der Unschärfeaspekt, der bei den beiden vorher genannten Attributen auftritt, entspricht beim *Attribut der Art* eher dem zunehmenden Wert des Aufwandes der Informationserfassung und Erarbeitung, sowie dem Grad der Kodierung der Information.

Für alle hier aufgeführten Attribute gilt gleichermaßen: Aus den Werten eines Attributes darf immer nur ein Wert oder gar kein Wert verwendet werden. Soll ein Link umfassend beschrieben werden, sollten Werte aus allen vorhandenen Attributen gewählt werden. Lässt sich ein Link innerhalb eines Attributes nicht durch die gegebenen Werte beschreiben, wird innerhalb dieses Attributes kein Wert vergeben.

Es dürfen keine weiteren Attribute hinzugefügt werden. Gleches gilt auch für die Werte innerhalb der gegebenen Attribute.

Die Werte zwischen den Attributen sind alle miteinander kombinierbar.

Die Anwendung der Attribute und Werte durch den Autor gestaltet sich nach diesen Regeln einfach und übersichtlich.

Angenommen der Autor eines technischen Produkthandbuchs möchte mit einem Link auf eine interaktive Lernumgebung verweisen, die den Leser befähigen soll, einen bestimmten Aspekt der Produktbedienung zu erlernen. Die Lernumgebung ist Teil des gesamten Handbuchs, also setzt er zunächst den Wert *Interne Seite* des *Attributes des Ortes*. Da die Erarbeitung der Inhalte der Lernumgebung unbedingter Bestandteil des Wissens über das Produkt ist, legt er als Zweites den Wert *Ausführung des Attributes des Inhalts* fest. Eine Lernumgebung ist eine komplexe Applikation, daher wird als Drittes der Wert *Applikativ* des *Attributes der Art* gesetzt.

Der Leser wiederum, der sich mit einem derartig beschriebenen Link konfrontiert sieht, erkennt anhand der gesetzten Werte:

- Der Link bewirkt das Anzeigen einer neuen Seite, dies bedeutet unter Umständen eine gewisse Ladezeit. Der Link führt zu einem Bestandteil des Produkthandbuchs.
- Der Link führt zu einem für das Wissen über das Produkt notwendigen Teil des Handbuchs, es ist daher sinnvoll dem Link zu folgen.
- Der Link führt zu einer Applikation, das bedeutet, wenn er dem Link folgt, ist dieser Schritt größer, als bei einem Link mit dem Wert *Linear*. Eine Applikation verlangt von ihm einen wesentlich größeren Aufwand hinsichtlich der Erarbeitung des Inhalts. Es ist daher möglicherweise sinnvoll zunächst den Inhalt der Seite, die den Link enthält, komplett zu lesen und erst dann dem Link zu folgen.

Ein weiteres Beispiel für den Umgang mit den Attributen und Werten könnte folgende Art eines Links sein:

Der Autor des gleichen Produkthandbuchs möchte mit einem Link auf eine Seite verweisen, auf der ein Anwender des Produktes einen Test des Produktes vorstellt. Der Autor setzt zu diesem Zweck die Werte *Externes Dokument*, *Bestätigung* und *Linear*. Der Leser entscheidet sich an Hand der gesetzten Werte, dem Link zunächst nicht zu folgen, denn er möchte sich erst die Inhalte des Produkthandbuchs komplett erarbeiten, um dann in einem zweiten Schritt erst nach weiterführenden Informationen zu suchen. Möglicherweise entscheidet er sich aber auch, dem Link zu folgen, da der Wert *Linear* ihm angezeigt, dass es sich bei der verlinkten Infor-

mationseinheit voraussichtlich nur um eine einzelne Seite handelt und er mit einem Schritt sofort wieder zum Ausgangsdokument zurückkommen kann.

Ein drittes Beispiel sei hier noch aufgeführt:

Der Autor möchte mittels eines Links auf andere Produkte seines Portfolios verweisen. Das Produkt wird in einem eigenen Hypertextdokument beschrieben.

Er legt die Werte *Externes Dokument* und *Verbunden* fest. Für das *Attribut des Inhalts* setzt er keinen Wert, weil eine derartige Seite weder zu seinen Ausführungen zählt, noch eine Bestätigung darstellt. Am ehesten würde noch der Wert *Assoziation* passen, wenn es sich zum Beispiel um Zusatzteile oder Produkterweiterungen handelt. Dies soll aber nicht so sein und er legt daher keinen Wert für das Attribut fest. Die Informationen über den Link wären somit zwar nicht so komplex wie im Falle des Setzens von drei Werten, aber der Leser erfährt immer noch genug über den Link, um ihn in seiner Relevanz beurteilen zu können.

Ein letztes Attribut eines Links soll zum Schluss noch festgelegt werden. Es ist das *Attribut der Wichtigkeit* und enthält nur den einen Wert *Wichtig*. Es soll dem Autor dazu dienen, Links, die er zur Vermittlung des Inhalts für essenziell hält, auch dementsprechend auszuzeichnen. Dem Leser hilft dieses Attribut, neben den anderen genannten, besonders bei der Bewertung einer Verknüpfung. In oben genanntem ersten Beispiel würde der Wert Wichtig am Link zur interaktiven Lernumgebung dem Leser anzeigen, dass der Autor es, neben der für den Leser erkennbaren Relevanz des Links, für unabdingbar hält, dem Link zu folgen.

# Die Beschreibungsmatrix

Wie dem Leser beim voraus gegangenen Abschnitt sicherlich aufgefallen ist, handelt es sich bei dem festgelegten System zur Linkbeschreibung um ein System aus drei Attributen, denen jeweils drei Werte zugeordnet wurden. Das *Attribut der Wichtigkeit* sei dabei zunächst erst einmal nicht berücksichtigt. Diese Anzahl hat neben der Frage einer überschaubaren Komplexität des Systems noch den Vorteil, dass die Attribute und deren Werte in Form einer Matrix zusammengefasst werden können.

Eine Matrix hat den Vorteil, alle Werte übersichtlich in sich zu vereinen und eine verständliche und einprägsame Form der Bündelung von Information darzustellen. Sie ist die Manifestation des Systems und all seiner Einzelfaktoren. Aus der Matrix lassen sich alle möglichen Verknüpfungen der Werte der festgelegten Attribute erzeugen.

Drei Spalten und drei Zeilen bilden die Form der *Beschreibungsmatrix*. Die Spalten beschreiben die Attribute, die Zeilen wiederum nehmen die Werte der einzelnen Attribute auf. Die erste Spalte steht für das *Attribut des Ortes*. In ihren Zeilen befinden sich von unten nach oben die Werte *Interner Anker*, *Interne Seite* und *Externes Dokument*. Die zweite Spalte steht für das *Attribut des Inhalts*, mit den Werten *Ausführung*, *Bestätigung* und *Assoziation*. Die dritte Spalte zum *Attribut der Art* enthält die Werte *Linear*, *Verbunden*, *Applikativ*. Das *Attribut der Wichtigkeit* ist in der Matrix nicht berücksichtigt (siehe Abbildung 4).

<b>ORT</b>	<b>INHALT</b>	<b>ART</b>
Externes Dokument	Assoziation	Applikativ
Interne Seite	Bestätigung	Verbunden
Interner Anker	Ausführung	Linear

↑ zunehmende Unschärfe

↑ zunehmender Aufwand

*Abbildung 4:  
Die Beschreibungsmatrix.*

Die Anordnung der Werte entsprechend ihrer zunehmenden Unschärfe bzw. dem zunehmenden Grad der Informationskodierung sorgt für eine zusätzliche Sortierung des Systems im Sinne einer besseren Orientierung innerhalb der Matrix. Die quasi „dichtesten“ Informationen stehen unten. Informationen, die am „weitesten“ entfernt sind, stehen oben.

Die Matrix stellt das Kodierungs- und Dekodierungswerkzeug des grafischen Systems zur Linkbeschreibung dar.

# *Von der Beschreibung zum Zeichen und wieder zurück.*

In diesem Kapitel werden einige Modelle zur Umsetzung eines Interfaces für die Verbesserung der Aussagefähigkeit eines Links vorgestellt. Weiterhin werden die grundlegenden Forderungen und Ansätze des grafischen Systems zur Linkbeschreibung erarbeitet.



# Bestehende Interfacemodele

Wie im voraus gegangenen Kapitel dargelegt, sind die Forderungen an einen nutzerfreundlichen Hypertext zum einen die exakte Beschreibung der Links und zum anderen die Einbindung dieser Beschreibungen in ein Interface in Verbindung mit dem Link. Nachdem das System zur Beschreibung nunmehr festgelegt wurde, soll sich dieses Kapitel mit den Interfaceaspekten eines derartigen Systems befassen.

Viele Autoren und Gestalter von Hypertextdokumenten sahen sich bereits mit dem Problem der rudimentären Möglichkeiten der Linkauszeichnung in HTML konfrontiert und haben daraufhin Lösungen entwickelt, um dem Link „*das Sprechen beizubringen*“. So unterschiedlich wie die Ansätze zur Linkbeschreibung sind auch die Lösungen für Interfacemodele bezüglich des Links in Hypertextdokumenten. Einige interessante Lösungen und für die vorliegende Arbeit anregende Lösungen sollen hier vorgestellt werden (Auch zu dieser Thematik befinden sich zusätzliche Quellen auf der beiliegenden CD).

Zunächst einmal versuchen viele Gestalter von Hypertexten das Aussehen des Links zu verändern. Einige ändern lediglich die Farbe des unterstrichenen Wortes oder des Textabschnittes, der den Links auszeichnet. Wieder andere lassen den Unterstrich ganz weg und heben den Link lediglich durch eine Färbung des Wortes hervor oder hinterlegen den Link mit einer Farbfläche. Um den Link vollends in den Text zu integrieren, verzichten einige Autoren oder Gestalter auch gänzlich auf eine Hervorhebung und beschränken sich auf eine textuelle Ausweisung eines Links, wie zum Beispiel durch die Worte: *weiter*, *mehr* oder auch *hier klicken*. Bei all diesen eher gestalterischen Maßnahmen ist jedoch zu bedenken, dass oftmals dadurch der Link nicht deutlicher, sondern zumeist undeutlicher in seiner Wahrnehmbarkeit wird. Zudem ist der Aspekt zu beachten, dass durch eine gewissermaßen grundlegend in HTML vorgesehene Farbgebung eines unbesuchten, besuchten und aktvierten Links durch eine eigene Farbgebung die allgemeine Verständlichkeit dieses speziellen Aspekts der Linkauszeichnung abnimmt.

Als einfaches und leicht zu verwendendes Mittel zur Verbesserung der Aussagefähigkeit eines Links empfiehlt Jacob Nielsen, ein Usability-Spezialist in Fragen der Webseitengestaltung, die sinnvolle Verwendung der in HTML vorgesehenen Möglichkeit *link-titles* zu setzen. Zu diesem Zweck wird in den *anchor-tag* das Attribut *title* eingefügt, das eine vom Autor selbst zu verfassende Erklärung über die Art des Links enthält.

(17) Jakob Nielsen (1998)

```
<a href="destination" title="Erklärung">
```

Bei zahlreichen Browsern wird dieser Text dann beim Überstreichen des Links in Form eines kleinen Pop-Up-Fensters, mit dem Erklärungstext als Inhalt, über dem Link angezeigt. Diese Hilfefenster sehen nicht in jedem unterstützenden Browser gleich aus. Nielsen empfiehlt weiterhin die Verwendung von maximal 60 Zeichen zur Erläuterung des Links. Er betont, dass der erläuternde Text natürlich nur dann Sinn macht, wenn er eine Zusatzinformation im Vergleich zum Link selbst enthält.

Eine ähnliche Variante wie der *link-title* bietet *OverLib*, eine von Eric Bosrup entwickelte Applikation. Sie gibt dem Autor einer Webseite die Möglichkeit, die Links mit Pop-Up-Fenstern zu versehen. Die Funktionalität von *OverLib* erlaubt es, im Gegensatz zum *link-title* diese Pop-Ups zu gestalten und in ihrem Verhalten zu steuern. So können die Pop-Ups ähnlich wie beim *link-title* wieder verschwinden, wenn der Leser den Mauszeiger wieder vom Link entfernt. Es gibt jedoch auch die Möglichkeit die Pop-Ups bestehen zu lassen, wenn sie einmal aktiviert wurden; sie können dann auch beim Scrollen der Seite mitwandern.

(18) Eric Bosrup (2001)

Technisch und in Fragen der Interfaceentwicklung sehr ausgereifte Varianten der Vermittlung einer Linkbeschreibung mittels Pop-Up-Fenstern bieten beispielsweise die Lösungen *Enriched Links*, *Link Lens* und *HyperScout*. Nach dem Überstreichen oder Klicken eines Links werden dem Leser komplexe Informationen über den Link mitgeteilt. Dies können Angaben über Autoren, Erstellungsdatum, Keywords und andere Metadaten bis hin zu Angaben über den Zeitpunkt des letzten Besuches der aufzurufenden Seite und voraussichtliche Ladezeiten der Seite sein. Teilweise werden sogar kleine Thumbnails der aufzurufenden Seite angezeigt. Eine derartige Fülle von Informationen macht allerdings zunächst nur Sinn, wenn es sich um die Ver-

(19) G. Geisler (2000)

(20) D. Stanyer, R. Procter (1999)

(21) H. Weinreich, W. Lamersdorf (2000)

**General Information**

How to [Preview | Overview | History]  
 officer  
 admin  
 FORU

**Event**  
 Listing  
 upcom

**Preview of "ACM SIGIR - General Information"**

ACM SIGIR ad  
 application of co  
 and distribution o

**ACM SIGIR New:**  
 Digital Libra  
 changed to ]

Recent note

**Size: 5709 bytes**

**Overview**   **Contents**   **Channel**   **Location**

Title Iguana Cafe and Bar publishes the One Magazine

Author Electronic Media Consulting

Top Links One Magazine Kylie BlueNote Mirrorball Fotofeis

key word thumbnail

Iguana Cafe and Bar publishes the One Magazine covering film, fashion, music, clubs, people and the arts. This page describes the aspirations and ethos of the Iguana Cafe and Bar. Use this page as a spring board to the on-line version of One Magazine and our partners such as Blue Note and JDJ.

bar cafe

15 4  
internal ext  
21 1 2  
links

wrote an online guide about the design of [usable Web information systems](#).

COI Title 10 WWW-Ergonomie-Leitlinien  
 Author Harald Weinreich  
 Language German  
 Last Visit Today, 12:36  
 Bookmark 4. October 1999, 11:32

Abbildung 5:  
*Interfacedarstellungen von  
 Enriched Links, Link Lens und HyperScout  
 (von oben nach unten).*

knüpfungen zu anderen Dokumenten handelt. Innerhalb eines Dokumentes würde dies den Leser im Sinn einer effektiven Arbeit mit dem Text eher behindern als nützen.

Einen etwas anderen aber doch ähnlichen Ansatz verfolgt die Anwendung *Fluid Links*. Bei dem Interface von *Fluid Links* werden die Zusatzinformationen direkt in Textform eingeschoben, wenn der Nutzer mit dem Mauszeiger über den Link fährt. Der eingeschobene Text verdrängt den restlichen Text, der nach dem Link folgt, weiter nach unten. Es besteht weiterhin die Möglichkeit innerhalb des eingeschobenen Textes Links zu setzen, somit ist auch die Mehrfachverlinkung eines Textelementes möglich. Bei sehr hoch auflösenden Monitoren soll es auch möglich sein, den Text oder Teile des Textes direkt an oder unter dem Link zu platzieren.

(22) P. T. Zellweger ...

Alle bis hierhin erwähnten Möglichkeiten eines Linkinterfaces haben die gemeinsame Eigenschaft, dass der Nutzer sich die Information über den Link quasi mit einer Bewegung der Maus erarbeiten muss. Es gibt aber auch Lösungen für ein Link-Interface, bei denen die Information über den Link immer und ohne Zutun des Lesers sichtbar ist.

Eine Variante zu diesem Lösungsansatz sind zum Beispiel die von Christopher S. Campbell und Paul P. Magilio entworfenen *Road-Signs*. In ihrem Aufsatz „*Fascilitating navigation in information spaces: Road-signs on the World Wide Web*“ beschreiben sie kleine farbige Rechtecke, die vor und nach einem Link im Text stehen. Durch ihre Farbigkeit vermitteln sie dem Benutzer, ob sich eine entsprechende verknüpfte Seite schnell, normal oder langsam laden lässt. Diese Information wird vor der Anzeige der Seite automatisch ermittelt und die Zeichen dementsprechend im Text generiert. In Untersuchungen zu den „Road-Signs“ weisen Campbell und Magilio nach, dass bei derartigen Symbolen die einen Link beschreiben, die Entscheidungszeit der Nutzer einen Link auszuwählen, signifikant kleiner wird im Vergleich zu herkömmlichen unkommentierten Links.

(23) C. S. Campbell, P. P. Magilio (1999)

Beispiele für ein Interface, das dem Leser die Information über den Link sofort sichtbar präsentiert, sind die vom Designbüro Matterform entworfenen *QBullets*. Die *QBullets* sind eine Sammlung von kleinen, zum Teil animierten Icons, die in Links eingefügt werden können, um deren Funktion zu beschreiben. Es gibt Navigations-

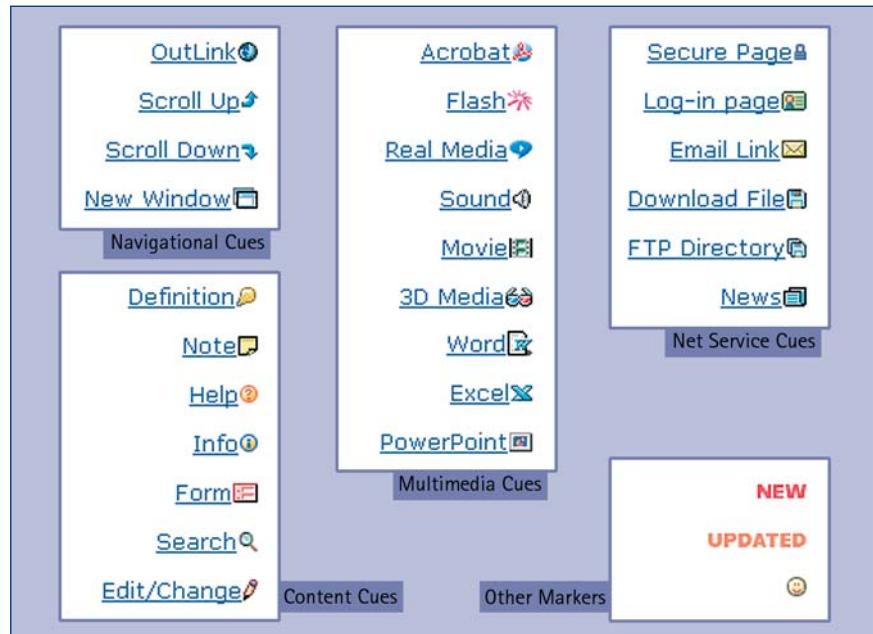


Abbildung 6:  
Die QBullets des  
Designbüros Matterform.

(24) Matterforms Qbullets (2001)

symbole (*Externer Link, Scroll Up, Scroll Down, neues Fenster*), Multimediasymbole (*Acrobat, Flash, Real Media, Sound, Movie, 3D Media, Word, Exel, PowerPoint*), Inhaltssymbole (*Definition, Fußnote, Hilfe, Info, Formular, Suche, Bearbeiten/Ändern*), Netzsymbole (*Gesicherte Seite, Log-In Seite, E-Mail, Download, FTP Verzeichnis, Newsseite*) und einige Zusatzsymbole (*Neu, Geändert, Smileys*).

Die QBullets dürfen nach bestimmten Regeln frei und kostenlos benutzt werden. Sie dürfen nicht verändert werden, müssen ihrer Definition entsprechend eingesetzt werden und sollen immer hinter dem Link stehen.

Neben den hier genannten Beispielen lassen sich noch einige mehr im WWW finden. Für alle jedoch gilt, dass keiner der Ansätze eine Verbreitung und Akzeptanz im Sinne eines quasi Standards im WWW gefunden hat. Vom Einsatz des *link-titles* als Bestandteil eines Standards sei dabei abgesehen.



# Entwurf des grafischen Interfaces

Einer der in den Beispielen des vorangegangenen Abschnittes angeführten Lösungsansätze entspricht am ehesten dem in dieser Arbeit umgesetzten Entwurf eines Interfaces für ein Beschreibungssystem der Eigenschaften von Links. Es ist der Ansatz, eine Beschreibung direkt dem Link anzuhängen, ähnlich der Linkbeschreibung durch die Zeichen der vorgestellten *Road Signs* oder der *Qullets*.

Dieser Ansatz ist im Prinzip die nutzerfreundlichste Variante. Eine derartige Benutzeroberfläche zeigt die Informationen, die sie in sich trägt sofort und muss nicht durch eine zusätzliche Aktivität neben dem Lesen erschlossen werden.

Die Varianten, die mit Pop-Up-Fenstern arbeiten, verstecken ihre Informationen zunächst und müssen vom Nutzer erst durch eine aktive Komponente, der Mausbewegung, zugänglich gemacht werden. Sicherlich ist jeder Link an sich ein interaktives Element, jedoch ist es ein Unterschied, ob diese Interaktion schon vor oder erst nach der Entscheidung über das Verfolgen eines Links stattfindet. Muss der Leser eines Hypertextes sich jeden Link erst durch eine Interaktion erschließen, wird er permanent aus dem Prozess des Lesens herausgerissen und die Rezeption eines Hypertextes wird neben der ohnehin schon vorhandenen Schwierigkeit des Lesens am Bildschirm noch zusätzlich erschwert. Bei einer Oberfläche, die Link und Beschreibung gleichzeitig repräsentiert, wird der Leser zwar auch in seinem Leseffluss gestoppt, wenn er an einen Link gerät, er verlässt jedoch die Ebene des Lesens nicht. Dies ist in der Regel nicht so ablenkend wie eine zusätzliche Interaktion.

Will man die Informationen, die beispielsweise in einem Pop-Up-Fenster von *Enriched Links*, *Link Lens* und *HyperScout* gespeichert sind, auf einer Oberfläche vereinen, bliebe nur die Möglichkeit, diese Informationen dem Link in Form eines erläuternden Textes anzufügen. Dies würde jedoch zu einer enormen Aufblähung des Textes führen, die ebenso den Leseffluss stören würde.

Bei dem im vorherigen Kapitel vorgestellten Beschreibungssystem wäre eine derartige Lösung sicherlich auf Grund der maximal drei bis vier genannten Werte noch einigermaßen umsetzbar. Jedoch scheint es dann auch noch nicht die optimale Lösung zu sein.

Um die Information zwar auf der Oberfläche und Ebene des Links zu positionieren, aber sie zusätzlich noch zu komprimieren, liegt eine Umsetzung der Information in Form spezieller Zeichen nahe.

Die Kodierung von Information in Form von Zeichen verstößt zwar auch wieder in gewisser Weise gegen die Regeln der Sichtbarkeit von Information, ein optimales Zeichen kann die Informationserfassung aber auch beschleunigen und vereinfachen. Entscheidend dabei ist, wie viel Information im Zeichen kodiert ist und wie sich diese Information in der Form des Zeichens repräsentiert. Harmonieren Form und Information nicht miteinander, wird das Zeichen missverständlich und bedarf einer zusätzlichen Erklärung.

Der Entwurf eines Zeichens zur Kodierung von Information stellt sich demnach wenig trivial dar. Eine zusätzliche Herausforderung wird die Suche nach den Zeichen, wenn sie dem im vorherigen Kapitel entworfenen Beschreibungssystem entsprechen sollen. Dabei sollen auch noch zusätzliche Anforderungen an die Form und Darstellung der Zeichen erfüllt werden, die sich aus dem Wunsch einer optimierten grafischen Umsetzung ergeben. Diese Anforderungen sind:

- Die Zeichen sollen möglichst kompakt sein und sich dem Text in seiner Form und Farbe unterordnen.
- Die Zeichen müssen eine Vielzahl von Informationen kodieren können, entsprechend den Möglichkeiten des Beschreibungssystems.
- Die in den Zeichen kodierte Information muss eindeutig erkennbar sein.
- Die Zeichen sollen trotz der Vielzahl der in ihnen kodierten Informationen ähnlich sein.
- Die Zeichen sollen ebenso skalierbar wie Text sein und auch bei einer kleinen Größe noch erkennbar dargestellt werden können.

Zeichen die nach diesen genannten Forderungen gestaltet wären, entsprächen in gewisser Weise einem Optimum der grafischen Kodierung von Information, die harmonisch in Lesetext integriert werden soll.

An den genannten Forderungen zeigt sich sehr gut der Kritikpunkt an den Zeichen der *QBullet*s. Während die Zeichen der *QBullet*s durch ihre Farbigkeit und Gestaltung aus dem Text heraustreten und sich nicht dem Lesetext unterordnen, sollen sich die zu entwerfenden Zeichen ähnlich wie Schriftzeichen verhalten.

Ein gutes Beispiel für die Form und den Ausdruck derartiger Zeichen stellt das @-Zeichen dar. Als Teil einer E-Mail-Adresse kennzeichnet es diese durch seine Anwesenheit sofort und deutlich, ohne sich dabei störend auf das Gesamtbild des Textes und den Lesefluss auszuwirken. Es ist eindeutig und lässt sich beim Überlesen eines Textes durch seine, sich etwas von den restlichen Buchstaben abhebende Form, recht leicht finden.

Würden sich also die zu entwerfenden Zeichen ebenso verhalten, wäre man dem Optimum einer grafischen Linkbeschreibung ein ganzes Stück näher.



# Entwurf der Zeichenkodierung

(25) Marti A. Hearst; (1995)

Einen interessanten Ansatz für die Umsetzung eines grafischen Systems findet man in dem Aufsatz von Marti A. Hearst „*TileBars: Visualization of Term Distribution Information in Full Text Information Access*“. Das von Hearst beschriebene System dient zwar nicht direkt der Linkbeschreibung, es ist aber trotzdem eine bemerkenswerte Variante der Umsetzung von komplexen Zusammenhängen in Form einer Grafik.

*TileBars* sind Visualisierungen von Suchergebnissen in Textdatenbanken. Sie illustrieren die relative Dokumentlänge, die Häufigkeit des Auftretens des gesuchten Begriffes und die Verteilung des Suchbegriffes innerhalb des gefundenen Dokumentes. All diese Angaben sind aus einer leicht verständlichen Grafik ablesbar.

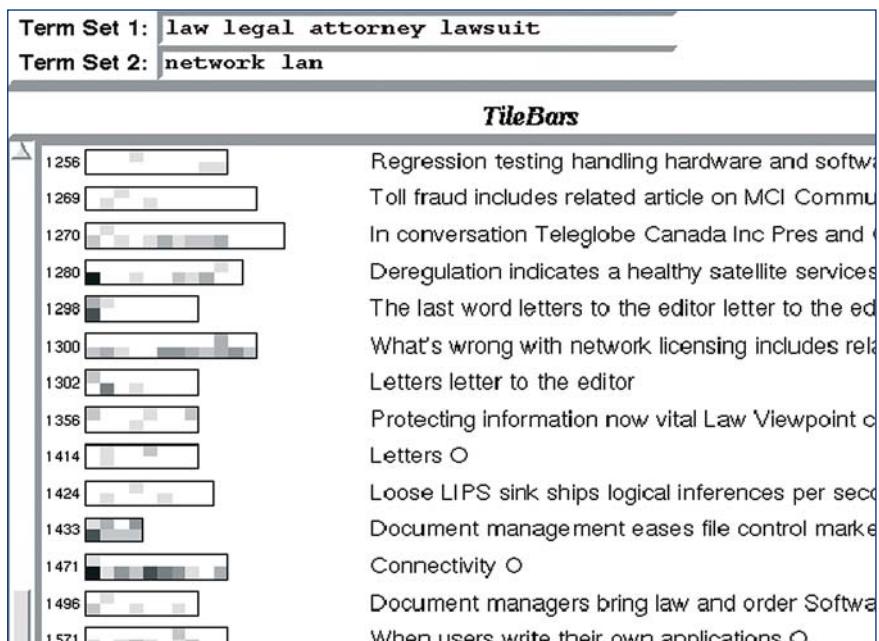


Abbildung 7:

Als Ergebnisanzeige auf eine Suchanfrage mit zwei Suchbegriffen stehen neben den Titeln der Texte zweizeilige TileBars.

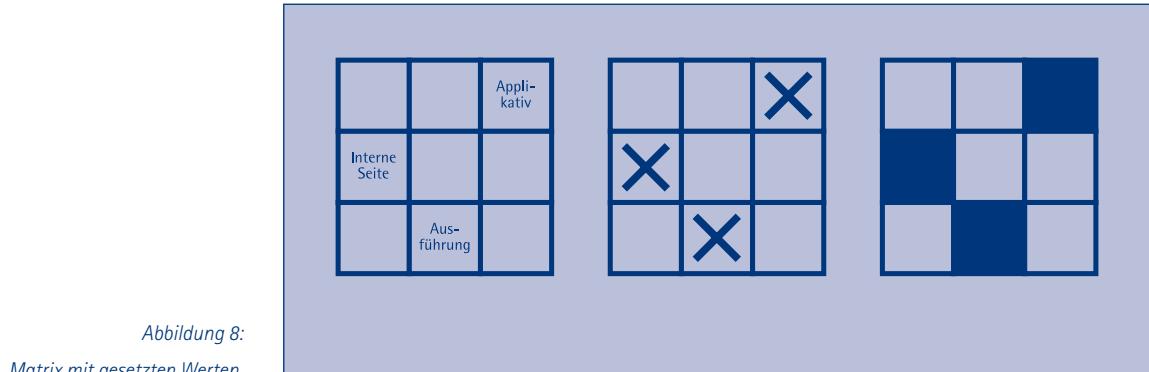
Nach der Eingabe eines oder mehrerer Suchbegriffe in eine Abfragemaske wird das Suchergebnis in Form einer Liste der gefundenen Texttitel angezeigt. Neben jedem Titel steht ein *TileBar* (siehe Abbildung 7). Die *TileBars* haben die Form eines gerasterten Balkens, ähnlich einem Balkendiagramm. Die Länge des Balkens gibt die relative Länge des Dokumentes im Vergleich zu den anderen gefundenen Dokumenten an. Die Länge eines gefundenen Textes lässt sich somit sofort ablesen.

Entsprechend der Anzahl der gesuchten Begriffe sind die *TileBars* in Zeilen unterteilt. Bei der Suche nach zwei Begriffen hat ein *TileBar* also zwei Zeilen. Die erste Zeile steht für den ersten Suchbegriff und die zweite für den anderen Suchbegriff. Jede Zeile ist wiederum in einzelne Segmente aufgeteilt, die die Abschnitte des Textes darstellen. Tritt ein Suchbegriff in einem Abschnitt des Textes auf, wird das entsprechende Segment eingefärbt. Diese Farbigkeit richtet sich nach der Häufigkeit des Auftretens des Suchbegriffes innerhalb des Abschnittes. Sind also bei einem zweizeiligen *TileBar* zwei übereinander stehende Segmente eingefärbt, bedeutet dies, dass beide gesuchten Begriffe gleichzeitig in einem Abschnitt des Textes auftreten (siehe Abbildung 7).

Die Segmente der *TileBars* sind mit den entsprechenden Abschnitten in den Texten verlinkt, sodass der Nutzer direkt über die *TileBars* zu einem Abschnitt springen kann.

Die *TileBars* sind eine beeindruckende Variante der Kodierung von Information in Zeichen. Sie vermitteln dem Leser sehr komplexe Informationen über einen Sachverhalt. In schriftlicher Form könnte man diese Information nicht so umfassend und exakt vermitteln. Diese Leistung können die *TileBars* jedoch nur erfüllen, wenn der Inhalt der Texte entsprechend mit Metadaten versehen wurde, das heißt, dem Text müssen zusätzliche Informationen gegeben werden, damit sich die Kodierung in ihrer Komplexität überhaupt umsetzen lässt.

Bei den in dieser Arbeit zu entwerfenden Zeichen handelt es sich ebenfalls um die Kodierung von komplexen Informationen. Es stellt sich also die Frage, wie und welche der Informationen kodiert werden sollen. Der Versuch der Visualisierung der einzelnen Werte ist ziemlich aussichtslos. Für einen Wert wie *Assoziation* lassen sich beispielsweise keine treffenden visuellen Aussagen finden. Außerdem würden Zeichen, die jeweils nur ein Attribut beschreiben, dem Ansatz widersprechen, den



Link möglichst komplex und mittels eines einzigen Zeichens zu beschreiben. Es musste also eine andere Art der zu kodierenden Information gefunden werden oder nach einer zu kodierenden Metainformation gesucht werden, die die Informationen des Beschreibungssystems zusammenfasst.

Die Metainformation, die Kodierungsgrundlage für die Zeichen sein soll, wurde in Form der *Beschreibungsmatrix* gefunden.

Die *Beschreibungsmatrix* ist eine formale Präsentation des ihr zugrunde liegenden Beschreibungssystems. Sie stellt das Raster zur Kodierung der Information über den Link dar. Bei Kenntnis der Anordnung der Attribute und Werte innerhalb dieses Rasters lassen sich durch das Setzen von Werten in der Matrix die Werte für die Linkbeschreibung ablesen. Aus der Rasterung der Matrix ergibt sich also eine sehr gute Basis für die Herleitung der Zeichen, zumal die Matrix selbst schon formalen Charakter besitzt. Der Aufwand einer Informationskodierung in einem Zeichen ist dadurch relativ gering.

Um die Zeichen, die auf einer Präsentation der Matrix beruhen, verstehen zu können, ist jedoch die Kenntnis der *Beschreibungsmatrix* zwingend notwendig. Dies ist ähnlich wie bei den *TileBars*. Ohne das Wissen über ihre Funktion sind die *TileBars* nicht verständlich. Jedoch mit diesem recht leicht erlernbaren und vermittelbaren Wissen erreichen die *TileBars* ihre Stärke in der Informationsvermittlung. Gleichermaßen gilt für das Wissen um den Aufbau der *Beschreibungsmatrix*.



# *Von Möglichkeiten und Machbarkeiten.*

Vor der Beschreibung der endgültigen Umsetzung des grafischen Systems im folgenden Kapitel, werden in diesem Kapitel vorab noch die Aspekte einer technischen Umsetzung eines derartigen Systems erörtert.



# Die optimale technische Lösung

Die Realisierung eines grafischen Systems zur Beschreibung von Links beinhaltet neben der Frage der Erstellung des Beschreibungssystems einerseits und der Frage der Umsetzung in Form eines Interfaces andererseits auch noch die Frage nach einer technischen Realisierbarkeit. An dieser Stelle soll zunächst das Optimum einer technischen Realisierung aufgezeigt werden.

Die Mehrzahl der Hypertextdokumente des WWW sind heutzutage mittels HTML beschrieben. Das heißt, Autoren eines derartigen Dokumentes erstellen die Links in ihren Hypertextkonstruktionen mittels HTML. Dies ist in einfachen Textprogrammen möglich oder mittels spezieller Programme, die dem Autor bei der Erstellung ihrer Hypertexte helfen. Auch wenn einige dieser HTML-Authoring-Tools den Autor durch bestimmte Entwurfsoberflächen fast gar nicht mehr mit dem HTML-Code konfrontieren, kann man davon ausgehen, dass ein Grossteil der Autoren von HTML-Dokumenten mit den Mechanismen der Linkerzeugung mittels HTML vertraut sind.

Auf Grund der Voraussetzung dieses Kenntnisstandes der Autoren und auf Grund der Überlegung, dem Autor die Verwendung des entworfenen Modells zur Linkbeschreibung so einfach wie möglich zu gestalten, wäre es für ein derartiges System also die optimale technische Lösung, wenn es direkt Teil der Linkauszeichnung mittels HTML wäre.

In HTML sind zwar schon einige Typen von Linkbezeichnungen als *data types* implementiert, aber wie gezeigt sind diese Linktypen zur Beschreibung des Links nur bedingt brauchbar und könnten somit im Prinzip durch das hier entworfene System ersetzt werden. Eine Implementierung dieses Systems ist auch auf Grund der Verwendung von Attributen und Werten in dieser Beziehung gut umsetzbar.

Die Implementierung des Beschreibungssystems innerhalb des *anchor-tags* könnte demnach folgendermaßen aussehen:

```
<a href="zieladresse" AttributOrt="Wert" AttributInhalt="Wert"  
AttributArt="Wert" Wichtigkeit=ja>
```

Eine genaue Benennung der Attribute und Werte entsprechend der Nomenklatur der HTML sei dabei noch nicht berücksichtigt. Die Bezeichnungen der Attribute und Werte sollten natürlich der englischen Sprache angepasst sein und möglicherweise auch noch verkürzt werden, um den HTML-Code einer Seite nicht mehr als nötig auszuweiten.

Man könnte natürlich auch nur ein Attribut zur Linkbeschreibung verwenden und diesem mehrere Werte zuordnen, aber im Sinne der Übersichtlichkeit und um eine Fehlverwendung der Werte zu vermeiden (beispielsweise die Verwendung von zwei Werten aus einem Attribut), ist es besser, die Attribute einzeln zu verwenden. Wenn ein Attribut nicht gesetzt werden soll, kann es innerhalb des *anchor-tags* weglassen werden.

Die Interpretation und eine entsprechende Darstellung eines derartigen Links würde dann der Browser übernehmen, der zur Betrachtung der Seite verwendet wird. Die Zeichen würden also durch den Browser im Moment der Darstellung einer Seite erzeugt, entsprechend der gesetzten Attribute. Eine richtige Sortierung der Attribute durch den Autor innerhalb des *anchor-tags* wäre nicht nötig, da diese Teil des Ausleseprozesses der Attribute des *anchor-tags* durch den Browsers wäre und die Eindeutigkeit der Attribute und Werte Fehlinterpretationen auf Grund der Sortierung ausschließt.

Einen weiteren Vorteil hätte die Implementierung des Beschreibungssystems in den *anchor-tag* auch für Suchmaschinen. Durch das Auslesen der Attribute ließe sich eine genauere Bewertung der Links im Sinne ihrer Relevanz realisieren.

# Annäherungen an das Optimum

Da es dieser Arbeit einerseits nicht ansteht, Teile der HTML neu zu definieren, andererseits aber auch durch ein Festhalten an dieser Form der Realisierung das in dieser Arbeit beschriebene Modell nur einen reinen Entwurfs-Charakter hätte, soll nunmehr nach Lösungen gesucht werden, die eine Umsetzung des Systems im Sinne einer Annäherung an das Optimum darstellen könnten.

Die wohl trivialste Lösung der Umsetzung der Zeichen innerhalb eines HTML-Dokumentes wäre die Umsetzung der Zeichen als Bilder in einem webtauglichen Format und die Einbindung dieser Bilder mittels HTML in den Text.

Die Einbindung als Bild ist prinzipiell kein Problem, jedoch der Umgang mit den Bildern. Zunächst müsste man alle aus der *Beschreibungsmatrix* ableitbaren Zeichen erstellen. Dies betrifft nicht nur die aus der 3er-Matrix generierbaren Zeichen, sondern nochmals alle Zeichen in Verbindung mit der Erweiterung für das *Attribut der Wichtigkeit*. Um wenigstens annähernd die genannten Forderungen zu erfüllen, dass die Zeichen skalierbar sein sollten und sich an die Textgröße anzupassen haben, sollten alle Zeichen nach Möglichkeit auch für verschiedene Skalierungsgrade optimiert sein, also in verschiedenen Größen vorliegen. Der Autor hätte dann die Möglichkeit, die Bildgröße entsprechend seiner gewählten Schriftgröße zu wählen. Um das so entstehende Gesamtbild des Textes zu erhalten, sollte in diesem allerdings die Schriftgröße fest definiert werden.

Alle erstellten Zeichen müssten in einer Bild-Sammlung mit eindeutigen Namen für jedes Bild zusammengefasst werden. Diese kann dem Autor von Hypertextseiten zur Verfügung gestellt werden, um aus ihr die entsprechenden Bilder zu entnehmen. Der Autor sollte diese Sammlung nach Möglichkeit auf dem Server, der auch sein Dokument enthält, ablegen.

Man könnte diese Sammlung natürlich auch zentral auf einem Webserver hinterlegen und der Autor könnte die Bilder von diesem Server beziehen. Hier könnte aller-

dings ein Problem entstehen, wenn dieser Server mal abgeschaltet oder vorübergehend nicht erreichbar sein sollte.

Neben den Fragen der Distribution der Bilder besteht zusätzlich noch das Problem der Ladezeiten für eine Seite, die viele dieser Bilder enthalten würde. In Zeiten von DSL und anderen Hochgeschwindigkeitszugängen ist dieses Problem zwar nicht mehr so relevant, aber derartige Zugänge sind noch nicht Standard und können daher nicht vorausgesetzt werden.

Einige genannte Probleme, die auf den Eigenschaften von Bildern in einer Formatierung für das WWW beruhen, könnten sicherlich durch den Einsatz von SVG, einem neuen Bildformat für das WWW, umgangen werden. SVG-Bilder sind Vektorbilder und lassen sich somit beliebig skalieren. In einen Text eingefügt, würden sie sich zusammen mit dem Text skalieren lassen. Sie haben außerdem im Vergleich zu herkömmlichen Bildformaten für das WWW eine geringere Datengröße. Um SVG-Seiten betrachten zu können, benötigt der Leser für den von ihm verwendeten Browser allerdings ein entsprechendes *Plug-In*. Dem Autor wiederum sollte nach Möglichkeit ein SVG-fähiges Layoutprogramm zur Erstellung der Seiten zur Verfügung stehen. SVG-Code ist zwar auch von Hand editierbar, kann aber bei komplexen Seiten sehr unübersichtlich werden.

Das größte Problem bei der Art der Realisierung mittels vorgegebener Bilder wäre jedoch nicht so sehr technischer Natur, sondern liegt wohl eher im absolut unzumutbaren Umgang für den Autor mit diesen Bildern. Schon die Auswahl des entsprechenden Bildes aus einer umfangreichen Sammlung von sehr ähnlichen Bildern würde viel Zeit in Anspruch nehmen, von der Frage der notwendigen, einzelnen Einbettung der Bilder in den Text einmal abgesehen. Eine derartige Umsetzung kam daher nicht in Frage.

Um die Benutzung der Zeichen für den Autor zu vereinfachen und unter Betrachtung der im vorherigen Abschnitt aufgezeigten optimalen Lösung wäre zur Realisierung auch folgende Variante möglich: Man könnte einen eigenen *tag* zur Beschreibung des Links festlegen. Dieser *Beschreibungs-tag* könnte ähnlich der in der optimale Lösung erläuterten Vorgehensweise der Linkbeschreibung mittels der Attribute und Werte funktionieren.

Der Autor würde diesen *tag* zunächst einfach in seinen Text hinter dem Link einfügen.

Der nächste Schritt wäre dann im Prinzip ein trivialer „Suchen und Ersetzen“- Algorithmus.

Eine denkbare Applikation müsste, bevor die Seiten im Browserfenster des Lesers dargestellt werden, den *tag* durch ein Zeichen ersetzen. Eine derartige Lösung lässt sich auf vielen Wegen realisieren, beispielsweise mit PHP oder JavaScript. Möglich wäre auch eine Umsetzung mittels Style Sheets. Statt des *tags* könnte man dazu *Klassen* definieren, die dann, durch das Style Sheet gesteuert, entsprechend dargestellt werden.

Die Verwendung der Zeichen durch den Autor wäre durch eine derartige Variante wesentlich einfacher. Die Umsetzung mittels eines *Beschreibungs-tags* stellt an den Autor jedoch zusätzliche Forderungen, bezüglich seiner technischen Kenntnisse der Dokumenterstellung für das WWW, die über die Erstellung mittels HTML hinausgehen. Neben der Frage der Distribution der Zeichen kann es bei derartigen Varianten zusätzlich noch zu Fragen der Distribution von „Bausteinen“ zur Erstellung der Seiten kommen. Somit scheinen auch derartige Lösungen noch nicht der richtige Weg zu sein, zumal sie auch zunächst nur die Benutzbarkeit des Systems beschreiben, jedoch die Fragen der Darstellung der Zeichen nicht beinhalten.

Die Lösung, die sich technisch am ehesten dem beschriebenen Optimum in Fragen der Benutzbarkeit und Darstellung der Zeichen annähert, ist die Umsetzung der Zeichen als Schrift. Schrift vereinigt beide Aspekte in einer Anwendung.

Schrift ist leicht benutzbar. Durch die Tastatur eines Rechners bietet sich im Umgang mit Schrift ein direktes Eingabemedium an. Das heißt, die Erzeugung der Zeichen durch den Autor könnte über einfache Eingaben mittels der Tastatur umgesetzt werden. Zum anderen lassen sich Zeichen gut in Form einer Schrift umsetzen. Wenn die Zeichen selbst als Schrift vorliegen, lassen sie sich einfach in Texte integrieren. Vergrößert man den Schriftgrad eines Textes beispielsweise im Browser, werden diese Zeichen entsprechend mit skaliert.

Um dargestellt werden zu können, muss die entsprechende Schrift jedoch auf dem Rechner des Autors und des Lesers vorhanden sein. Das heißt, auch bei der Umsetzung der Zeichen in einer Schrift, muss diese zunächst Leser und Autor zur Verfügung gestellt werden. Durch ihr kleines Datenformat und durch die einfache Installation einer Schrift bietet sie jedoch entscheidende Vorteile im Gegensatz zu oben genannten Varianten der Umsetzung. Da die Installation einer Schrift auf der Ebene des Betriebssystems eines Computers stattfindet, kommt noch der Vorteil hinzu, dass sie nach einmaliger Installation in allen textverarbeitenden Programmen benutzbar ist. Sie steht somit beispielsweise im Browser als auch in einem HTML-Autoring-Tool gleichzeitig zur Verfügung. Dies ist für den Autor bei der Erstellung eines Dokumentes unter Verwendung der Schrift von großem Nutzen. Denn er hat dadurch die Möglichkeit, seine Dokumente sofort zu kontrollieren und zu testen.

Ein Nachteil der Verwendung einer speziellen Schrift bei der Erstellung von Webseiten ist deren Einbettung mittels HTML. Verwendet der Autor nur einen einfachen Texteditor zur Erstellung seiner HTML Dokumente, muss er jede Einbettung der Schrift von Hand auszeichnen. Je nach Menge der verwendeten Zeichen (also Links) in einem Text kann dies einen nicht zu vernachlässigenden Zeitaufwand bedeuten. Da aber die meisten HTML Dokumente heutzutage mittels spezieller HTML-Autoring-Tools erstellt werden, ist dieses Problem nicht mehr allzu relevant. In derartigen Programmen reicht oft ein bloßes Auswählen eines Textabschnittes und die Auswahl einer entsprechenden Schrift für den Textabschnitt.

Die Auszeichnung in HTML erfolgt dann in der Regel automatisch.

Ein weiterer Nachteil soll hier noch genannt werden: Führt man sich Menge der möglichen Zeichen vor Augen, wird schnell klar, dass eine sinnvolle und übersichtliche Zuordnung der Zeichen zu einer Tastaturbelegung nicht möglich ist. Eine spezielle Lösung dieses Problems soll jedoch im folgenden Kapitel vorgestellt werden.

# *Es werde Schrift.*

In diesem Kapitel werden die Zeichen des grafischen Systems entworfen und vorgestellt. Weiterhin wird die finale Umsetzung der Zeichen in Form einer Schrift beschrieben.



# Die Entwicklung der Zeichen

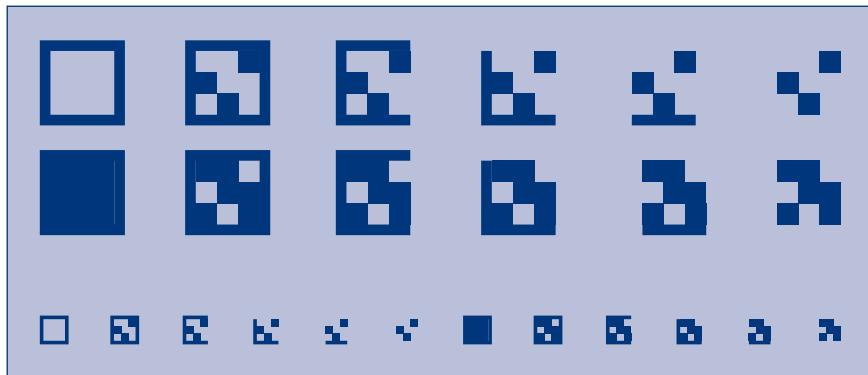
Wie bereits erwähnt, sollen die Zeichen zur Beschreibung von Links keine Präsentationen der Attribute und Werte sein, sondern die Form der *Beschreibungsmatrix* darstellen. Dazu soll zunächst die Form der Matrix analysiert werden, um mögliche Ansatzpunkte für die Herleitung der Zeichen zu ermitteln. Gleichzeitig ist auch der Aspekt der Umsetzung der Zeichen als Teile einer Schrift zu beachten.

Die *Beschreibungsmatrix* ist eine 3er-Matrix, das heißt, sie besteht aus jeweils drei Segmenten in der Höhe und in der Breite. Dies ergibt insgesamt neun Segmente innerhalb der gesamten Matrix, die mit Werten besetzt werden können.

Den Spalten sind die drei Attribute zugeordnet. Die Zeilen stellen nur die Zahl der möglichen Werte dar, also drei Werte pro Attribut.

Die Korrespondenz von Höhe und Breite der Matrix hat für die Aussage der Matrix keine Bedeutung. Sie ist also in beiden Richtungen beliebig horizontal oder vertikal skalierbar. Wichtig dabei ist lediglich, dass die Größen der Segmente untereinander immer gleich bleiben, da es sonst zu einer Gewichtung von einzelnen Segmenten kommen würde. (Dies würde einer Hervorhebung einzelner Werte eines Attributes gleichkommen, was aber keinen Sinn macht, da der Wert an sich noch keine Aussage im Sinne einer Relevanz oder dergleichen in sich birgt) Egal in welcher Form die Matrix also dargestellt wird, die Symmetriearchsen der Matrix sollten immer bestehen bleiben.

Interpretationen der Matrix entsprechend der oben genannten Eigenschaften sind vielfach möglich. Es gilt dabei jedoch, ein ausgewogenes Verhältnis zwischen Begrenzungsmerkmalen der Matrix (Rand) und der Darstellung der gesetzten Werte zu finden. Zu beachten ist dabei, dass das Zeichen für die Matrix auch sehr klein verwendet werden soll. Abbildung 9 zeigt Varianten einer quadratischen 3er-Matrix, mit dem Segment/Randstärke-Verhältnis von 2:1. Deutlich zu erkennen ist das notwendige Vorhandensein eines Randes, also eines Begrenzungselementes, der die Matrix überhaupt erst als solche erkennbar macht. Ohne Rand verschwimmen die



*Abbildung 9:  
Varianten der grafischen Umsetzung  
einer 3er-Matrix mit dem  
Segment/Randstärke-Verhältniss von 2:1.*

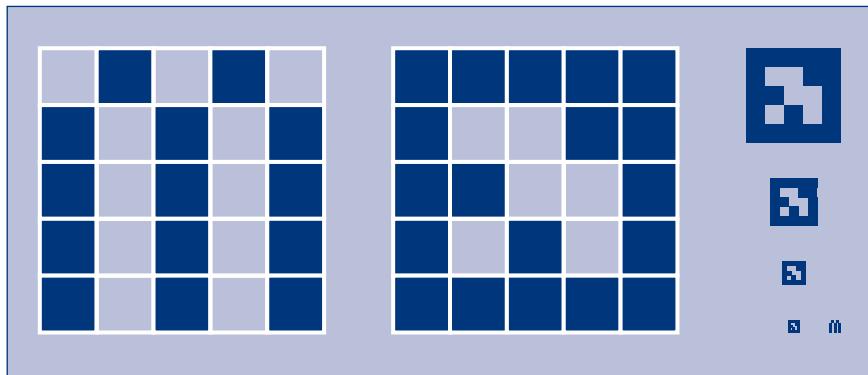
Hypertext is text which contains [links](#) to other texts.  
The term was coined by [Ted Nelson](#) around 1965 (see [History](#)).

Hypertext is text which contains [linksm](#) to other texts.  
The term was coined by [Ted Nelsonm](#) around 1965 (see [Historym](#)).

Hypertext is text which contains [nlinksm](#) to other texts.  
The term was coined by [nTed Nelsonm](#) around 1965 (see [nHistorym](#)).

Hypertext is text which contains [nlinksn](#) to other texts.  
The term was coined by [nTed Nelsonn](#) around 1965 (see [nHistoryn](#)).

*Abbildung 10:  
Anordnung und Unterordnung eines  
Zeichens zur Linkbeschreibung  
in das Textbild und Einfluss dieses  
Zeichens auf die Textlänge.*



*Abbildung 11:  
Umsetzung einer 3er-Matrix auf einem  
Raster von 5 Pixel in der Höhe, das bei  
einer Schriftgröße von 9 Pixel entsteht.  
rechts unten: ca. Originalgröße.  
Zur Veranschaulichung ist die  
Darstellung eines vereinfachten kleinen  
„m“ gegenübergestellt.*

Segmente zu einer Form. Am anschaulichsten ist dies vor allem an den invertierten Darstellungen der Matrix. An den Beispielen der Abbildung 9 wird weiterhin deutlich, dass die Segmente, um sie klar von einander trennen zu können, möglichst einen Abstand zwischen einander als auch zum Rand hin haben sollten. Ohne diese Abgrenzungen würden die Zeichen bei einer sehr kleinen Reproduktion zu einer undeutlichen Form verschwimmen.

Das Zeichen sollte sich, wie erwähnt, möglichst der Schrift des restlichen Textes unterordnen, um harmonisch in das Textbild integriert werden zu können. In der normalen Schrift dominiert die Höhe der Minuskeln (Kleinbuchstaben) über der Grundlinie der Schrift das Gesamtbild der Zeile eines Textes. Auffälliger ist dieser Effekt im englischen Schriftbild auf Grund des noch selteneren Vorhandenseins von Versalien (Großbuchstaben). Da das Zeichen zur Linkbeschreibung diesem angehängt werden soll, muss es sich mit dem Wort, das den Link beschreibt harmonisch verbinden. Dazu darf es jedoch nicht über die Höhe der Minuskeln hinausragen und sollte auch nicht die Grundlinie der Schrift unterschreiten (siehe Abbildung 10).

Diese Höhe, auch x-Höhe genannt, ist also das Maß für die Höhe des Zeichens. Die Breite des Zeichen ist dabei nicht so entscheidend, sie sollte sich jedoch an der Breite der Buchstaben einer Schrift orientieren (siehe auch Abbildung 10).

Da das Zeichen, vor allem was seine Erkennbarkeit anbelangt, auf die Gegebenheiten der Monitors angepasst werden muss, ergeben sich Bedingungen, die für die Gestaltung des Zeichens von entscheidender Bedeutung sind. Bei einer neun Pixel großen Schrift stehen zur Umsetzung der Minuskeln lediglich noch fünf Pixel in der x-Höhe zur Verfügung. Diese fünf Pixel stellen also das entscheidende Rastermaß in der Höhe der Zeichen dar, denn eine entsprechende neun Pixel große Schrift markiert in der Regel die Grenze der Lesbarkeit am Monitor.

Es gibt zwar einige spezielle Pixelschriften die mit einer x-Höhe von vier Pixel auskommen und trotzdem noch gut am Monitor lesbar sind, sie haben jedoch vor allem eine gestalterische Bedeutung. Bei Lesetext, wie er im definierten Anwendungsfeld des hier beschriebenen Zeichensystems vorrangig auftritt, spielen derartige Schriftgrößen keine entscheidende Rolle.

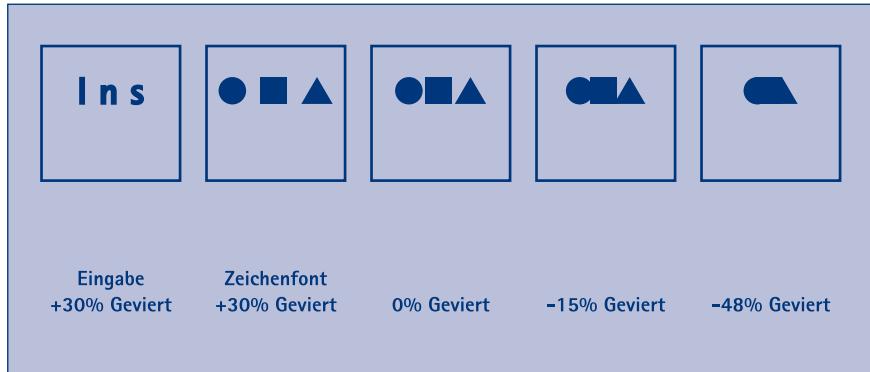
Das unterste Maß zur Umsetzung der Zeichen sollte also eine x-Höhe von fünf Pixel darstellen. Führt man sich die Übertragung eines der in Abbildung 9 gezeigten Zeichen auf ein Raster von fünf Pixel in der Höhe vor Augen, werden schnell die Probleme und Grenzen einer Umsetzung der Zeichen bewusst. Wie in Abbildung 11 gezeigt ist dabei eine Unterscheidung in Rand und Segment, was die Größenverhältnisse und Wichtungen anbelangt, nicht mehr möglich. In Originalgröße am Monitor wäre bei einem derartigen Zeichen die Unterscheidung einzelner markierter Segmente schlecht möglich.

Aus der *Beschreibungsmatrix* lassen sich zahlreiche Kombinationen von Werten generieren. Jede zulässige Kombination sollte auch als Zeichen zur Linkbeschreibung vorhanden sein. Dies würde eine große Anzahl von Zeichen bedeuten, die in Form der Schrift zusammengefasst werden müssten. Da jeder Wert aber eine eindeutige Präsentation innerhalb des einzelnen Zeichens, als auch innerhalb aller möglichen Zeichen haben muss, liegt die Überlegung nahe, das Zeichen aus Teilzeichen zusammenzusetzen, die jeweils für einen Wert stehen. In diesem Fall bedarf es dann nur neun Teilzeichen, von denen jeweils drei zusammengefügt, das Gesamtzeichen ergeben würden (jedes der drei mit Werten zu versehende Attribut darf genau einen Wert haben, also ergibt sich eine Kombination aus drei Teilzeichen). Neben den neun Teilzeichen für die gesetzten Werte benötigt man allerdings noch drei Teilzeichen für die nicht mit Werten besetzten Attribute, um die Gesamtform des Zeichens zu erhalten. Es ergeben sich also insgesamt nur zwölf Teilzeichen, im Gegensatz zur Umsetzung aller möglichen Zeichen. Dies bedeutet eine enorme Erleichterung des Umgangs mit den Zeichen für den Anwender.

Im Umgang mit der Schrift ergibt sich daraus, dass das Gesamtzeichen durch die Eingabe von genau drei Tastaturzeichen erstellt werden kann, entsprechend der Unterteilung des Gesamtzeichens in drei Teile (siehe Abbildung 12).

Zwei Zeichen müssten jedoch zu den zwölf genannten Zeichen noch hinzukommen. Das wäre zum einen das Zeichen für das *Attribut der Wichtigkeit* und zum anderen ein Zeichen, das den Linkanfang kennzeichnet.

Abbildung 12:  
 Die Tastenfolge aus drei eingegebenen Buchstaben ergibt eine Folge von Zeichen bei Verwendung des Zeichenfonts Zapf Dingbats. Bei entsprechendem Kerning der Einzelzeichen kann man das Verschmelzen zu einer Form erreichen.



Das Zeichen für das *Attribut der Wichtigkeit* sollte als Einzelzeichen umgesetzt werden, um es mit dem Zeichen zur Linkbeschreibung kombinieren zu können. Eine visuelle Metapher für Wichtigkeit lässt sich sicherlich finden, jedoch kommt es bei der Umsetzung des Zeichens vor allem darauf an, dass das Zeichen ein Hervortreten des Links aus dem Gesamttextbild bewirkt. Um den Zusammenhang von Linkbeschreibung und dem zusätzlichen *Attribut der Wichtigkeit* zu manifestieren, sollte das Wichtungszeichen direkt mit dem Zeichen der Linkbeschreibung kombiniert werden. Es sollte dem Zeichen der Linkbeschreibung hintenangestellt werden, mit diesem in seiner Form harmonieren und eine Einheit bilden.

Das Zeichen zur Kennzeichnung des Linkanfangs, im folgenden *Startzeichen* genannt, steht vor dem Wort, das den Link markiert. Es sollte vor allem dann Verwendung finden, wenn der Autor keine zusätzliche Hervorhebung der Links durch einen Farbkontrast oder die Unterstreichung vorgesehen hat. In seiner Gestaltung muss es natürlich ebenso mit dem Zeichen zur Linkbeschreibung harmonieren. Es sollte den Leser zwar auf den Beginn eines Links eindeutig hinweisen, den Lesefluss aber nicht unnötig stoppen.

Um also die Gesamtheit des Beschreibungssystems grafisch umzusetzen, benötigt man genau 14 Zeichen (Teilzeichen).



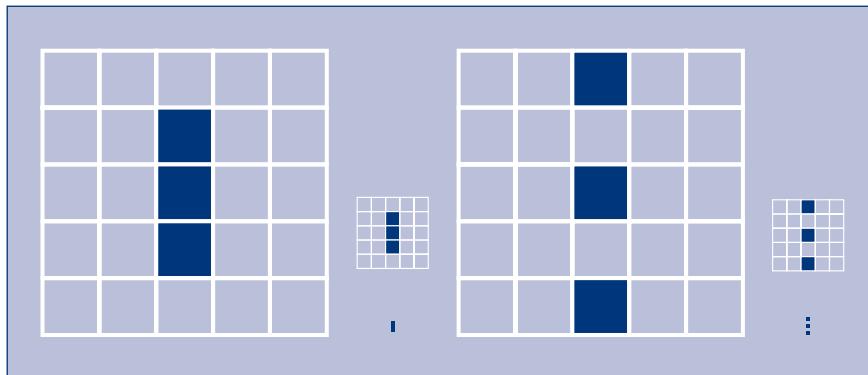
# Die finalen Zeichen

An dieser Stelle sollen nunmehr die finalen Zeichen des Systems zur Linkbeschreibung vorgestellt werden.

Der Gestaltungsrahmen innerhalb eines Rasters von fünf Pixel in der Höhe bietet für den Entwurf der Zeichen nur sehr begrenzte Freiheitsgrade an. Dieser Gestaltungsrahmen sollte aber unbedingt als gegeben betrachtet werden, da er die unterste Grenze der Darstellungsgröße des Zeichens definiert. Alle Überlegungen zur Umsetzung der Zeichen basieren also zunächst auf einer Zeichenhöhe von fünf Pixel. Wie weiterhin erläutert, benötigt eine Darstellung der Matrix im Sinne ihrer Erkennbarkeit einerseits Elemente zur Präsentation der Begrenzung der Matrix, die der Zuordnung der Lage der hervorgehobenen Segmente dienen, als auch die Segmente selbst. Wichtig ist auch eine klare Abtrennung der Segmente voneinander sowie die Abgrenzung der Segmente zu den Begrenzungselementen, um die Erkennbarkeit des Zeichens auch bei sehr kleinen Darstellungsgrößen zu gewährleisten.

Um eine Trennung der Elemente in vertikaler Richtung überhaupt zu ermöglichen, stehen zur Darstellung der Segmente nur drei Pixel in der Höhe zur Verfügung. Der unterste, der mittlere und der oberste Pixel sollen also die Zeilen markieren, in denen die Segmente der Matrix umgesetzt werden. Dadurch ergibt sich eine Höhe der Zeilen von jeweils einem Pixel und eine Trennung der Zeilen von jeweils einem Pixel (siehe Abbildung 13).

Würde man es bei der Umsetzung der Segmente bei einer Größe von einem Pixel belassen, wäre allerdings die Erkennbarkeit des einzelnen Segmentes an der untersten Grenze der Darstellungsgröße schwierig. Eine Vergrößerung der Darstellung der Segmente ist also unumgänglich. Dies ist jedoch nur noch durch die Ausweitung der Segmente in horizontaler Richtung möglich. Wie sich in einigen Entwürfen dazu herausstellte, reicht es, das Segment um einen weiteren Pixel horizontal zu erweitern. Ein Segment kann also minimal durch eine Höhe von einem Pixel und einer



*Abbildung 13:  
Die gleichmäßige Anordnung von drei  
Segmenten mit jeweils einem Pixel Höhe  
auf einem Raster von fünf Pixel Höhe.*

Breite von zwei Pixel dargestellt werden. Die Segmente bedürfen jedoch auch in horizontaler Richtung einer Abtrennung voneinander. Hierbei hat sich bei der Entwurfsarbeit an den Zeichen herausgestellt, dass diese Abtrennung mindestens so groß wie das Segment sein muss, da ansonsten die Segmente die in einer Zeile liegen, zu leicht zu einer Linie verschwimmen und die Erkennbarkeit der einzelnen Segmente dann abnimmt.

Bei der kleinsten Darstellungsgröße des Zeichens ergibt sich also eine Größe für ein Segment, von 1x2 Pixel und eine Gesamtbreite, inklusive der Abstände, von zehn Pixel. Die Darstellung für den Fall, dass innerhalb eines Attributes keine Werte vergeben sind, also kein Segment hervorgehoben wird, muss ebenso viel Raum einnehmen wie das Segment selbst, um die Gesamtbreite des Zeichens beizubehalten. Dies ist unbedingt notwendig, um die Zuordnung der Segmente zu den Attributspalten eindeutig zu gewährleisten.

Es fehlen nunmehr noch die Begrenzungselemente der Matrix. Sie erfüllen vor allem den Zweck, die horizontale Anordnung der Zeilen zu markieren. Weiterhin definieren sie die Breite des Gesamtzeichens. Diese Angabe ist vor allem dann wichtig, wenn Werte, also Segmente eines Attributes, nicht vergeben werden, um die Lage und Zuordnung der anderen Segmente eindeutig zu definieren. Als beste Lösung ergibt sich hierbei die Form einer das Zeichen begrenzenden Klammer. Sie beschreibt einerseits durch ihre Form die Breite und markiert andererseits die Abmaße in der Höhe des Zeichens, also die Lage der obersten und untersten Zeile. Die Lage

der mittleren Zeile muss dann nicht mehr extra ausgewiesen werden, da sie sich durch diese Begrenzung eindeutig genug erkennen lässt. In Abbildung 14 ist das gesamte Zeichen in seiner finalen Ausführung dargestellt.

In Abbildung 14 sind weiterhin alle Zeichen dargestellt, die sich aus diesem Prinzip heraus ableiten lassen und die entsprechend der Definition des Umgangs mit der *Beschreibungsmatrix* zulässig sind.

Der Wunsch der Optimierung der Breite des Zeichens an die Breite der Buchstaben konnte aufgrund der Konstruktion nicht umgesetzt werden. Aber die Breite des Zeichens ist noch akzeptabel im Sinne einer Einordnung in das Textbild wie beispielsweise Abbildung 17 entnommen werden kann.

Neben den Zeichen zur Linkbeschreibung sollen nunmehr noch das *Startzeichen*, sowie das *Wichtungszeichen* entworfen werden.

Das *Startzeichen* nimmt in seiner Form die Höhe des Zeichens zur Linkbeschreibung auf und definiert durch seine Form die Lage der Zeilen. Dadurch kann es zum Zeichen der Linkbeschreibung hinleiten und eine direkte Korrespondenz zu diesem herstellen. Durch die Betonung der Horizontalen stoppt das *Startzeichen* den Lesefluss nur gering. Es bietet aber durch seine Gestaltung genügend Kontrast zu den normalen Buchstaben des Textes, um noch ausreichend aufzufallen (siehe Abbildung 14).

Das *Wichtungszeichen* wurde in Form einer Fläche umgesetzt. Es entspricht der Dimension des *Startzeichens*. Dadurch korrespondiert es optisch mit diesem. Durch den starken Kontrast des *Wichtungszeichens* zu den restlichen Zeichen des Textes soll der Leser in seinem Lesefluss unterbrochen werden. Auf Grund seiner flächigen Gestaltung tritt das *Wichtungszeichen* deutlich aus dem Text hervor und kann dadurch auch beim Überfliegen eines Textes noch schnell erfasst werden (siehe Abbildung 14).

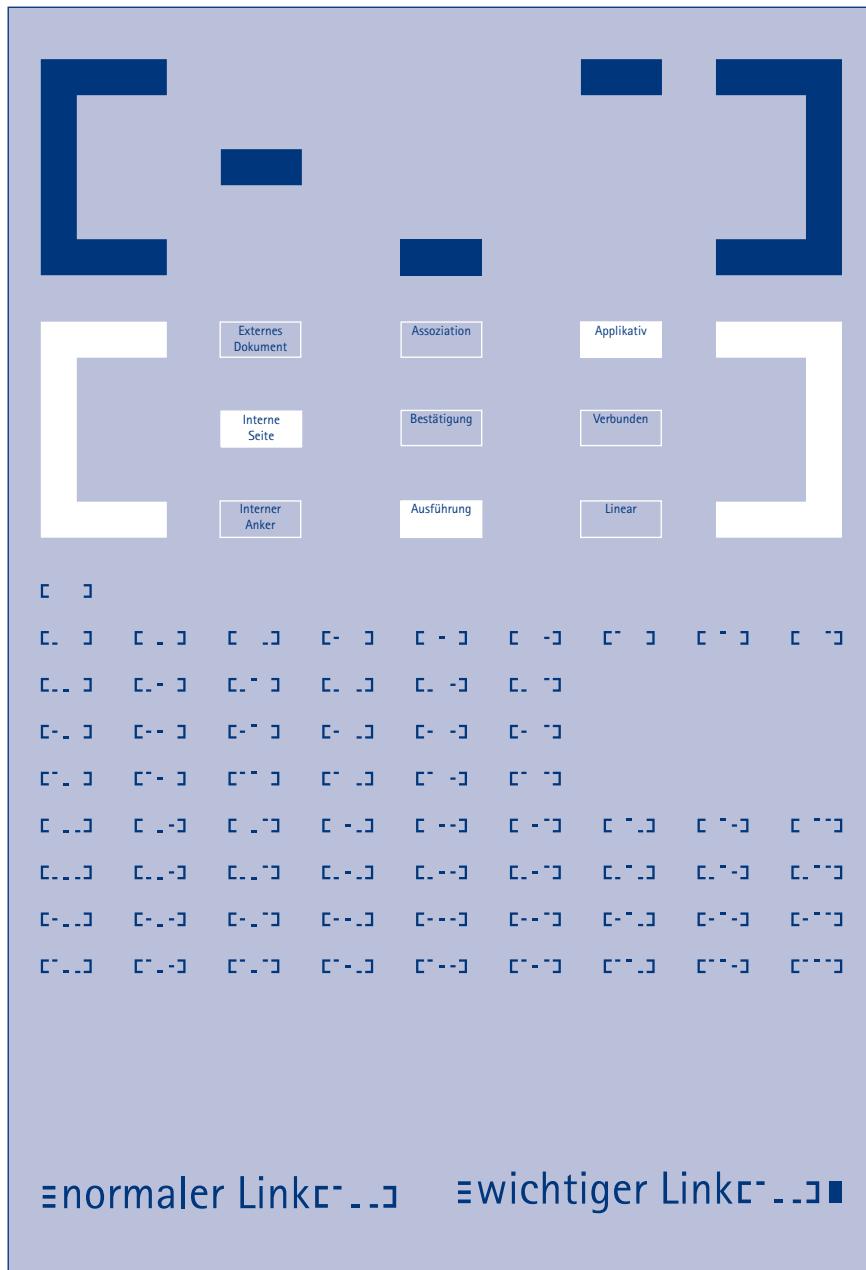


Abbildung 14:

- Darstellung des finalen Zeichens
- Zeichen/Matrix-Referenz
- Darstellung aller möglichen Zeichen
- Start-und Wichtungszeichen in ihrer Anwendung

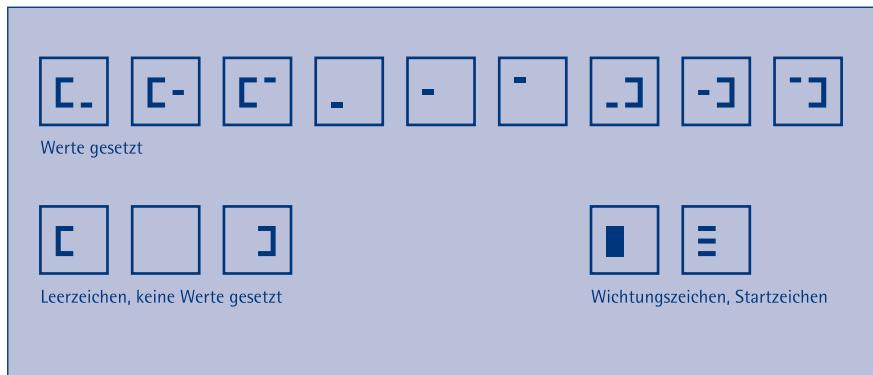


Abbildung 15:

Die Einzelzeichen zur Konstruktion  
des Gesamtzeichens.

Wie bereits erwähnt, soll das Zeichen zur Linkbeschreibung in seine Teile zerlegt werden, um quasi als „Baukasten“ für die Konstruktion des Gesamtzeichens dem Anwender zur Verfügung zu stehen. Das entworfene Gesamtzeichen wird daher, wie in Abbildung 15 gezeigt, in seine Einzelzeichen unterteilt. Durch die Hinzunahme der Begrenzungselemente zur jeweils ersten und letzten Segmentspalte erreicht man dabei, dass eine Fehlverwendung der Einzelzeichen sofort sichtbar wird. Der Anwender hat dadurch eine Kontrolle über die Richtigkeit der erstellten Zeichen, zumindest in der Frage der Vergabe von jeweils nur einem Wert innerhalb einer Attributspalte.

Das *Startzeichen* und das *Wichtungszeichen* sind nicht zerlegt, da sie als Einzelzeichen dem Link bzw. dem Linkzeichen angefügt werden.

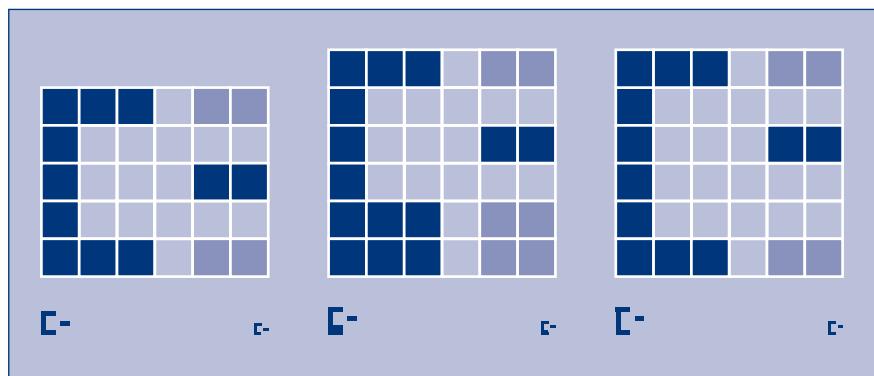


# Der Link Description Font

Nachdem die Zeichen des grafischen Systems zur Beschreibung von Links entworfen wurden, sollen diese Zeichen nunmehr in Form einer Schrift umgesetzt werden.

Ein entscheidendes Kriterium für die Darstellung von Schrift auf Monitoren ist die Übertragung der Zeichen auf die der Größe der Schrift entsprechenden Pixelwerte. Dieser Prozess wird durch das sogenannte Hinting der Schrift gesteuert. Durch die Hinting-Festlegungen einer Schrift kann deren Adaption auf verschiedene Pixelgrößen beeinflusst werden. Die Bedeutung dieser Steuerung wird zum Beispiel bei der Übertragung der entworfenen Zeichen auf ein Sechs-Pixel-Raster deutlich. Das Sechs-Pixel-Raster ergibt sich bei der Skalierung der Zeichen auf die nächstgrößere Schriftgröße nach der definierten kleinsten Schriftgröße. Durch eine ungesteuerte Übertragung eines Zeichens auf dieses Raster kann es zu Verzerrungen des Zeichens in den Segmenten kommen, die eine Erkennbarkeit des Zeichens erschweren oder gänzlich unmöglich machen (siehe Abbildung 16). Sollten derartige Verzerrungen bei den Segmenten der Einzelzeichen noch dazu unregelmäßig auftreten, wäre die Interpretation des Gesamtzeichens schwer gestört.

*Abbildung 16:  
Adaption des fünf Pixel hohen  
Grundzeichens auf eine  
x-Höhe von sechs Pixel.  
Entweder kommt es zur Verzerrung eines  
Segmentes oder zur Verzerrung  
des Abstandes zwischen den Zeilen.  
Die Lage der anderen Segmente  
ist angedeutet.*



Aus diesem Grund wurden für die Teilzeichen in der Schrift die Hinting-Festlegungen manuell erstellt. Dadurch ergibt sich eine Verzerrung der Teilzeichen bei ungeeigneten Rasterwerten (wie zum Beispiel dem Sechs-Pixel-Raster) nur innerhalb der Abstände der Zeilen zueinander und nicht in der Größendarstellung der Segmente selbst. Das heißt, dass es im Fall einer für die Zeichen eher ungeeigneten Schriftgröße maximal zu einer Verschiebung der Abstände der Zeilen kommt. Da die Hinting-Festlegungen für jedes Teilzeichen von Hand erstellt wurden, konnte gewährleistet werden, dass der Verzerrungseffekt auch bei allen Zeichen gleich ist, so dass es trotz einer Verzerrung der Einzelzeichen zu einer homogenen Darstellung des Gesamtzeichens kommt. Ebenso wie die Teilzeichen werden auch die Zeichen für die Wichtung und den Linkbeginn verzerrt. Dies führt beispielsweise dazu, dass man anhand der Darstellung des Startzeichens sofort erkennen kann, ob eine Schriftgröße für die Darstellung der Zeichen geeignet ist oder in welcher Art die Abstände der Zeilen verzerrt werden (siehe Abbildung 17).

*Der in den Beispielen in Abbildung 17 verwendete Text entspricht dem der Abbildung 1. Die in der Abbildung verwendeten Zeichen zur Beschreibung der Links sind in ihrer Bedeutung frei erfunden und sollen nur das Zusammenspiel verschiedener Zeichen illustrieren.*

Die Schrift wurde als TrueType Schrift umgesetzt, weil sich in diesem Format das Hinting der Zeichen am besten steuern lässt. Sie wurde für die Verwendung auf Apple und Windows Computern angelegt. Das heißt, es wurde für jedes der beiden Betriebssysteme ein entsprechender Font erzeugt.

Die Schrift wurde *Link Description Font*, kurz *Linkfont*, benannt.

Die genaue typographische Konstruktion des *Link Description Fonts* kann den Tafeln zu den Einzelzeichen am Ende der Arbeit entnommen werden.

Ein weiterer Aspekt bei der Umsetzung der Zeichen in Form einer Schrift, ist die Frage nach der Zuordnung der Zeichen zu einer Tastaturbelegung.

Die Tastatur eines Computers enthält die Zeichen zur Linkbeschreibung nicht. Daher stellt sich die Frage, welcher der Tasten man in der Schrift die 14 Zeichen zuordnet. Nahe liegend wäre zunächst, nach einer Wert-Buchstaben-Referenz zu suchen.

## **≡Hyperlinks**

In addition to general purpose elements such as paragraphs and lists, HTML documents can express hyperlinks. An HTML `<a href=>` element allows the user to navigate these hyperlinks.

## **≡Hyperlinks**

In addition to general purpose elements such as paragraphs and lists, HTML documents can express hyperlinks. An HTML `<a href=>` element allows the user to navigate these hyperlinks.

## **≡Hyperlinks**

In addition to general purpose elements such as paragraphs and lists, HTML documents can express hyperlinks. An HTML `<a href=>` element allows the user to navigate these hyperlinks.

## **≡Hyperlinks**

In addition to general purpose elements such as paragraphs and lists, HTML documents can express hyperlinks. An HTML `<a href=>` element allows the user to navigate these hyperlinks.

## **≡Hyperlinks**

In addition to general purpose elements such as paragraphs and lists, HTML documents can express hyperlinks. An HTML `<a href=>` element allows the user to navigate these hyperlinks.

**≡user agent**

**≡user agent**

Abbildung 17:  
Anwendung des Linkfonts in einem Text.  
(vergrößerte Screenshots)

Darstellung verschiedener  
Skalierungsgrade des Textes

Darstellung der Referenz von  
Startzeichen und Beschreibungszeichen  
bei ungeeigneter Schriftgröße

Aber dies ist schon bei der deutschen Beschreibung der Werte nicht möglich und verbietet sich unter Berücksichtigung verschiedener Sprachräume eigentlich sofort.

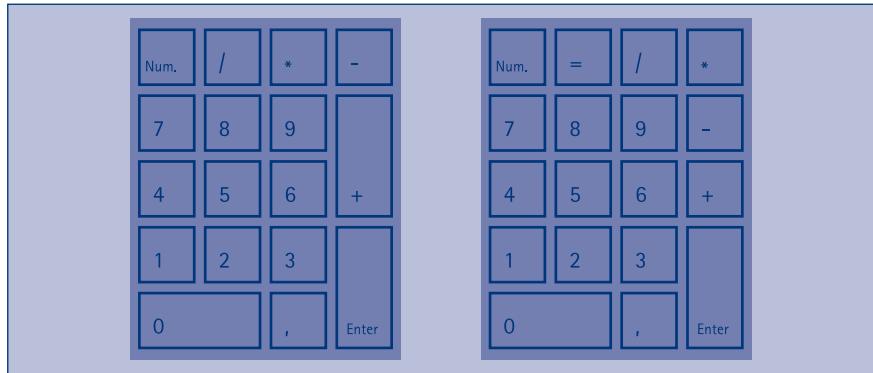
Da das Zeichensystem eine 3er-Matrix widerspiegelt, könnte diese auch der Schlüssel zu einer sinnvollen Übertragung der Zeichen auf eine Tastaturbelegung sein. Verschiedene Sprachräume verwenden jedoch auch verschiedene Tastaturbelegungen, so dass eine Auswahl der entsprechenden Tasten dies berücksichtigen müsste. Hinzu kommt, dass die normalen Tasten der Tastatur nicht senkrecht übereinander angeordnet sind. Die Matrix könnte also nur verzerrt dargestellt werden.

Ein spezieller Bereich der meisten Tastaturen ist jedoch sowohl senkrecht angeordnet als auch in Form einer 3er-Matrix realisiert. Dies sind die Zahlen der numerischen Tastatur. Dieser Teil der Tastatur ist sowohl international als auch bei verschiedenen Betriebssystemen gleich aufgebaut. Unterschiede existieren lediglich in der Anordnung der Operationszeichen beispielsweise zwischen einem Apple-Computer und einem Windows-Computer. Die Anordnung der Zahlen jedoch ist immer gleich und wäre somit die optimale Referenz zur 3er-Matrix des Beschreibungssystems. Man hat damit neben der Vereinfachung des Umgangs mit den Zeichen auf grafischer Ebene auch noch eine zusätzliche Vereinfachung bezüglich der Eingabe der Zeichen durch den Anwender gefunden (siehe Abbildung 18).

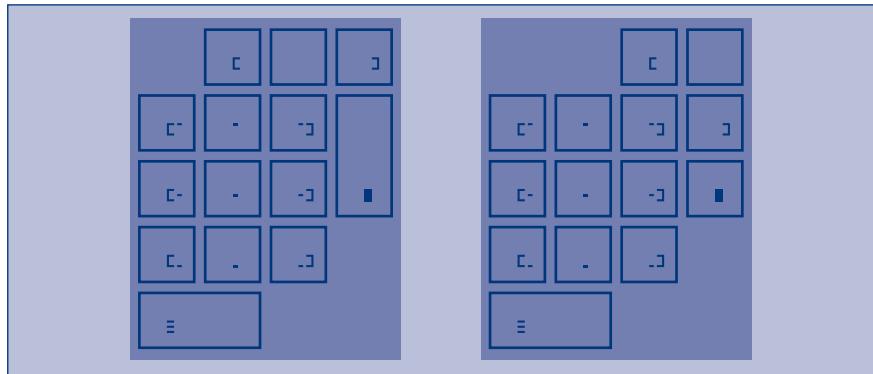
Da alle Zahlen und Operationszeichen der numerischen Tastatur nochmals in der normalen Tastenbelegung vorhanden sind, bedeutet eine Belegung dieser Tastatur mit den Zeichen ebenfalls eine Belegung der entsprechenden Werte mit den Zeichen innerhalb der normalen Tastatur. Mit anderen Worten: belegt man die Taste „1“ mit einem Zeichen, ist dieses Zeichen sowohl auf der numerischen Tastatur, als auch auf der normalen Tastatur vorhanden. Damit ist eine Verwendung der Zeichen auch auf Tastaturen, die keine zusätzliche numerische Tastatur besitzen, möglich. Hier fehlt nur die vereinfachende optische Referenz zur Matrix.

Die Zuordnung der Einzelzeichen auf den Zahlenblock der numerischen Tastatur entspricht genau der Lage der Segmente in der Matrix. Da auch die Zeichen für die Nichtbesetzung eines Wertes als Einzelzeichen adaptiert werden müssen, kommt nur eine Belegung der Tasten für die mathematischen Operationszeichen für diese

*Abbildung 18:  
Tastaturbelegung  
der numerischen Tastaturen von  
Windows- (links) und  
Apple-Computern (rechts).*



*Abbildung 19:  
Tastaturbelegung  
der numerischen Tastatur  
mit den Zeichen des Linkfonts  
bei Windows- (links) und  
Apple-Computern (rechts).*



Zeichen in Frage. Da sich hier jedoch ein Unterschied im Aufbau der numerischen Tastaturen beispielsweise zwischen Apple- und Windows-Tastaturen ergibt, musste für diesen Fall eine Festlegung getroffen werden. Die Belegung der Operationszeichtasten wurde in Anbetracht der Mehrzahl derartiger Tastaturen für die Windows-Tastatur optimiert (siehe Abbildung 19).



# Festlegungen zum Umgang mit dem Link Description Font

Der Umgang mit dem *Link Description Font*, bedarf einiger Regeln, die an diese Stelle zusammengefasst werden sollen:

- Der *Linkfont* darf frei verwendet werden.
- Die Zeichen des *Linkfonts* dürfen nur entsprechend ihrer vorgesehenen Funktion eingesetzt werden.
- Der *Linkfont* darf in seiner Form nicht verändert werden.
- Bei der Verwendung des *Linkfonts* auf einer Hypertextseite ist dies zu kennzeichnen. Um dem Leser die Möglichkeit der Verwendung des *Linkfonts* zu geben, sollte ein Hinweis zur Bezugsmöglichkeit des Fonts gegeben werden.
- Als Hilfestellung für die Anwendung kann auf die Webseite zum *Linkfont* verwiesen werden oder dort beziehbare Darstellungen und Beschreibungen zum *Linkfont* in einer eigenen Hilfe zum Dokument verwendet werden.
- Sollte der Autor eines Hypertextes auf eine zusätzliche Auszeichnung des Links verzichten, ist unbedingt das *Startzeichen* zur Kennzeichnung des Linkanfangs zu verwenden.
- Zwischen dem Wort, das den Link beschreibt, und den Zeichen des *Linkfonts* dürfen keine Leerzeichen verwendet werden. Dies gilt auch für die Einzelzeichen.



# *Zum Schluß heißt nicht am Ende.*

In diesem Kapitel wird die Arbeit zusammengefasst und kritisch betrachtet. Außerdem wird ein Ausblick über die weitere Entwicklung des Linkfonts gegeben.



# Zusammenfassung und kritische Betrachtung

Ziel dieser Arbeit war der Entwurf und die Umsetzung eines grafischen Systems zur Beschreibung von Links im World Wide Web.

Zu diesem Zweck wurde zunächst ein System zur Beschreibung der Eigenschaften eines Links, unter besonderer Berücksichtigung der Aspekte einer inhaltlichen Relevanz von Quell- und Zieltypen entworfen. Dieses System vereint Attribute von Link-eigenschaften und Werte dieser Attribute. Die Werte wurden in einer Matrix, der *Beschreibungsmatrix*, zusammengefasst.

Des weiteren wurde nach einer grafischen Lösung zur Visualisierung dieses Systems gesucht. Als Grundlage für eine grafische Umsetzung wurde die *Beschreibungsmatrix* bestimmt. Auf der Darstellung einer Matrix beruhende grafische Zeichen wurden entwickelt. Dabei lag der Schwerpunkt auf der Entwicklung von Zeichen, die auch bei einer sehr kleinen Darstellungsgröße noch deutlich erkennbar sind und sich somit harmonisch in das Textbild eines digital präsentierten Lesetextes integrieren.

Die entwickelten Zeichen wurden in eine Art Baukastensystem zerlegt und in Form einer Schrift zusammengefasst. Diese Schrift wurde zum Zweck einer eindeutigen Darstellung der Zeichen, in allen möglichen Skalierungsgraden, optimiert. Als Untergrenze der Darstellbarkeit wurde dazu eine Schriftgröße von neun Pixel ermittelt und festgelegt. Die Schrift wurde für Apple- und Windows-Computer im TrueType-Format erstellt. Als Hilfe für die Verwendung der Schrift wurde eine Tastaturoberfläche durch die Zeichen der Schrift gewählt, die dem Anwender die Benutzung der Schrift erleichtert.

Das entwickelte System wurde für die primäre Anwendung in komplexen wissenschaftlich-technischen Hypertexten entworfen. Die Nützlichkeit eines derartigen Systems besteht für den Leser einerseits in einer Unterstützung bei der Konstruk-

tion seines Gedankenmodells und andererseits in der verbesserten Orientierung in komplexen Hypertexten. Der Autor wiederum hat durch das System die Möglichkeit, die Benutzbarkeit eines Hypertextes zu verbessern und Links im Sinne ihrer Relevanz für den Inhalt eines Hypertextdokumentes zu beschreiben.

Das System wurde unter dem Gesichtspunkt der Entwicklung eines Standards zur Linkbeschreibung entworfen. Um diesem Anspruch jedoch gerecht zu werden, bedarf es einerseits noch einer Diskussion des Entwurfs in entsprechenden Fachkreisen. Andererseits sollte das System auch noch in seiner Anwendung und Benutzbarkeit auf Autoren- und Leserseite durch Usability-Tests geprüft werden. Auch die technische Realisierung wäre noch nach dem beschriebenen Optimum umzusetzen.

Diese Aspekte konnten in der vorliegenden Arbeit naturgemäß nicht realisiert werden. Es handelt sich trotz der Ausarbeitung der Thematik vom Ansatz bis zur beschriebenen technischen Umsetzung nur um einen Entwurf, obwohl das System in seiner vorliegenden Form von jedem Autor eines Hypertextes nunmehr vollständig anwendbar ist.

Die vorliegende Arbeit stellt somit eine in sich logische und im Sinne der Umsetzbarkeit optimierte Konstruktion dar. Sie vereinigt Aspekte des Informations- und Interfacedesigns sowie konkrete Aspekte der Typographie im digitalen Umfeld.

## Ausblick

Die Relevanz eines Systems, wie das in dieser Arbeit entworfene, bemisst sich zum großen Teil in seiner Bekanntheit und Akzeptanz im World Wide Web.

Das System soll auch nach dieser Arbeit weiterentwickelt werden, jedoch zunächst zur Diskussion gestellt werden.

Ein nächster entscheidender Schritt wird jedoch zunächst die Übertragung und Adaption der Inhalte dieser Arbeit in die englische Sprache sein. Nur dadurch bekommt das System zur Beschreibung von Links eine wirkliche Relevanz innerhalb des WWW.

Die Webseite zum *Link Description Font* soll daher auch nach dieser Arbeit noch weiter ausgebaut werden.



# *Teile des Ganzen.*

In diesem Teil der Arbeit befinden sich Tafeln mit den typografischen Konstruktionsdarstellungen der Einzelzeichen, die Litraturliste, sowie Hinweise auf den Inhalt der beigelegten CD und der Webseite zur Arbeit.



## Tafeln der Einzelzeichen

Im Folgenden sind die Einzelzeichen auf Tafeln einzeln dargestellt. Den Tafeln lässt sich die konkrete typografische Konstruktion der Zeichen, sowohl in ihrer Anpassung auf das Raster, als auch in ihrer Hintig-Steuerung, entnehmen. Weitere Angaben betreffen die Maße der Schrift.

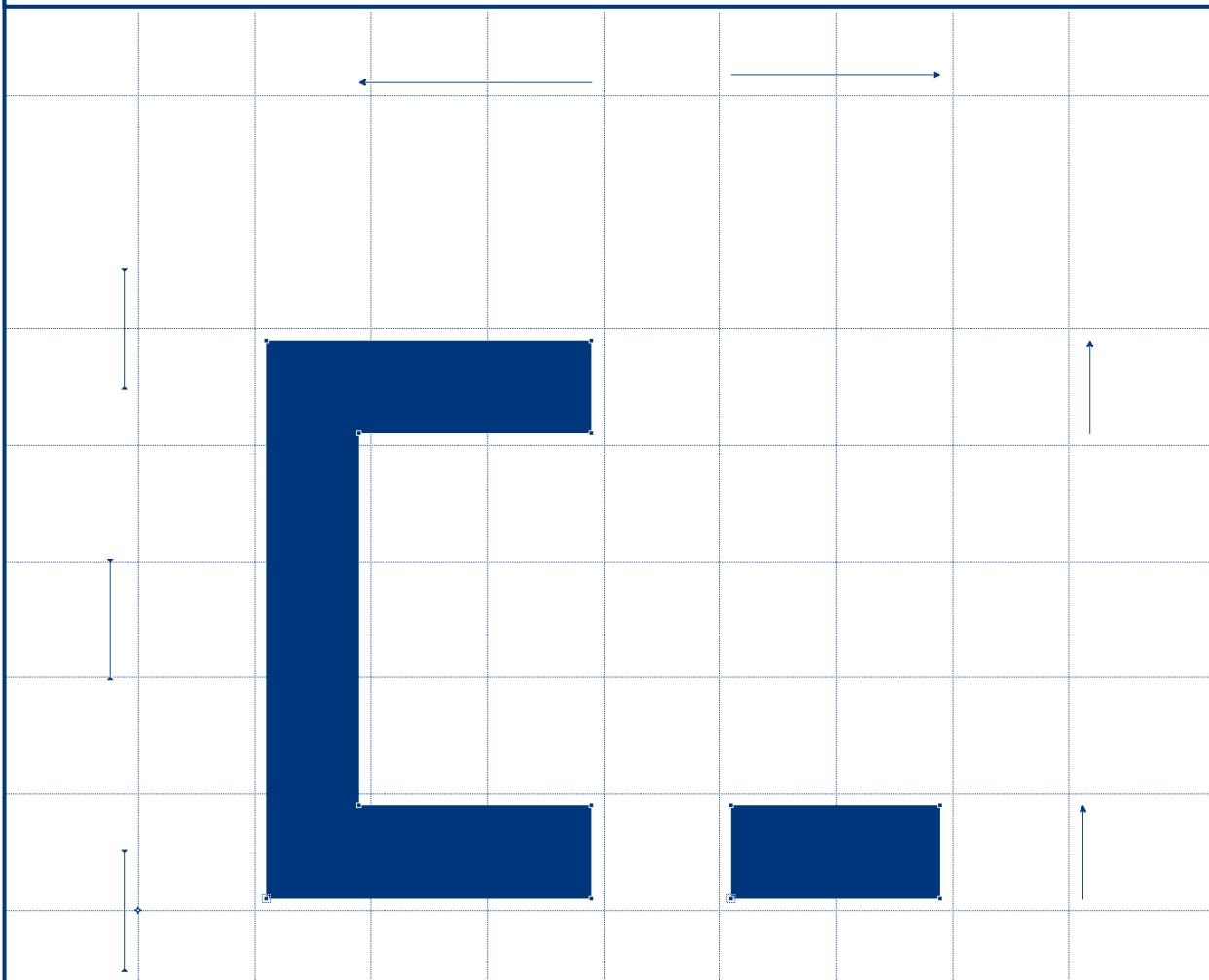
## Zeichen: Interner Anker

Link Description Font © Robert Zinner 2002

Character on keyboard: 1

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

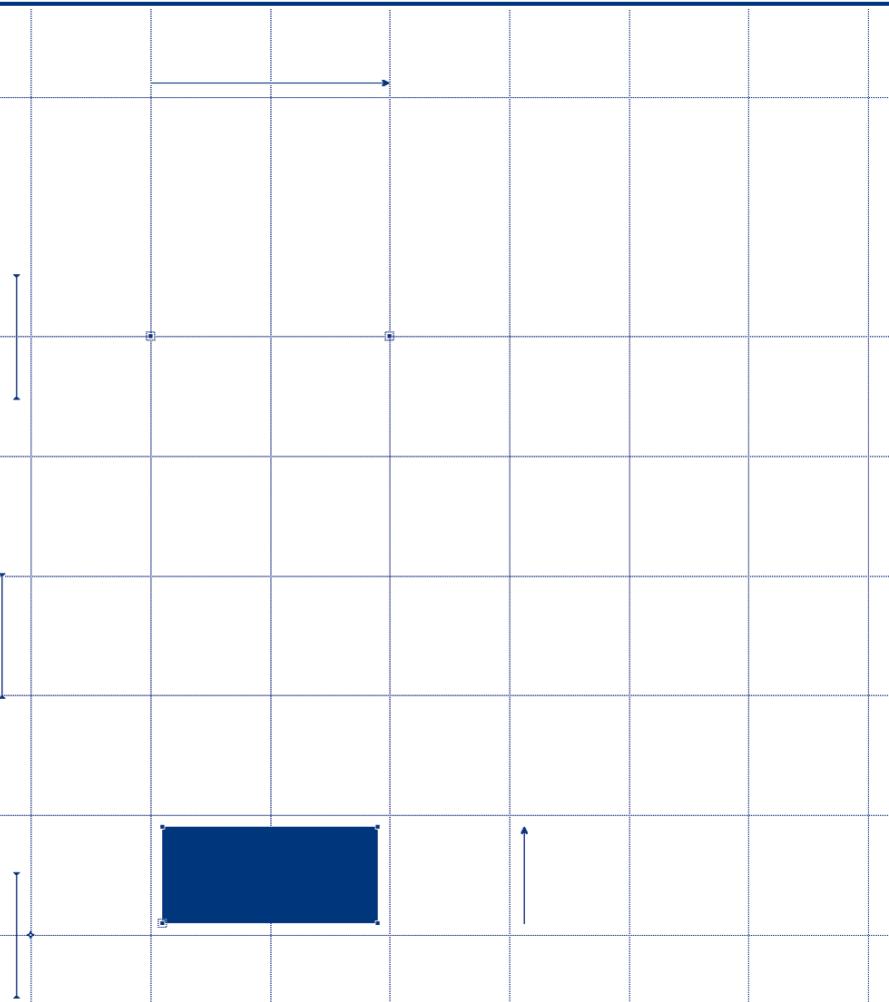
## Zeichen: Ausführung

Link Description Font © Robert Zinner 2002

Character on keyboard: 2

Metrics: width 400 · left 100 · right 100 · kerning 0

Height at 9 pix fontsize: 5 pix · width 4 pix



FONTPNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

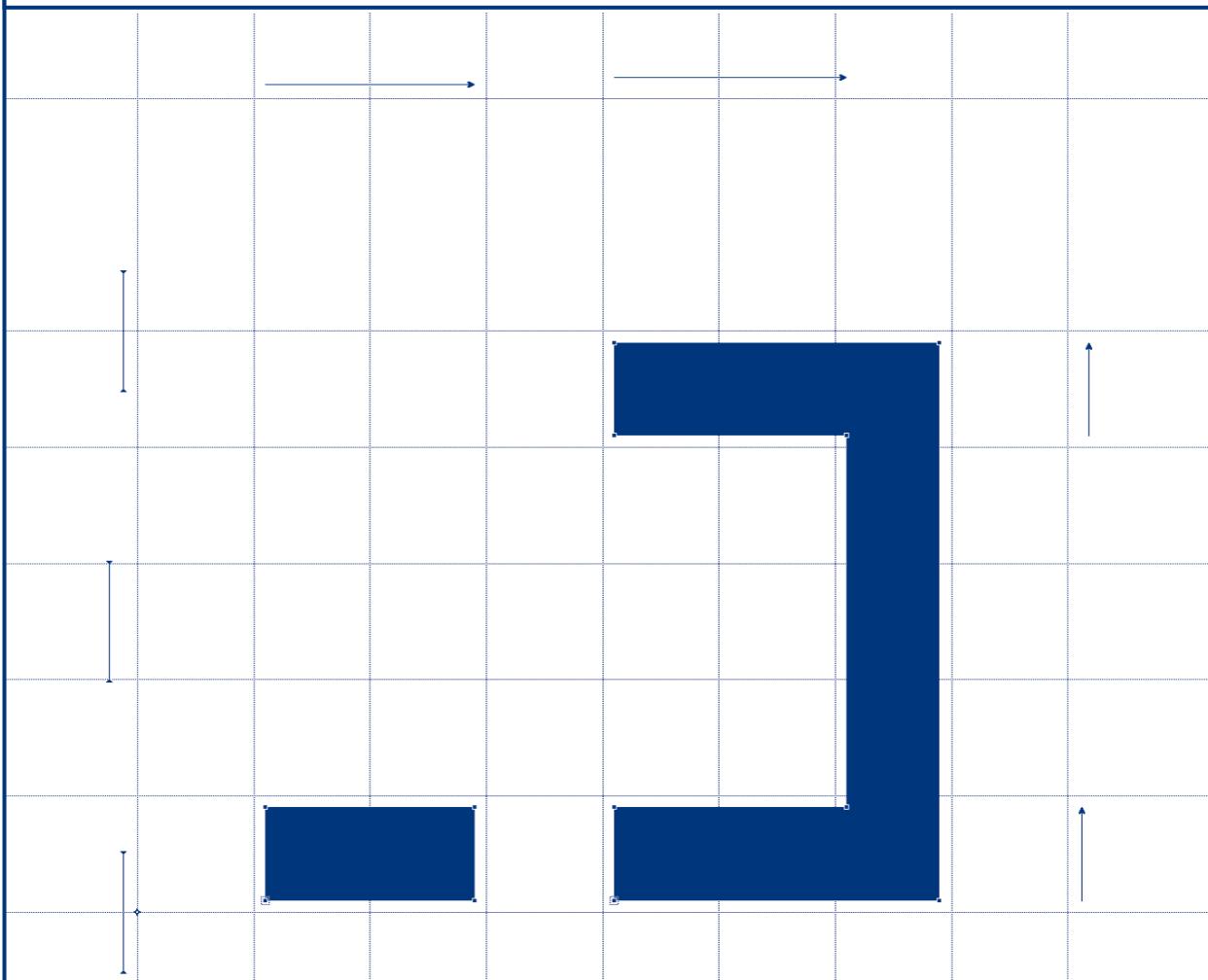
## Zeichen: Linear

Link Description Font © Robert Zinner 2002

Character on keyboard: 3

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTNAME: linkfont.suit

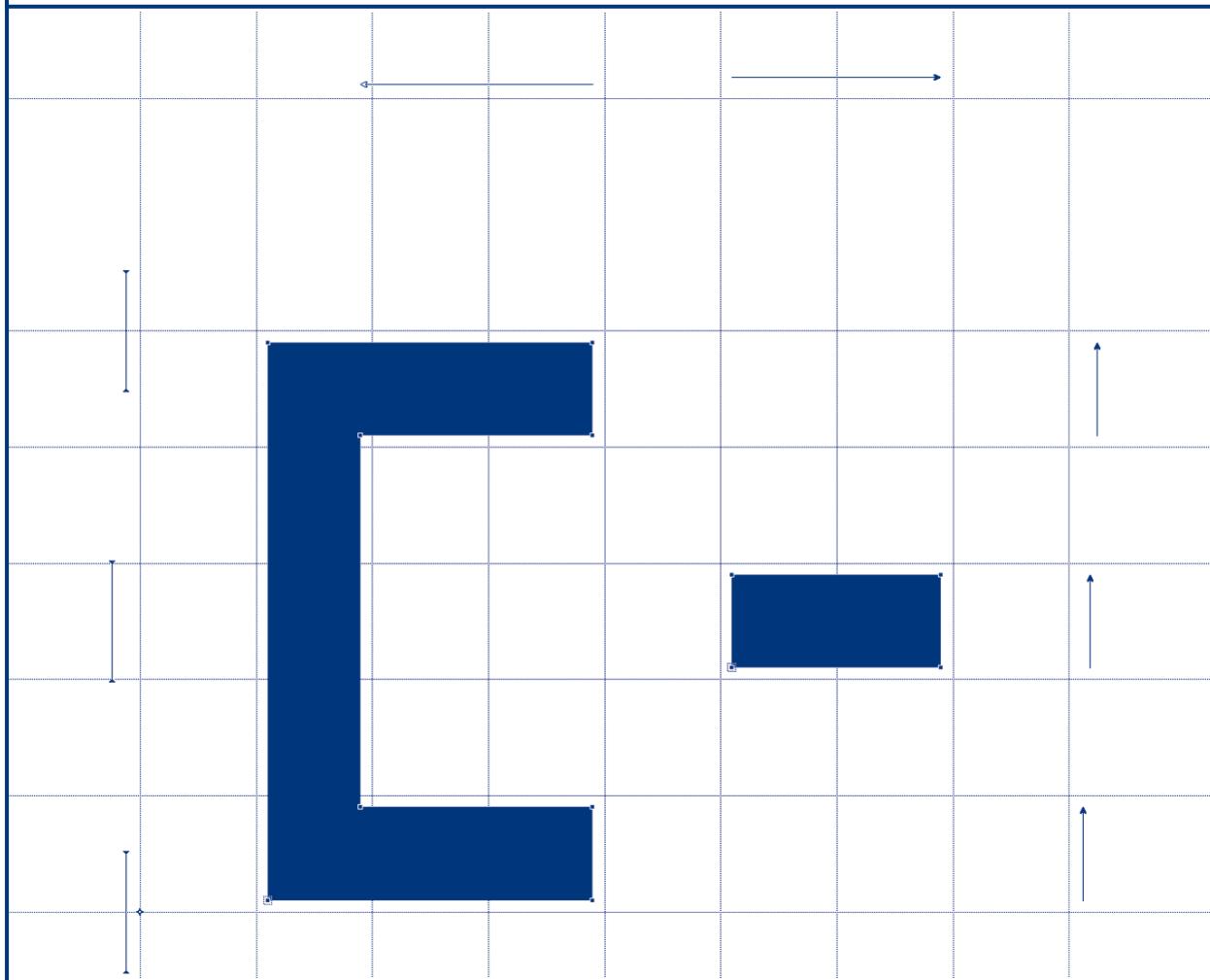
FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

Character on keyboard: 4

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTPNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

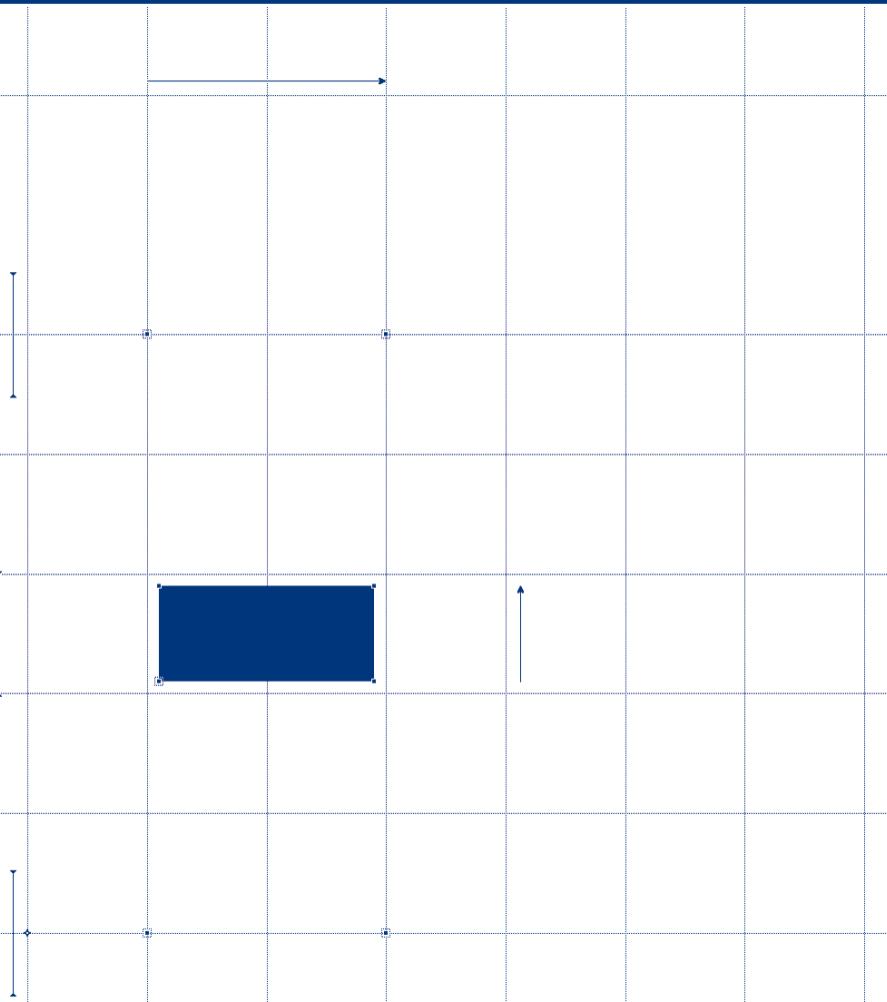
## Zeichen: Bestätigung

Link Description Font © Robert Zinner 2002

Character on keyboard: 5

Metrics: width 400 · left 100 · right 100 · kerning 0

Height at 9 pix fontsize: 5 pix · width 4 pix



FONTNAME: linkfont.suit

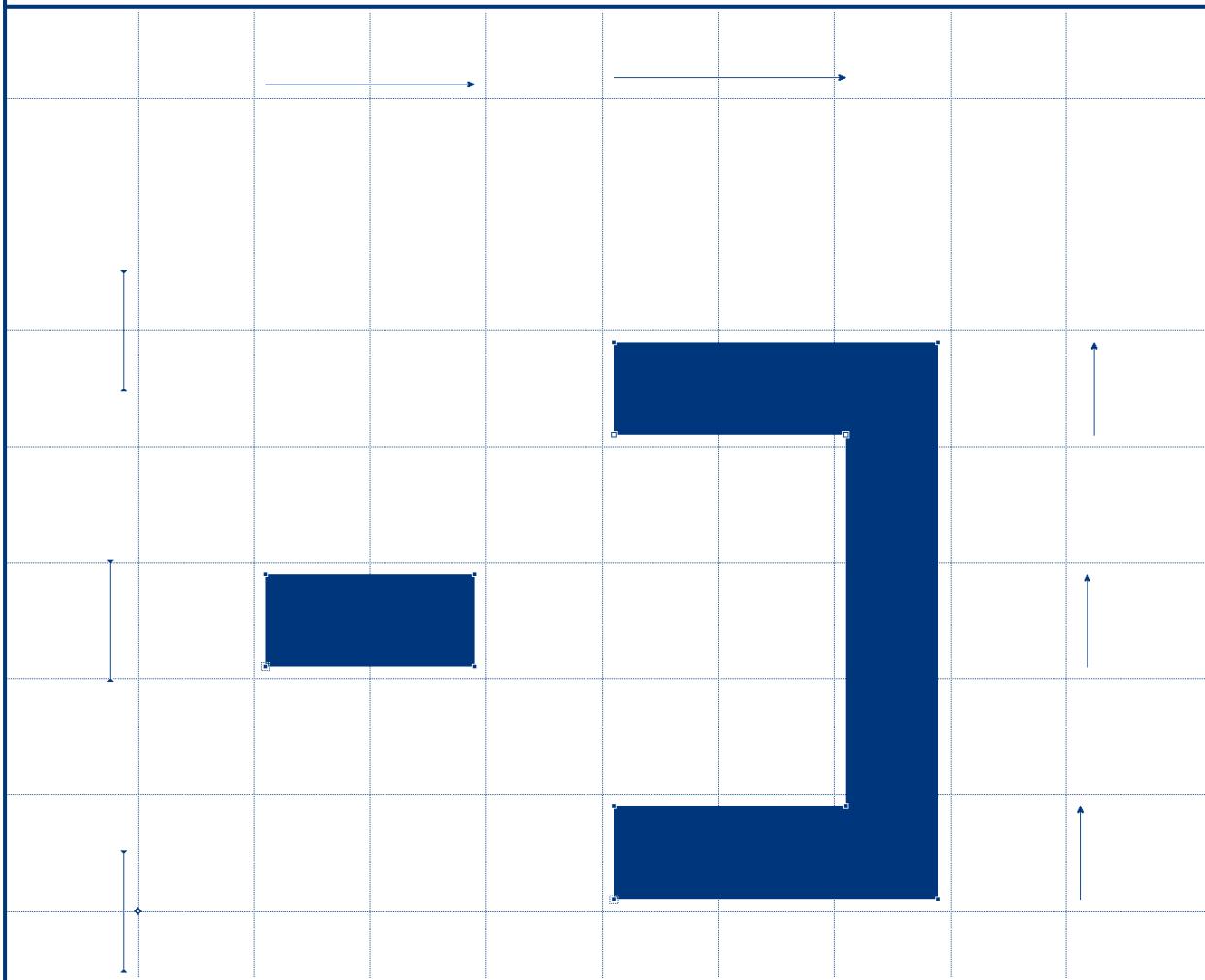
FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

Character on keyboard: 6

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

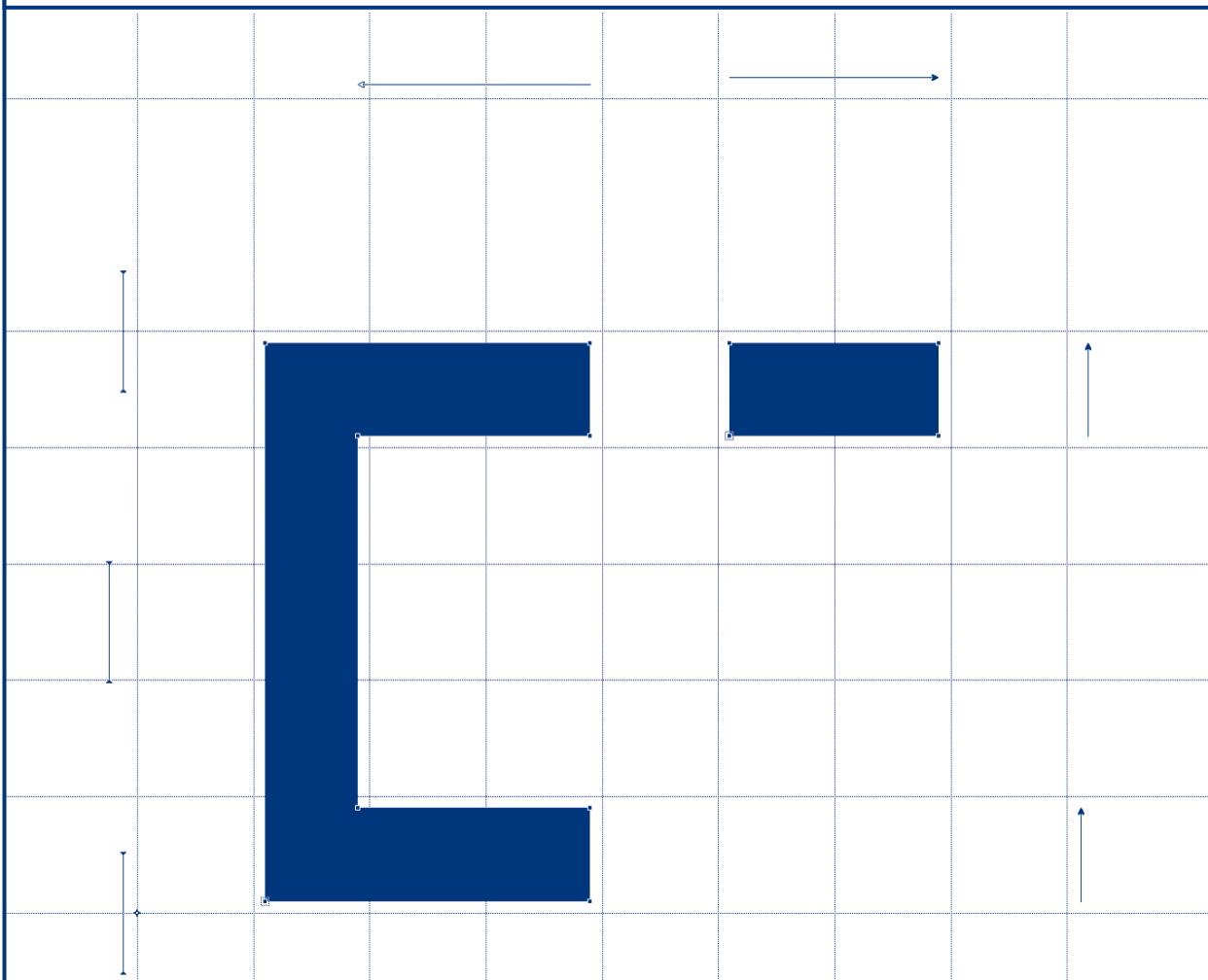
## Zeichen: Externes Dokument

Link Description Font © Robert Zinner 2002

Character on keyboard: 7

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

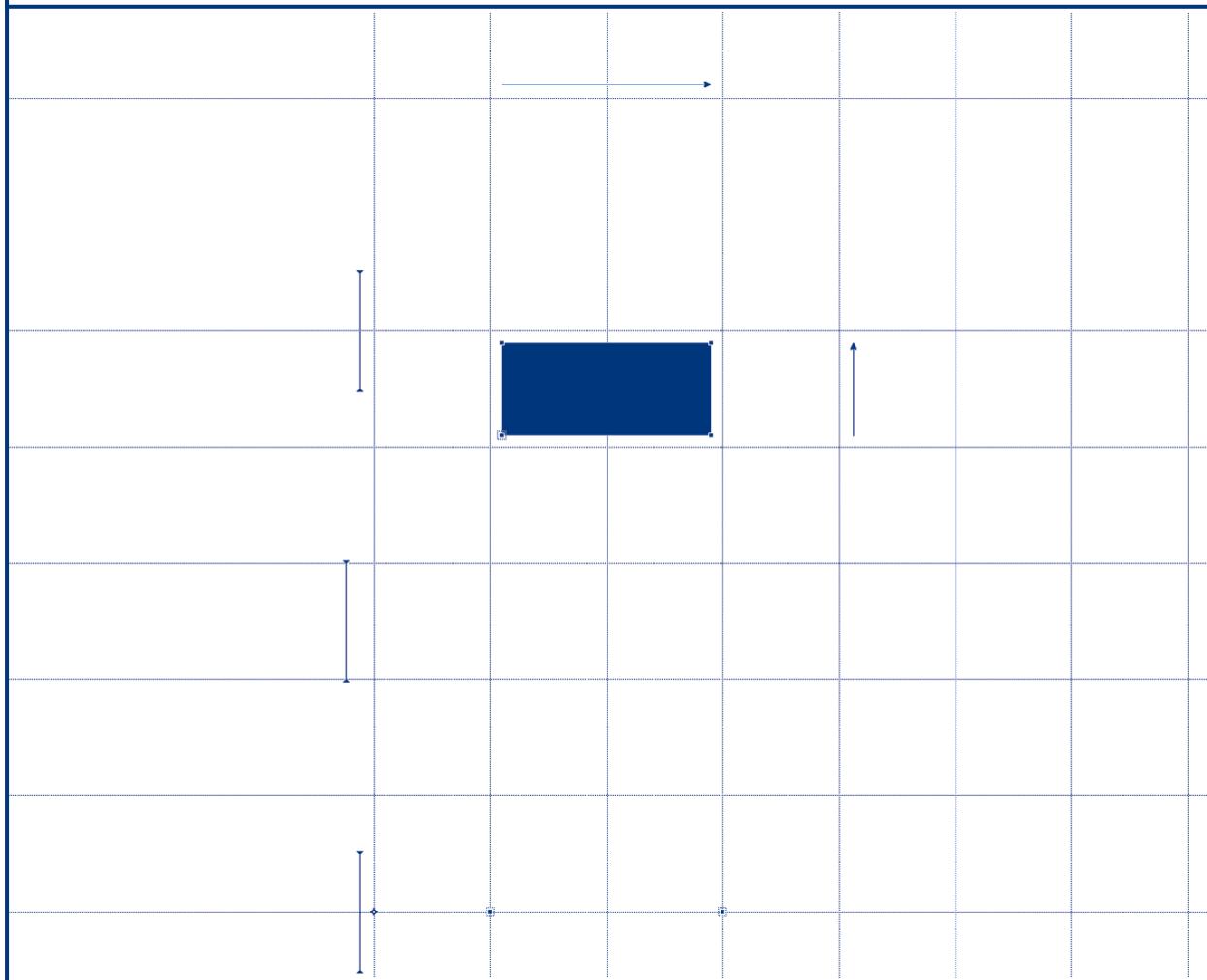
## Zeichen: Assoziation

Link Description Font © Robert Zinner 2002

Character on keyboard: 8

Metrics: width 400 · left 100 · right 100 · kerning 0

Height at 9 pix fontsize: 5 pix · width 4 pix



FONTPNAME: linkfont.suit

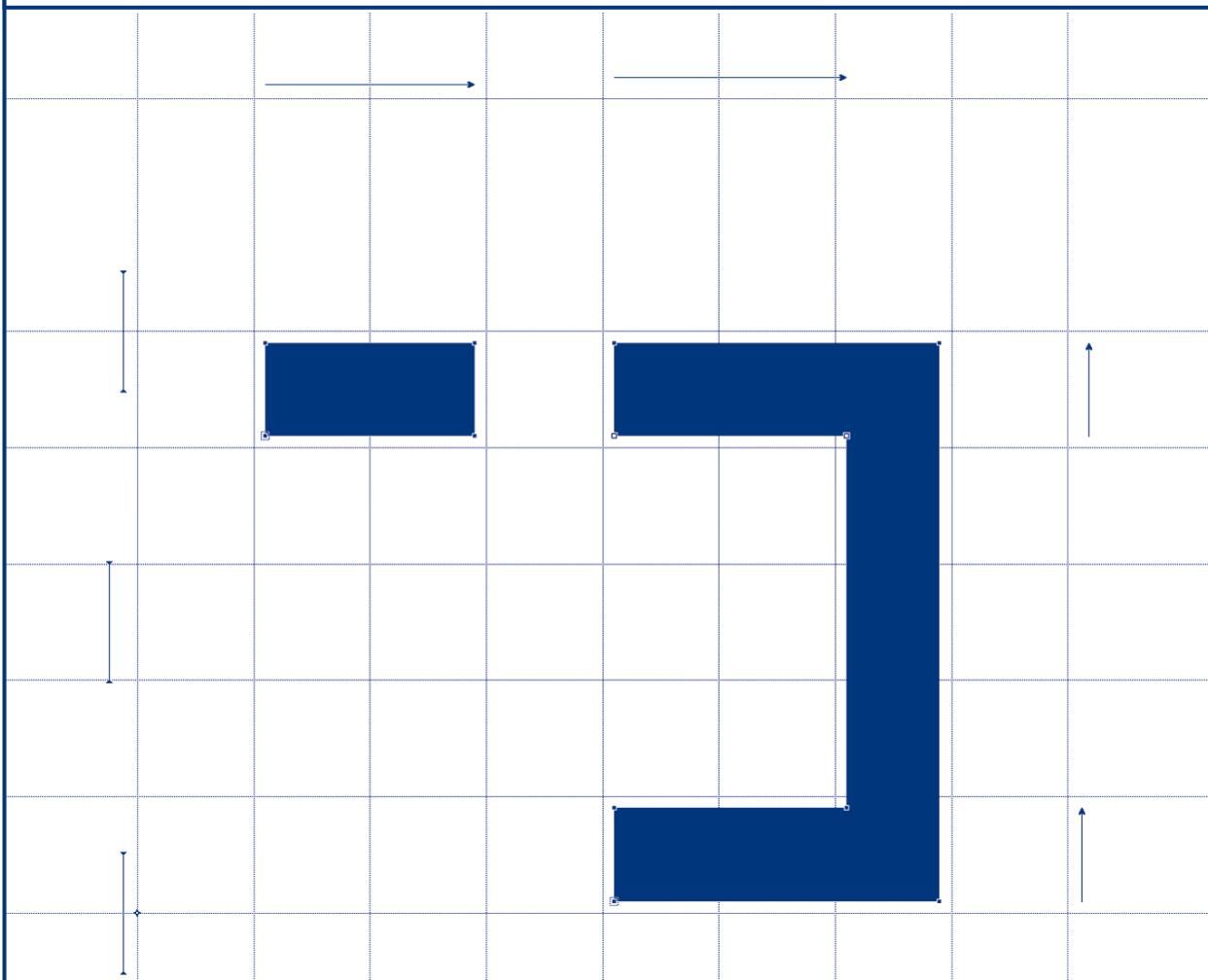
FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

Character on keyboard: 9

Metrics: width 800 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

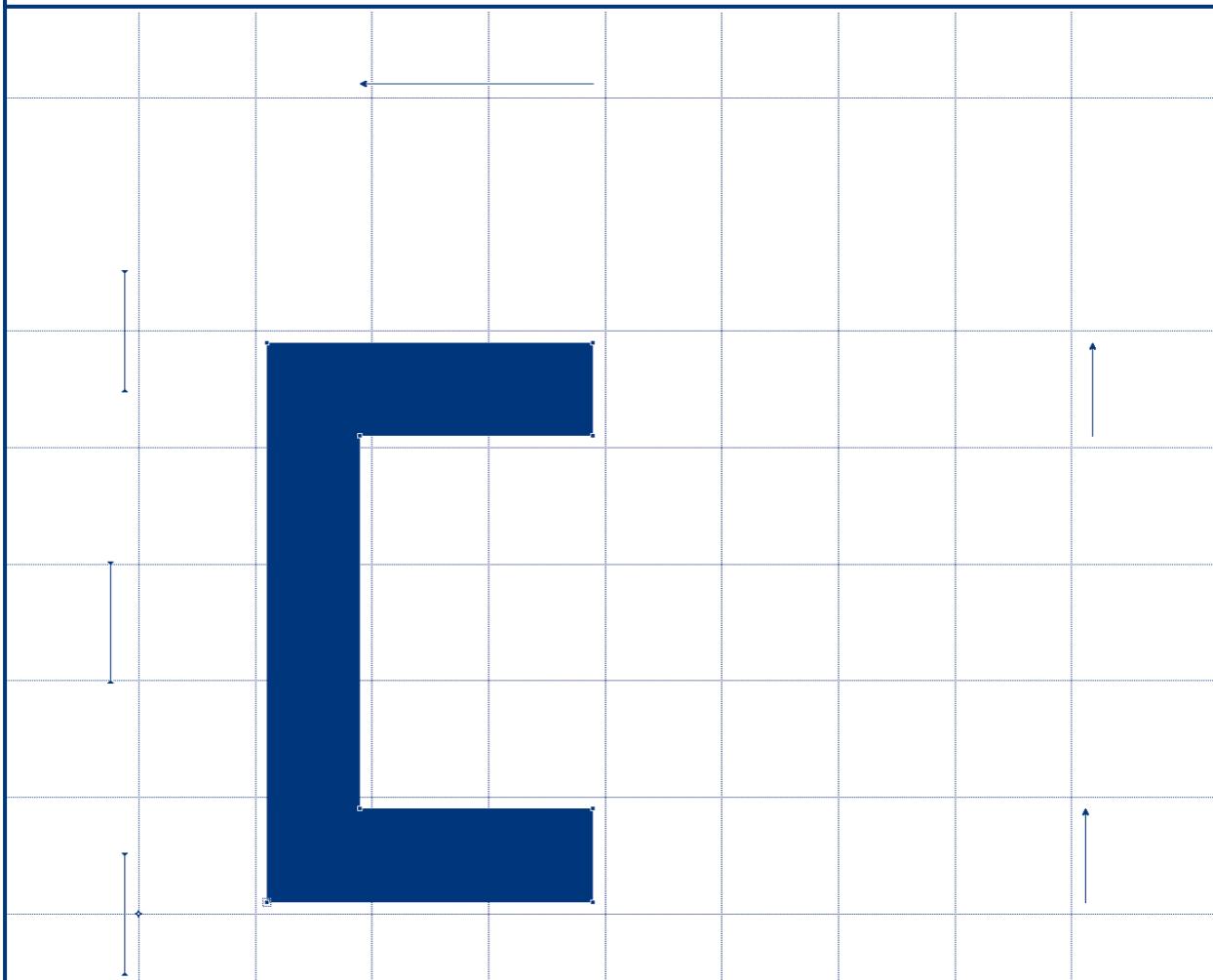
## Zeichen: Leerzeichen Ort

Link Description Font © Robert Zinner 2002

Character on keyboard: /

Metrics: width 800 · left 110 · right 410 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTPNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

## Zeichen: Leerzeichen Inhalt

Link Description Font © Robert Zinner 2002

Character on keyboard: \*

Metrics: width 400 · left 100 · right 100 · kerning 0

Height at 9 pix fontsize: 5 pix · width 4 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

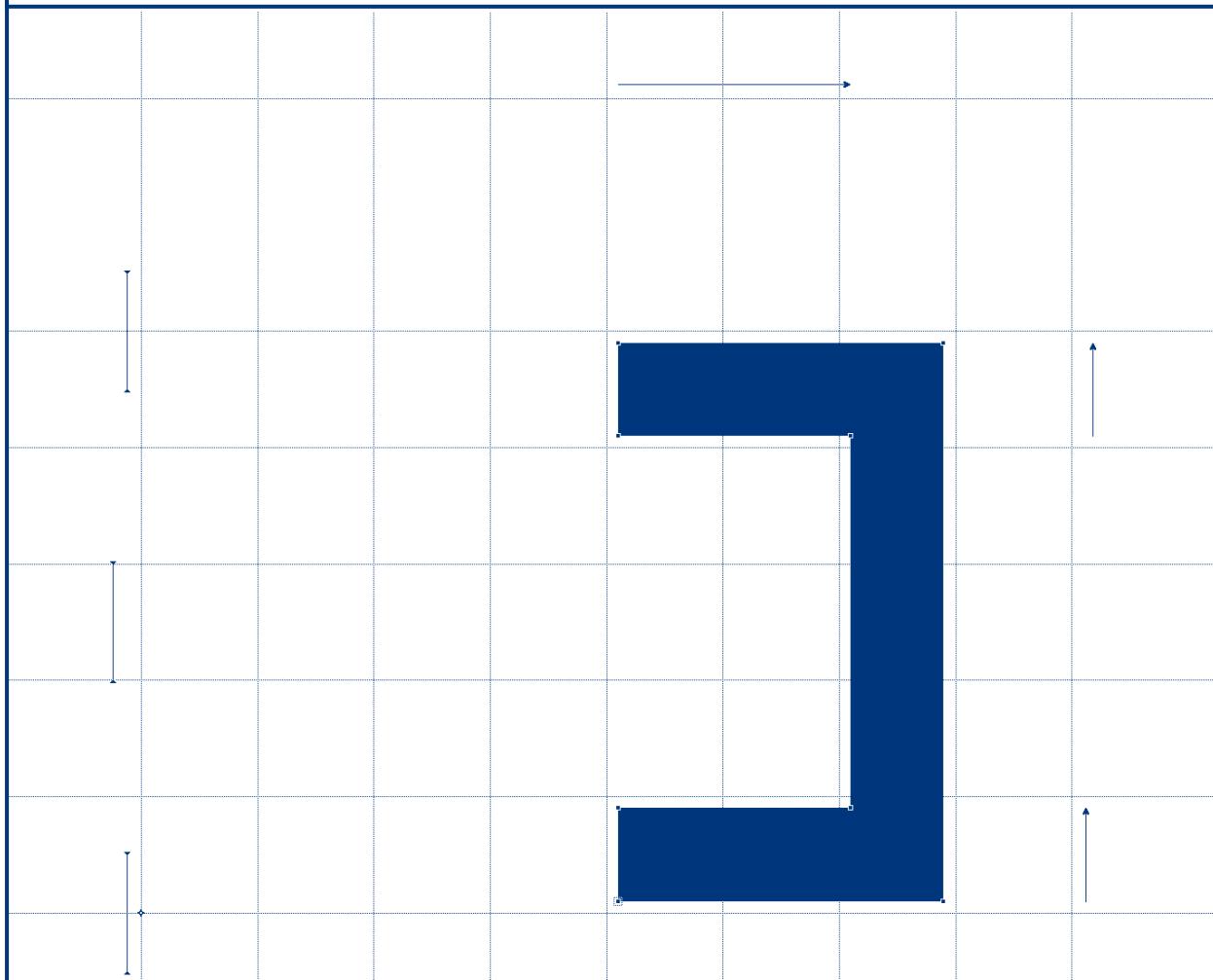
## Zeichen: Leerzeichen Art

Link Description Font © Robert Zinner 2002

Character on keyboard: -

Metrics: width 800 · left 410 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 8 pix



FONTPNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

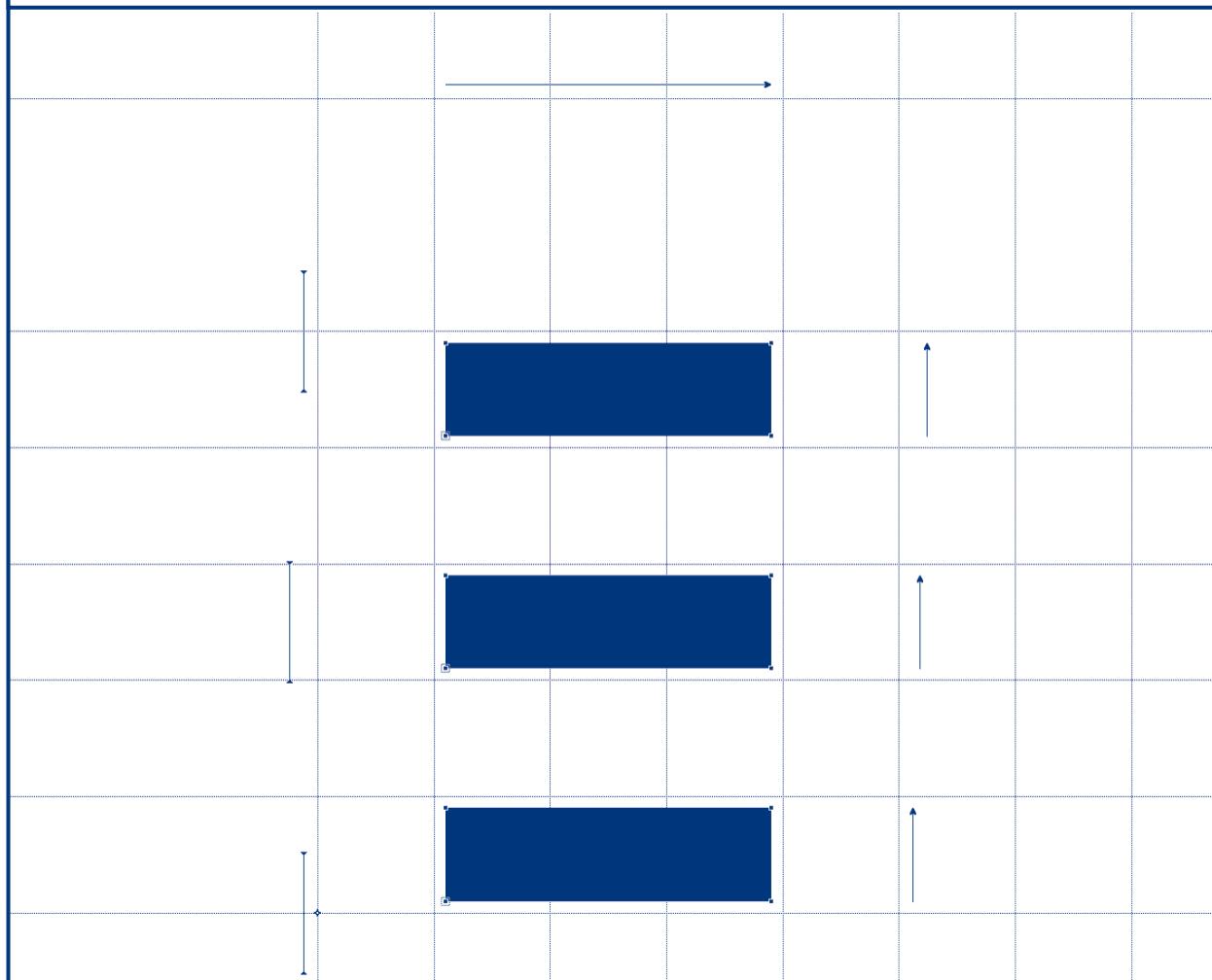
## Zeichen: Startzeichen

Link Description Font © Robert Zinner 2002

Character on keyboard: 0

Metrics: width 500 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 5 pix



FONTNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14

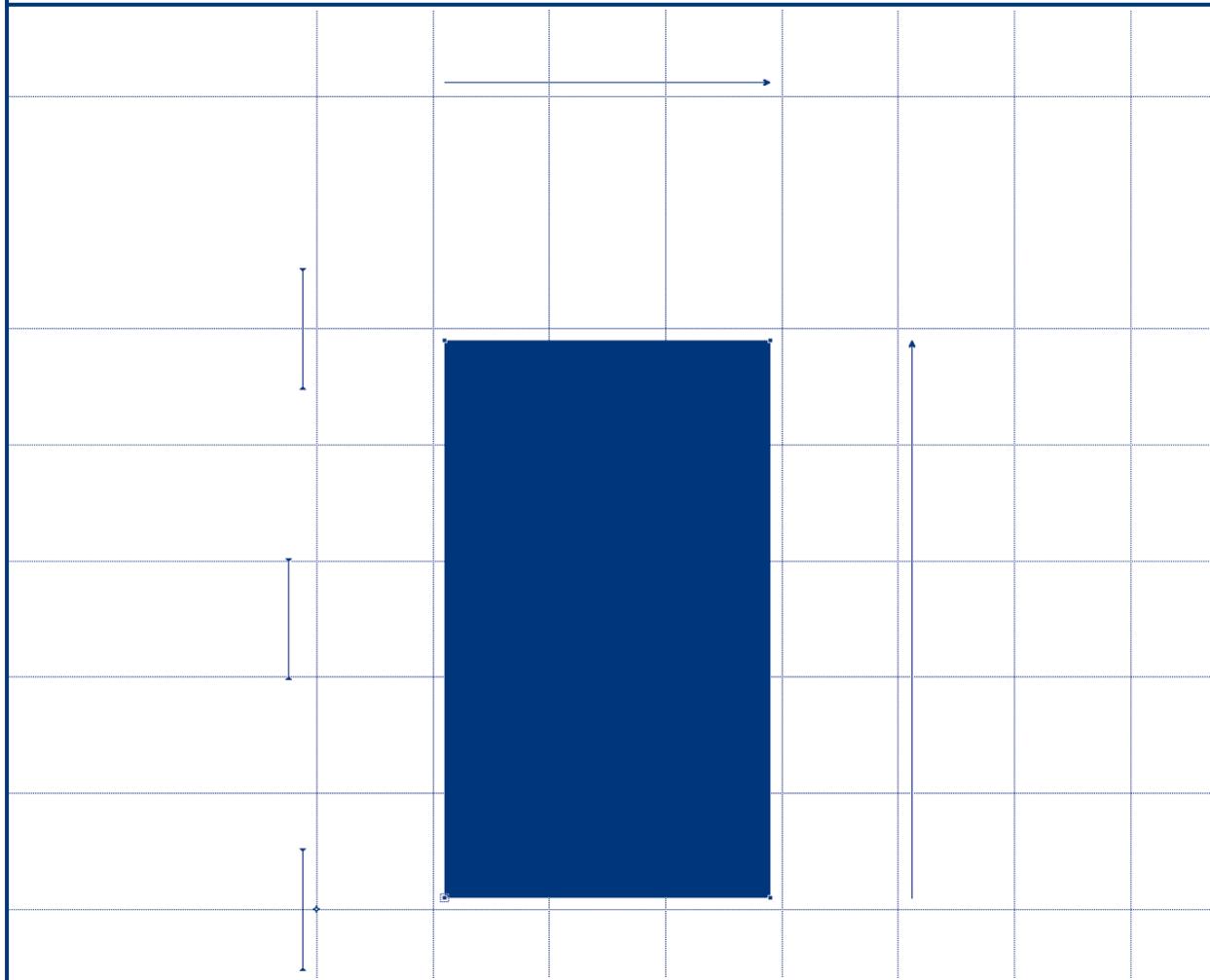
## Zeichen: Wichtungszeichen

Link Description Font © Robert Zinner 2002

Character on keyboard: +

Metrics: width 500 · left 110 · right 110 · kerning 0

Height at 9 pix fontsize: 5 pix · width 5 pix



FONTPNAME: linkfont.suit

FONT METRICS: ascent 700 · descent 200 · em square 900

CHARACTERS IN FONT: 14



# Literatur

Alle blau hervorgehobenen Quellen befinden sich nach den vorangestellten Zahlen sortiert auf der beigelegten CD. Außerdem befinden sich dort, neben den hier aufgeführten noch weitere Quellen, die im Verlauf der Arbeit gesichtet wurden.

(01) Tim Berners-Lee, D. Connolly (1995): „*Hypertext Markup Language - 2.0 - Hyperlinks*“, [http://www.w3c.org/MarkUp/html-spec/html-spec\\_7.html](http://www.w3c.org/MarkUp/html-spec/html-spec_7.html), Stand: 18. 10. 2002

(02) T. H. Nelson (1965): „*A File Structure for the Complex, The Changing and the Indeterminant*.“, in ACM 20th National Conference 1965, online verfügbar über: [http://elib.cs.berkeley.edu/cgi/pl\\_dochome?collection=Digital+Documents&tid=4](http://elib.cs.berkeley.edu/cgi/pl_dochome?collection=Digital+Documents&tid=4), Stand: 10. 06. 2002

(03) W3C (1992): „*What is Hypertext?*“, <http://www.w3.org/WhatIs.html>, Stand: 29. 05. 2002

(04) W3C (1992): „*Hypertext Terms*“, <http://www.w3.org/Terms.html>, Stand: 29. 05. 2002

(05) HCIL, University of Maryland: „*Hypertext Research: The Development of HyperTIES*“, <http://www.cs.umd.edu/hcil/hyperties/>, Stand: 23. 05. 2002

(06) Tim Berners-Lee, CERN (1989/90): „*Information Management: A Proposal*“, <http://www.w3.org/History/1989/proposal.html>, Stand: 29. 05. 2002

(07) Robert Cailliau, W3C (1995): „*A little History of the World Wide Web*“, <http://www.w3.org/History.html>, Stand: 29. 05. 2002

(08) D. Raggett, J. Lam, I. Alexander, M. Kmiec (1998): „*Raggett on HTML 4*“, Addison Wesley Longman, Essex, Kapitel 1 online verfügbar über:  
<http://www.w3.org/People/Raggett/book4/ch02.html>, Stand: 04. 09. 2002

(09) M. E. Holzschlag (1999): „*Special Edition Using HTML 4, Fifth Edition*“, Que Corporation, USA

(10) S. DeRose, E. Maler, D. Orchard (2001); „*XML Linking Language (XLink) Version 1.0*“; <http://www.w3.org/TR/xlink/>, Stand: 28. 05. 2002

(11) S. DeRose, E. Mahler, R. Daniel Jr. (2001): „*XML Pointer Language (XPointer) Version 1.0*“; <http://www.w3.org/TR/xlink/>, Stand: 28. 05. 2002

(12) M. Thüring, J. Hannemann, J. M. Haake (1995): „*Hypermedia and Cognition: Designing for Comprehension*“, in Communications of the ACM, Vol. 38, No. 8; August 1995

(13) Tim Berners-Lee (1990): „*Link Types*“, <http://www.w3.org/DesignIssues/LinkTypes.html>, Stand: 15. 05. 2002

(14) D. Raggett, A. Le Hors, I. Jacobs (1999): „*HTML 4.01 Specification – 6 Basic HTML data types*“, <http://www.w3.org/TR/REC-html40/types.html>, Stand 28. 05. 2002

(15) Randall Trigg (1983): „*A Network-Based Approach to Text Handling for the Online Scientific Community*“; Ph.D. Dissertation, University of Maryland, Chapter 4: „A Taxonomie of Link Types“ online verfügbar über: <http://www.work-practice.com/trigg/thesis-chap4.html>, Stand 24. 05. 2002

(16) Mark Bernstein (1998): „*Patterns of Hypertext*“, in Proceedings of Hypertext'98, ACM, New York, online verfügbar über:  
<http://www.eastgate.com/patterns/Patterns.html>, Stand: 14. 01. 2002

- (17) Jakob Nielsen (1998): „*Using Link Titles to help Users Predict Where They Are Going*“, <http://www.useit.com/alertbox/980111.html>, Stand: 26. 03. 2002
- (18) Eric Bosrup (2001): „*OverLIB*“, <http://www.bosrup.com/web/overlib/>, Stand: 13. 03. 2002
- (19) G. Geisler (2000): „*Enriched Links: A Framework For Improving Web Navigation Using Pop-Up Views.*“, SILS Technical Report TR-2000-02
- (20) D. Stanyer, R. Procter (1999): „*Improving Web Usability with the Link Lens*“, in WWW8 Conference Refered Papers, online verfügbar über: <http://www8.org/w8-papers/4b-links/improve/node1.html>, Stand 26. 03. 2002
- (21) H. Weinreich, W. Lamersdorf (2000): „*Concepts for Improved Visualisation of Web Link Attributes*“, in WWW9 Conference Refered Papers, online verfügbar über: <http://www9.org/w9cdrom/319/319.html>, Stand: 26. 03. 2002
- (22) P. T. Zellweger, B. Chang, J. D. Mackinlay (1998): „*Fluid Links for Informed and Incremental Link Transitions*“, in Proceeding of Hypertext'98, Pittsburg, PA, June 20-24
- (23) C. S. Campbell, P. P. Magilio (1999): „*Fasciliating navigation in information spaces: Road-signs on the World Wide Web*“, in International Journal for Human-Computer Studies Vol. 50; 1999
- (24) Matterforms Qbullets (2001): <http://www.matterform.com/qbullets/index.php>, Stand: 18. 10. 2002
- (25) Marti A. Hearst; (1995): „*TileBars: Visualization of Term Distribution Information in Full Text Information Access*“, CHI'95 Proceedings Papers, ACM, online verfügbar über: [http://www1.acm.org/sigs/sigchi/chi95/Electronic/documents/papers/mah\\_bdy.htm](http://www1.acm.org/sigs/sigchi/chi95/Electronic/documents/papers/mah_bdy.htm), Stand 24. 01. 2002



## Inhalt der CD-ROM

Auf der beigelegten CD befinden sich folgende Daten:

- der *Linkfont* im True Type Format für MacOS und Windows
- ein Manual zur Verwendung des *Link Description Fonts* im pdf-Format
- die digital verfügbaren Quellen der Arbeit sowie weitere unsortierte Quellen zur Thematik der Arbeit

## Der Link Description Font im WWW

Der *Link Description Font* ist über die Webseite  
[www.izake.de/linkfont](http://www.izake.de/linkfont) erhältlich.

