

# Universität Stuttgart

## Fakultät Informatik

**Prüfer:** Prof. Dr. Kurt Rothermel

**Betreuer:** Dipl.-Inform. Hartmut Benz

**Begonnen am:** 9. Februar 1998

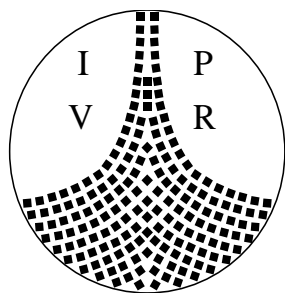
**Beendet am:** 13. Juli 1998

**CR-Nummer:** H.4.3, H.5.1

Diplomarbeit Nr.1618

## “Benutzererzeugte Annotationen auf WWW-Dokumenten”

Tilman Wolf



Institut für Parallele und Verteilte  
Höchstleistungsrechner  
Breitwiesenstraße 20-22  
D-70565 Stuttgart



## **Kurzfassung**

In dieser Arbeit wird ein System zum multimedialen Annotieren von WWW-Dokumenten entwickelt. Die Vor- und Nachteile von existierenden Systemen werden analysiert und als Grundlage für die Spezifikation eines verbesserten Annotationsystems verwendet. Es wird ein prototypischer Proxyserver mit Annotationsfunktionen implementiert, der sich anderen WWW-Komponenten gegenüber transparent verhält und somit einfach in das WWW integrierbar ist.



<b>1.0</b>	<b>Einleitung</b>	<b>9</b>
1.1	Annotationen	9
1.2	Gliederung	9
1.3	Danksagungen	10
<b>2.0</b>	<b>Existierende Systeme</b>	<b>11</b>
2.1	Überblick	11
2.2	Einfache Systeme	11
2.2.1	CoNote	11
2.2.2	Mole	12
2.2.3	Futplex	13
2.2.4	HyperNews	14
2.2.5	Bewertung	16
2.3	Komplexere Systeme	16
2.3.1	ComMentor	16
2.3.2	Annotator	17
2.3.3	Bewertung	18
2.4	Zusammenfassung	18
<b>3.0</b>	<b>Anforderungen an WWW-Annotationssysteme</b>	<b>21</b>
3.1	Benutzeranforderungen	21
3.1.1	Einfache Integration in Arbeitsumgebung	21
3.1.2	Multiuser und Informationsmanagement	21
3.1.3	Multimedia	22
3.1.4	Unterscheidbarkeit und semantische Beziehung	22
3.1.5	Verweise und benutzerdefinierte Wiedergabe	22
3.2	Implementierungsaspekte	23
3.2.1	Überdeckungen	23
3.2.2	Zeitachsen	23
3.2.3	Veränderungen	23
3.2.4	Datenmengen	24
3.3	Besonderheiten von WWW-Dokumenten	24
3.3.1	Veränderungen in WWW-Dokumenten	24
3.3.2	Skalierungsaspekte	24
3.4	Zusammenfassung	25
<b>4.0</b>	<b>DIANE</b>	<b>27</b>
4.1	DIANE Dokumente	27
4.2	Systemarchitektur	28
4.3	DIANE und WWW-Annotationssysteme	29
<b>5.0</b>	<b>Ansätze für WWW-Annotationssysteme</b>	<b>31</b>
5.1	WWW-Komponenten	31
5.1.1	Klienten und Server	31
5.1.2	Proxyserver	32
5.2	Architekturansätze	33

5.3	Globale Annotationssysteme . . . . .	35
5.3.1	Schreibzugriff auf Basisdokument . . . . .	35
5.3.2	Annotationssystem auf Server . . . . .	37
5.3.3	Annotationen bei Verfasser. . . . .	39
5.4	Lokale Annotationssysteme . . . . .	40
5.5	Annotationsproxy . . . . .	42
5.6	Zusammenfassung . . . . .	44
<b>6.0</b>	<b>WWW-Protokolle . . . . .</b>	<b>45</b>
6.1	Schichtenmodell . . . . .	45
6.2	TCP/IP . . . . .	45
6.3	HTTP . . . . .	46
6.3.1	Nachrichtentypen . . . . .	46
6.3.2	Anfrageoptionen . . . . .	47
6.3.3	Serverantworten . . . . .	47
6.3.4	Beispiel . . . . .	48
6.4	HTML. . . . .	48
6.4.1	Komponentenbeschreibung. . . . .	49
6.4.2	HTML-Kopf . . . . .	49
6.4.3	HTML-Rumpf. . . . .	50
6.5	Weitere WWW-Protokolle. . . . .	51
6.5.1	CGI-Skripte. . . . .	51
6.5.2	Java-Applets . . . . .	51
6.5.3	JavaScript . . . . .	52
6.5.4	Dynamic HTML . . . . .	52
6.5.5	SMIL. . . . .	53
<b>7.0</b>	<b>Spezifikation . . . . .</b>	<b>55</b>
7.1	Systemspezifikation . . . . .	55
7.2	Spezifikation für Annotationen . . . . .	55
7.3	Spezifikation für Datenverwaltung . . . . .	56
7.4	Klassendesign. . . . .	56
7.4.1	Klasse ProxyServer. . . . .	56
7.4.2	Klasse ProxyDatabase. . . . .	57
7.4.3	Klasse HTMLGenerator . . . . .	57
7.5	Ablaufdiagramme. . . . .	58
<b>8.0</b>	<b>Implementierung . . . . .</b>	<b>61</b>
8.1	Dokumentenmodell . . . . .	61
8.2	Darstellung von annotierten WWW-Dokumenten. . . . .	62
8.2.1	Kontrollrahmen. . . . .	62
8.2.2	Dokumentenrahmen . . . . .	63
8.2.3	Erzeugen neuer Annotationen. . . . .	63
8.3	Abläufe im Proxy . . . . .	64
8.3.1	Anfragen eines WWW-Dokuments . . . . .	65
8.3.2	Spezielle URL. . . . .	65

8.4	Datenbank.....	66
8.5	Integration weiterer Medientypen .....	67
8.6	Bewertung .....	68
<b>9.0</b>	<b>Zusammenfassung und Ausblick .....</b>	<b>69</b>
9.1	Zusammenfassung .....	69
9.2	Ausblick .....	69
<b>A</b>	<b>Annotationssysteme im WWW .....</b>	<b>71</b>
<b>B</b>	<b>Bibliographie. ....</b>	<b>73</b>
<b>C</b>	<b>Erklärung .....</b>	<b>75</b>



## **1.0 Einleitung**

Das World-Wide Web ist eine der wichtigsten Informationsquellen der heutigen Zeit. Es wird von den unterschiedlichsten Benutzergruppen verwendet. Allerdings beteiligen sich die meisten nur passiv. Sie rufen Daten ab, fügen aber keine neuen Informationen zum WWW hinzu. Ein Grund dafür ist, daß der Benutzer oft nur kleine Informationsstücke hinzufügen möchte, deren Veröffentlichung auf einer eigenen WWW-Seite zu umständlich ist.

Durch die Verwendung von Annotationssystemen ist es möglich, kleine, oft kurzlebige Annotation zu einem WWW-Dokument hinzuzufügen. Die gemeinsame Nutzung von Annotationen durch mehrere Benutzer ermöglicht somit einen Meinungs- und Gedankenaustausch.

Diese Arbeit betrachtet deshalb solche Annotationssysteme. Es werden existierende Systeme analysiert, Verbesserungen herausgearbeitet und ein neues System entwickelt.

### **1.1 Annotationen**

Der Begriff der Annotation ist besonders wichtig und soll deshalb hier definiert werden.

Annotationen sind zusätzliche Informationen, die sich auf ein Dokument beziehen. Ein Benutzer erzeugt Annotationen zu einem Basisdokument um dieses zu kommentieren, Fragen zu stellen oder einen Bezug zu anderen Dokumenten herzustellen.

Für Annotation stehen alle Medien zur Verfügung, die von einem Computer verarbeitet werden können. Typisch sind Annotationen, die aus Text, Grafik, Mauszeigerbewegungen, Audio und Video zusammengestellt sind.

Annotationen weisen prinzipiell den gleichen Aufbau auf, wie das Basisdokument. Deshalb können Annotationen wie ein Dokument behandelt und wiederum annotiert werden.

### **1.2 Gliederung**

Diese Arbeit befaßt sich mit dem theoretischen Teil der Entwicklung eines Annotationssystem. Neben dieser schriftlichen Ausarbeitung wurde aber auch ein funktionsfähiger Prototyp implementiert.

In Kapitel zwei werden einige Annotationssysteme vorgestellt, die es für WWW-Dokumente bereits gibt. Es zeigt sich, daß weder die einfachen, noch die komplexeren Systeme eine ausreichende Funktionalität bereitstellen.

In Kapitel drei werden deshalb die Anforderungen herausgearbeitet, die für ein Annotationssystem wichtig sind, so daß es sinnvoll nutzbar ist.

In Kapitel vier wird das DIANE System vorgestellt, das bereits umfangreiche Annotationsfunktionalität bereitstellt. Es wird beschrieben, wie es für WWW-Dokumente erweitert werden kann.

In Kapitel fünf wird dann ausführlich analysiert, wie die Architektur eines Annotationssystems für WWW-Dokumente aussehen kann. Dabei werden globale und lokale Annotationssysteme betrachtet. Als vorteilhafteste Architektur zeichnet sich der Annotationsproxy aus.

In Kapitel sechs wird ein Blick auf die Protokolle geworfen, die im WWW für Dokumentübertragungen verwendet werden. Das ist von Bedeutung, damit das Annotationssystem problemlos in die existierende WWW-Architektur integriert werden kann.

In Kapitel sieben ist die Spezifikation für den Annotationsproxy gegeben, der in dieser Arbeit entwickelt wurde.

In Kapitel acht wird abschließend auf einige Aspekte der Implementierung eingegangen. Es wird gezeigt, wie Dokumente und Annotationen dargestellt werden können. Außerdem werden die Abläufe im Annotationsproxy und das Datenbankmodell beschrieben.

Schließlich wird ein Ausblick auf weitere Entwicklungen gegeben.

### **1.3 Danksagungen**

Ich danke meinem Betreuer Hartmut Benz für die hilfreiche Unterstützung während der Entwicklung dieser Arbeit.

Außerdem danke ich meinen Kommilitonen Jürgen Hauser, Sebastian Heldt und Christine Heß, die mir mit ihren Ratschlägen weitergeholfen haben.

## 2.0 Existierende Systeme

In diesem Kapitel werden einige Systeme aus der Praxis vorgestellt, mit denen WWW-Dokumente annotiert werden können. Diese Annotationssysteme werden auf Funktionalität und Brauchbarkeit hin untersucht. Das Ergebnis zeigt, daß es zwar einige Ansätze gibt, aber kein komplettes System, das die wichtigsten Funktionen in sich vereinigt.

### 2.1 Überblick

Die Idee WWW-Dokumente mit Annotationen zu versehen, ist nicht neu. Die ersten Annotationssysteme, die auf dem Internet basieren, wurden Anfang der 90'er Jahre entwickelt. Anhang A enthält eine umfangreiche Auflistung dieser Annotationssysteme.

Diese Systeme wurden auf folgende Fragen hin untersucht:

- Welche Art von Annotationen (Text oder Multimedia) sind möglich?
- Wo im Dokument können Annotationen plaziert werden?
- Welche WWW-Dokumente können annotiert werden?
- Welche Software wird zusätzlich zum Browser benötigt?

Es hat sich gezeigt, daß diese Systeme grob in zwei Klassen eingeteilt werden können. Zum einen gibt es die einfachen Annotationssysteme, die sich auf textbasierte Kommentare beschränken. Zum anderen gibt es einige komplexere Systeme, die multimediale Annotationen oder komplexes Informationsmanagement erlauben.

### 2.2 Einfache Systeme

Folgende Charakteristika zeichnen einfache Annotationssysteme aus:

- nur textbasierte Annotationen, gegebenenfalls mit HTML-Formatierungen
- nur auf dem Annotationsserver gespeicherte Dokumente sind annotierbar

Es gibt etwa ein Dutzend Systeme, die diese grundlegenden Annotationsfunktionen für WWW-Dokumente bereitstellen. Die hier vorgestellten sollen einen repräsentativen Überblick darstellen.

#### 2.2.1 CoNote

CoNote ist ein Annotationssystem, das von Jim Davis und Dan Huttenlocher an der Cornell University entwickelt wurde. Ursprünglich wurde CoNote als Diskussionsforum für Vorlesungen und Übungen entwickelt. Zu den Übungsaufgaben können Studenten ihre Fragen auf dem Annotationssystem veröffentlichen oder Antworten zu anderen Fragen geben. Abbildung 1 zeigt das CoNote-System.

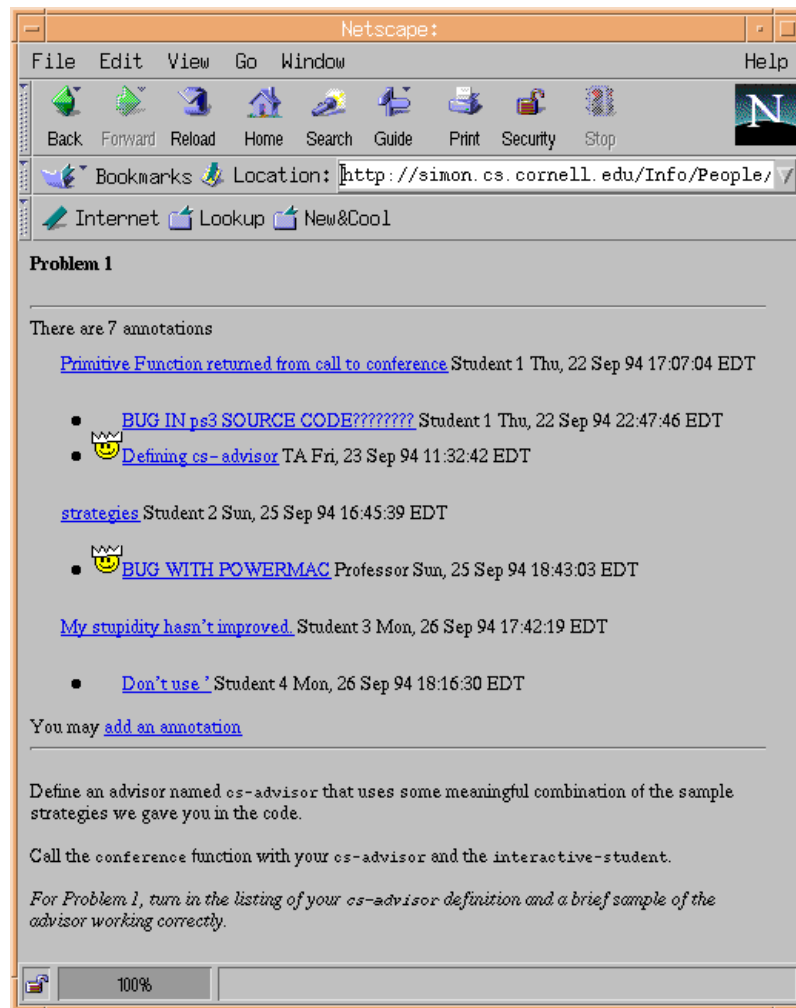


Abbildung 1 : Beispiel für das CoNote System. Auf der Basis einer Diskussionsgrundlage (hier 'Problem 1') können Kommentare, Fragen und Antworten veröffentlicht werden.

Als Basis für Annotationen dient ein Referenzartikel, zu dem am Ende dann Kommentare angefügt werden können. Die Annotationen sind als Verweise dargestellt. Beim Anklicken des Verweises wird ein weiteres Dokument mit dem Annotations-text geladen. Neue Annotationen werden über ein HTML-Formular verfaßt und an den Server übertragen.

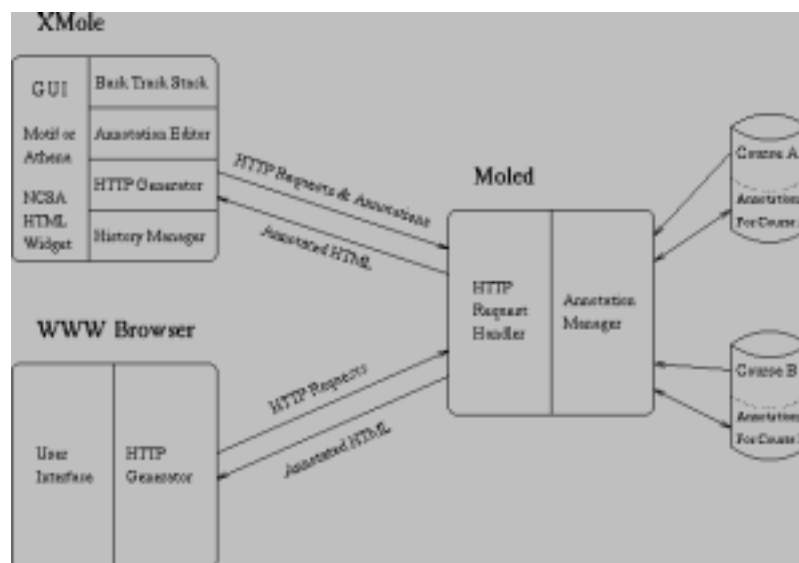
Annotationen können allerdings nur an der im Dokument dafür vorgesehenen Stelle eingefügt werden. Eine kontextabhängige Annotation ist also nicht möglich. Dennoch sind die Kommentare hierarchisch strukturiert, um Abhängigkeiten zwischen Fragen und Antworten darzustellen.

## 2.2.2 Mole

Ein weiteres einfaches Annotationssystem ist Mole. An diesem Beispiel kann die typische Architektur eines Annotationssystems gezeigt werden. Mole wurde von Dave Whittington an der University of Strathclyde at Glasgow entwickelt.

Abbildung 2 zeigt das grobe Design des Mole Systems. Zentraler Bestandteil ist der WWW-Server ('Moled'). Dieser behandelt die HTTP-Anfragen der Benutzerprogramme. Außerdem speichert der Server die WWW-Dokumente und die dazugehörigen Annotationen. Bei einer Anfrage an den Moled-Server wird das originale WWW-Dokument zusammen mit den Annotationen übertragen. Das annotierte Dokument ('annotated HTML') kann von jedem Browser dargestellt werden.

Um Annotationen erzeugen zu können, wird bei Mole eine spezielle Benutzerschnittstelle ('XMole') benötigt. Diese stellt Funktionen zur Verfügung um eine Annotation zu erzeugen und an den Server zur Speicherung zu übertragen. Bei anderen Systemen wird dies über HTML-Formulare implementiert, um zusätzliche Software beim Benutzer zu vermeiden.

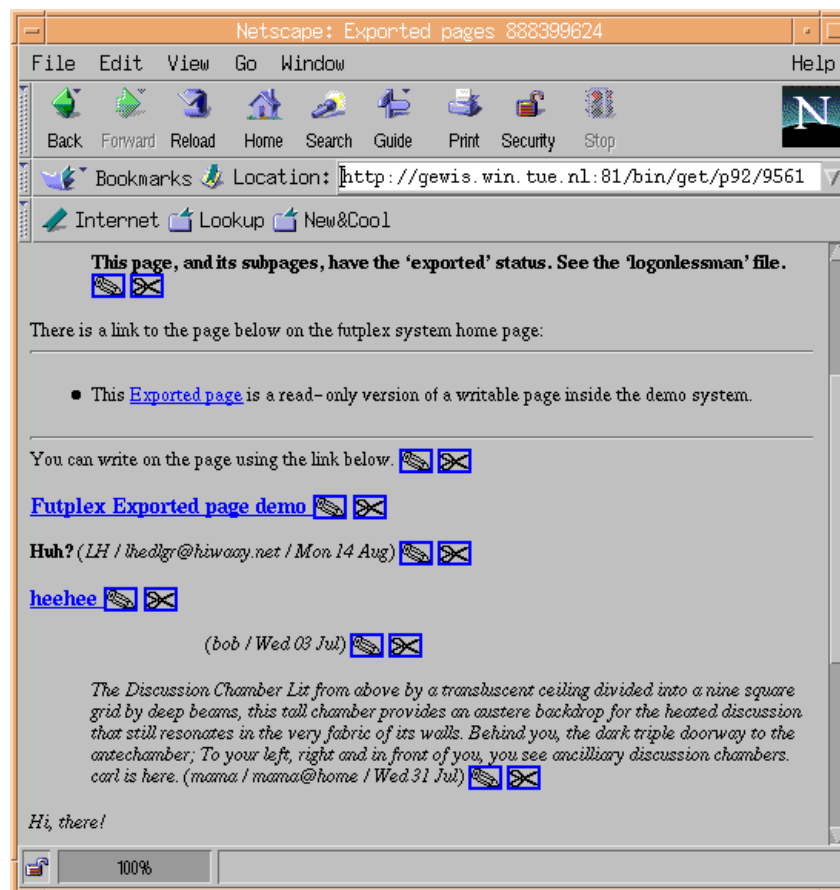


**Abbildung 2 : Architektur des Mole Systems.** Diese Architektur ist charakteristisch für einfache Annotations-Systeme. Der Server ('Moled') verwaltet die Annotationen und bearbeitet die HTTP-Anfragen. Annotationen können mit einfachen WWW-Browsern angesehen werden. Um Annotationen hinzufügen zu können, ist eine spezielle Benutzerschnittstelle ('XMole') nötig.

Wie bei den anderen vorgestellten Systemen, sind auch bei Mole die annotierbaren WWW-Dokumente lokal gespeichert. Dadurch kann der Server die Anfragen einfach bearbeiten, aber es können keine WWW-Dokumente annotiert werden, die entfernt verwaltet werden.

### 2.2.3 Futplex

Das Futplex System wurde an der Technische Universiteit Eindhoven entwickelt. Es ist zwar den vorigen Systemen sehr ähnlich, aber es bietet eine zusätzliche Funktion. Die Annotationen können beliebig formatiert werden. Das Futplex System ist in Abbildung 3 zu sehen.



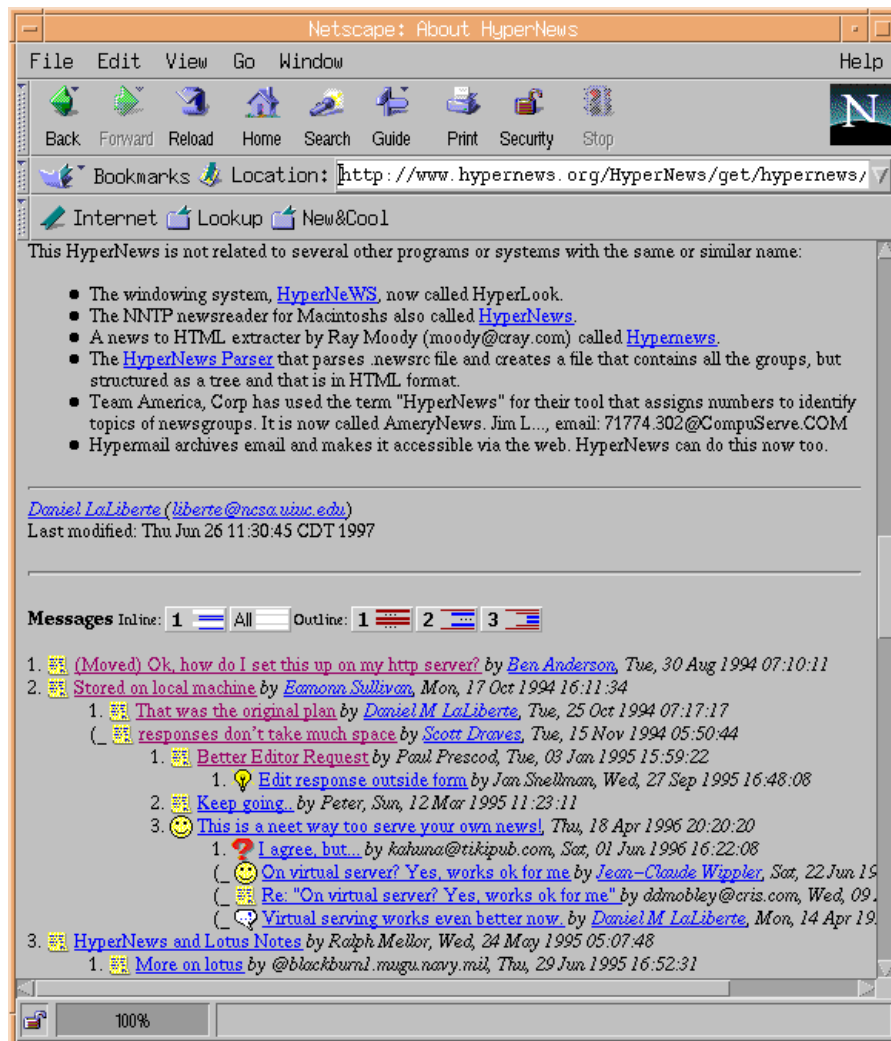
**Abbildung 3 : Das Futplex System.** Bei den Annotationen sind Sprachelemente von HTML zur Formatierung erlaubt. Die beiden Icons am Ende jeder Annotation stellen die Funktionalität zum Löschen oder Erzeugen von Annotationen bereit.

Bei der Erstellung von Annotationen können Sprachelemente von HTML verwendet werden. So kann der Text formatiert und mit WWW-Verweisen versehen werden. Die Icons am Ende jeder Annotation erlauben es, eine weitere Annotation in der nächsten Zeile einzufügen, oder die entsprechende Annotation zu löschen.

Das Futplex System stellt zwar viele wichtige Funktionen für ein Annotationssystem bereit, allerdings fehlt eine Organisation der Information. Das ganze System ist unübersichtlich, da die Annotationen uneinheitlich formatiert sind und die Annotationen nicht ein- und ausgeblendet werden können. Um dieses System sinnvoll verwenden zu können, ist sehr viel Disziplin von allen Benutzern gefragt. Bei einer Verwendung im WWW ist das aber schwierig durchzusetzen.

## 2.2.4 HyperNews

Das HyperNews System bietet im Gegensatz zu Futplex einen Ansatz, bei dem die Strukturierung der Information im Vordergrund steht. Abbildung 4 zeigt das HyperNews System.



**Abbildung 4 : Das HyperNews System.** Die Annotationen folgen dem Referenzdokument, wobei sie hierarchisch strukturiert sind. Mit den Message-Inline und -Outline Buttons können verschiedene Sichten auf die Annotationen eingestellt werden.

Am Anfang des Dokuments steht der Referenzartikel, auf den sich die Annotationen beziehen. Am Ende des Dokuments werden dann die Annotationen eingefügt, wobei diese klar hierarchisch strukturiert sind. Der Referenzartikel hat die oberste Hierarchiestufe, die Annotationen beginnen eine Stufe tiefer, wobei eine Annotation zu einer Annotation wiederum eine Stufe weiter eingerückt wird.

Es können verschiedene Sichten auf die Annotationen eingestellt werden. Die 'Messages-Inline' Knöpfe machen neben dem Annotationstitel auch den Annotationstext sichtbar. Die 'Messages-Outline' Knöpfe legen fest, bis zu welcher hierarchischen Stufe die Annotationen dargestellt werden.

### 2.2.5 Bewertung

Alle einfachen Annotationssysteme sind textbasiert. Bei manchen Systemen ist eine Formatierung der Annotation möglich. Diese Funktion kann aber dazu führen, daß das einheitliche Erscheinungsbild verloren geht und das System unübersichtlich wird.

Die Platzierung von Annotationen ist generell nur am Ende des Bezugsdokuments möglich. Bei längeren Dokumenten ist es dann schwierig den Bezugspunkt genau zu beschreiben.

Nur Dokumente, die von dem Annotationsserver verwaltet werden, können mit Annotationen versehen werden. Die meisten Systeme benötigen auf der Benutzerseite keine zusätzliche Software. Nur der Server muß entsprechend angepaßt werden.

## 2.3 Komplexere Systeme

Es gibt einige wenige Systeme, die umfangreichere Annotationsfunktionen bereitstellen als die Modelle im vorigen Abschnitt.

Diese zeichnen sich durch folgende Punkte aus:

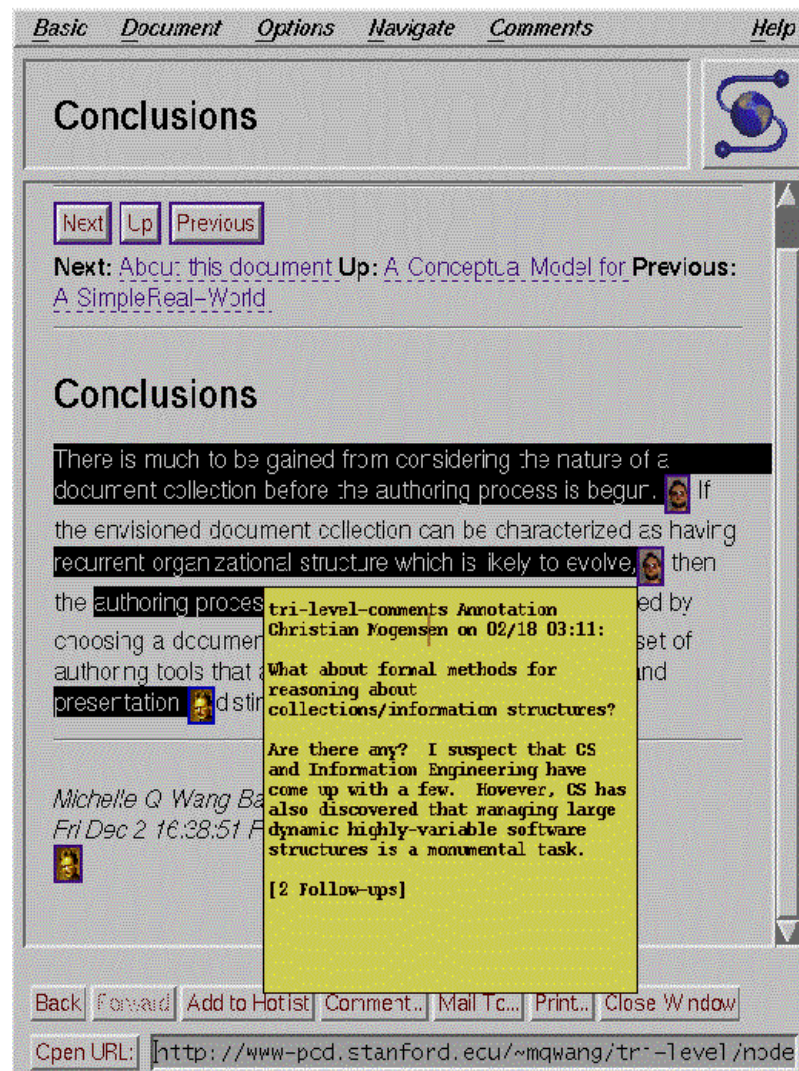
- Platzierung der Annotationen im Kontext
- multimediale Dokumente
- Informationsmanagement

### 2.3.1 ComMentor

Das ComMentor System wurde von Martin Röscheisen, Christian Morgensen und Terry Winograd an der Stanford University entwickelt. Abbildung 5 zeigt das ComMentor-System.

Die Annotationen sind an einer beliebigen Stelle im ursprünglichen WWW-Dokument platzierbar. Der dunkel markierte Bereich zeigt an, worauf sich eine Annotation bezieht. Ein Icon zeigt das Gesicht des Verfassers der Annotation. Beim Anklicken des Icons erscheint der Annotationstext in Form einer gelben Notiz auf dem Basisdokument.

Der Server des ComMentor Systems kombiniert das originale Dokument mit den Annotationsicons bei jeder Anfrage aufs Neue. Dadurch kann der Server abhängig von den Präferenzen des Benutzers verschiedene Sichten auf Annotationen ermöglichen.



**Abbildung 5 : Das ComMentor System.** Annotationen können mitten im Text angebracht werden. Sie werden als Icons mit dem Gesicht des Autors dargestellt. Bei Anklicken einer Annotation erscheint ein 'Post-it-Notiz' mit dem Annotationstext.

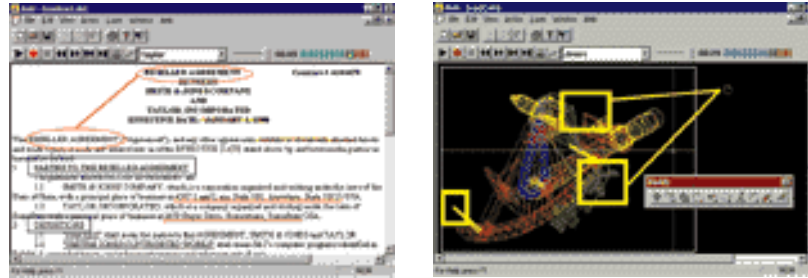
Eine weitere Besonderheit ist die Fähigkeit des Systems auch Seiten zu annotieren, die nicht auf dem ComMentor Server gespeichert sind. Diese werden auf Anfrage vom WWW geladen und mit den Annotationen versehen.

Allerdings erlaubt das ComMentor System nur textuelle Kommentare. Multimediale Annotationen sind nicht möglich. Außerdem benötigt das System spezielle Software auf Benutzerseite. Diese basiert auf alten Mosaic Browsern und ist mit modernen Browsern nicht mehr verwendbar.

### 2.3.2 Annotator

Ein neueres System ist der Annotator von FutureSoft, Inc. Dieses System ist ein umfangreiches Annotationspaket für Windows Systeme. Annotationen können Doku-

menten aller Art hinzugefügt werden, darunter auch WWW-Dokumenten. Die Annotationen werden dem Basisdokument oder einer Kopie davon beigelegt. Es werden verschiedene Schichten von Annotationen, also auch Annotationen zu Annotationen, unterstützt. Abbildung 6 zeigt Annotationen zu Text- und CAD-Dokumenten. Entsprechend können WWW-Dokumente mit Annotationen versehen werden.



**Abbildung 6 : Das Annotator System.** Die Abbildung zeigt Annotationen zu Text- und CAD-Dokumenten. Entsprechend können WWW-Dokumente annotiert werden.

Das Annotator System ist das einzige System, das Annotationen erlaubt, die nicht nur aus Text bestehen. Es können alle Medien eingebunden werden, die unter Windows verwaltet werden. Damit sind also auch multimediale Annotationen möglich.

### 2.3.3 Bewertung

Das ComMentor System implementierte als erstes System Annotationen, die beliebig platzierbar sind. Außerdem ermöglicht das umfangreiche Annotationsmanagement verschiedene benutzerdefinierte Sichten auf die Informationen.

Das Annotator System stellt sehr umfangreiche Annotationsfunktionen bereit. Allerdings ist es auf Windows Plattformen beschränkt. Zudem befindet sich die Software zur Zeit noch im Beta Test. Deshalb sind keine Informationen verfügbar, wie sich das System in der Praxis bewährt.

## 2.4 Zusammenfassung

Zwar gibt es eine Vielzahl von Annotationssystemen, aber das ideale System gibt es bis jetzt noch nicht. Die textbasierten Systeme sind nützlich, solange keine multimedialen Annotationen verwendet werden. Die komplexeren Systeme sind veraltet oder nicht ausgereift. Die größte Beschränkung liegt vor allem darin, daß in praktisch allen Systemen nur Dokumente annotiert werden können, die auch lokal vom Annotationsserver verwaltet werden.

Abschließend sind alle Eigenschaften der vorgestellten Systeme in Tabelle 1 zusammengefaßt.

**Tabelle 1: Vergleich zwischen verschiedenen Annotationssystemen für WWW-Dokumente**

System	Annotationstypen	Position	Benutzersoftware	Dokumente	Kommentar
CoNote	Text	am Ende	Browser	nur lokal	
Mole	Text	am Ende	Browser + Interface	nur lokal	
Futplex	Text + HTML	am Ende	Browser	nur lokal	unübersichtlich
Hyper-News	Text	am Ende	Browser	nur lokal	gut strukturiert
ComMentor	Text	beliebig	erweiterter Browser	lokal + entfernt	veraltet
Annotator	Multimedia	beliebig	umfangreiches System	lokal	noch im Beta-Test



## **3.0 Anforderungen an WWW-Annotationssysteme**

Um ein Annotationssystem für WWW-Dokumente zu entwickeln, das nicht die Schwächen der im vorigen Kapitel dargestellten Systeme aufweist, werden nun die grundlegenden Anforderungen beschrieben. Zunächst werden die Funktionen beschrieben, an denen der Benutzer interessiert ist. Dann werden die Auswirkungen dargestellt, die sich für die Implementierung ergeben. Lösungen werden dann in Kapitel 5 vorgestellt.

### **3.1 Benutzeranforderungen**

In diesem Abschnitt werden die Merkmale für Annotationssysteme erläutert, die für eine effektive und effiziente Nutzung durch den Benutzer wichtig sind. Insbesondere sind das:

- Einfache Integration in die Arbeitsumgebung
- Multiuser und Informationsmanagement
- Multimedia
- Unterscheidbarkeit und semantische Beziehung
- Verweise und benutzerdefinierte Wiedergabe

Diese Anforderungen gelten nicht nur für Annotationssysteme, die mit WWW-Dokumenten arbeiten, sondern auch für Annotationssysteme allgemein.

#### **3.1.1 Einfache Integration in Arbeitsumgebung**

Die wohl wichtigste Anforderung für ein Annotationssystem ist die einfache Integration des Systems in das Arbeitsumfeld des Benutzers. Nur so kann das System erfolgreich verwendet werden. Idealerweise arbeitet das Annotationssystem mit allen Anwendungen des Benutzers zusammen, die Dokumente darstellen können. So kann der Benutzer praktisch bei jeder Verwendung von Dokumenten auf deren Annotationen zurückgreifen.

#### **3.1.2 Multiuser und Informationsmanagement**

Ein weiterer wichtiger Aspekt von Annotationen ist die Möglichkeit, diese Information anderen Personen zugänglich zu machen. So können Gedanken und Meinungen ausgetauscht werden. Deshalb muß es möglich sein, daß mehrere Benutzer an einem Annotationssystem arbeiten.

Allerdings gibt es auch Annotationen, die ein Benutzer möglicherweise nicht der Allgemeinheit zugänglich machen will. Deshalb sollten Annotationen bei ihrer Erstellung nach ihrer Sichtbarkeit kategorisiert werden. Verschiedene Benutzer haben dann verschiedene Sichten auf Dokumente und Annotationen, die von den Interes-

sen und Zugriffsrechten des Benutzers abhängen.

### **3.1.3 Multimedia**

Eine Annotation sollte in allen möglichen Medien verfaßt werden können. Vor allem sollte ein Annotationssystem auf neue Medien erweiterbar sein. Damit läßt sich sicherstellen, daß das System in vielen Spezialgebieten verwendbar ist.

Spezielle Medien benötigen oft spezielle Hardware zur Wiedergabe. Falls ein solches Medium auf einem Rechner ohne diese Hardware wiedergegeben wird, sollte es möglich sein, die Daten in ein wiedergabefähiges Medium zu konvertieren.

### **3.1.4 Unterscheidbarkeit und semantische Beziehung**

Der Benutzer muß leicht erkennen können, ob ein Stück Information zum Basisdokument gehört oder eine Annotation ist. Das ist wichtig, weil der Autor des Basisdokuments nicht unbedingt mit den verschiedenen Meinungen und Ansichten übereinstimmt, die in den Annotationen geäußert werden. Es ist auch hilfreich, wenn zu jeder Annotation ersichtlich ist, wer der Verfasser ist. Damit können direkte Rückfragen an den Autor der Annotation gestellt werden.

Eine Annotation kann sich nicht nur auf das Basisdokument, sondern auch auf eine andere Annotation beziehen. Diese semantische Beziehung sollte durch eine 'a annotiert b'-Relation ausgedrückt und mit der Annotation gespeichert werden. Die semantische Beziehung ist auch insofern wichtig, daß eine Annotation, die für sich alleine dargestellt wird, oft wenig Sinn ergibt. Deshalb muß bei jeder Wiedergabe der Annotation auch das Basisdokument dargestellt werden können.

### **3.1.5 Verweise und benutzerdefinierte Wiedergabe**

Manche Annotationen, wie Text oder Grafik, können direkt zusammen mit dem Basisdokument dargestellt werden. Andere Annotationen, die auch eine bestimmte zeitliche Ausdehnung haben, wie Audio oder Mauszeigerbewegungen, müssen irgendwann aktiviert werden. Damit der Benutzer den Aktivierungszeitpunkt bestimmen kann, ist es empfehlenswert, manche Annotation als Verweise (Links) darzustellen. Durch einen Mausklick kann dann das Abspielen der Annotation initiiert werden.

Die Darstellung einer Annotation als Verweis ist auch dann sinnvoll, wenn eine große Annotation durch ihre Darstellung wichtige Teile des Basisdokuments überdeckt, oder falls die Menge der Annotationen unübersichtlich wird.

Es ist anzumerken, daß die Aktivierung einer Annotation mehrere Medien starten kann. Falls beispielsweise eine Mauszeigerannotation auf ein Video Bezug nimmt, muß natürlich auch dieses Video gestartet werden.

Auch während der Wiedergabe soll der Benutzer die Ansicht der Medien steuern

können. So müssen Funktionen wie Anhalten, langsamer und schneller Vorlauf dem Benutzer bereitstehen. Komponenten sollen auch verschoben werden können, um verdeckte Bereiche freizugeben.

## **3.2 Implementierungsaspekte**

Die Implementierung eines Annotationssystems, das die oben genannten Anforderungen erfüllt, ist eine Herausforderung. In diesem Abschnitt werden nun einige Aspekte angesprochen, die genau betrachtet werden müssen, um eine erfolgreiche Implementation zuzulassen.

### **3.2.1 Überdeckungen**

Beim Darstellen einer Annotation muß beachtet werden, daß die Annotation das Dokument nicht so überdeckt, daß ihr Bezugspunkt für den Benutzer unsichtbar wird. Sonst ist die Annotation für den Benutzer schwierig zu verstehen.

Normalerweise wird die Annotation beim Aufzeichnen vom Verfasser richtig platziert. Allerdings können Veränderungen in der Annotation dazu führen, daß Komponenten eine größere Ausdehnung haben, und damit andere Bereiche des Basisdokuments verdecken. Beispielsweise kann eine andere Standardschrift dazu führen, daß ein Textfenster mehr Platz benötigt.

### **3.2.2 Zeitachsen**

Wenn in einem Dokument mehrere Medien vereint sind, die eine zeitliche Ausdehnung haben, stellt sich die Frage, wie diese dargestellt werden sollen. Beispielsweise kann eine Annotation zu einem Video auf ein bestimmtes Standbild im Video Bezug nehmen, so daß beim Abspielen der Annotation das Video gestoppt werden muß. Oder es kann eine Annotation sein, die parallel zum Video etwas erläutert. Deshalb muß beim Aufzeichnen des Videos genau verfolgt werden, welche Komponenten zu welchem Zeitpunkt abgespielt werden.

### **3.2.3 Veränderungen**

Veränderungen in Annotationen und Basisdokumenten sind besonders problematisch. Selbst einfache Änderungen, wie Verschieben von Komponenten kann zu erheblichen Problemen bei der Wiedergabe führen. Zum Beispiel kann eine aufgezeichnete Mauszeigerbewegung dadurch ihren Bezugspunkt verlieren.

Besonders bei der Verwendung von WWW-Dokumenten sind Veränderungen eine häufig auftretende Situation. Dieser Aspekt wird im folgenden Abschnitt genauer betrachtet.

Eine ausführliche Beschreibung der Veränderungsproblematik und der Lösung durch Versionsverwaltung ist in [Hau98] dargestellt.

### **3.2.4 Datenmengen**

Bestimmte Medien erzeugen selbst bei kurzen Aufzeichnungen sehr große Datenmengen. Besonders Video und Aufzeichnungen von Anwendungsfenstern (Application Output Recording, [Ben97a]) sind dabei zu nennen. Das stellt eine Herausforderung für die Übertragungsraten aller Komponenten, besonders aber des Netzwerks, sowie für die Speicherkapazitäten des Annotationssystem dar.

Bei manchen Medien lassen sich durch die Verwendung bestimmter Codierungsalgorithmen hohe Kompressionsraten erzielen. Für Video kann Motion-JPEG oder MPEG verwendet werden. Bei Application Output Recording kann die Datenmenge verringert werden, indem Komponenten der Anwendung nicht als Bitmap gespeichert werden, sondern nur deren Charakteristika. Beispielsweise wird nur die Größe und Position des umgebenden Fensters gespeichert. Damit kann bei der Wiedergabe das gleiche Fenster erzeugt werden.

## **3.3 Besonderheiten von WWW-Dokumenten**

Bei Annotationssystemen für WWW-Dokumenten ergeben sich zusätzliche Problematiken durch die Besonderheiten, die mit WWW-Systemen verbunden sind.

### **3.3.1 Veränderungen in WWW-Dokumenten**

Viele WWW-Dokumente enthalten Informationen, die häufig aktualisiert und geändert werden. Häufig werden WWW-Dokumente auch dynamisch erzeugt und den Wünschen eines Benutzers angepaßt. Deshalb zeichnen sich WWW-Dokumente besonders durch Instabilität aus. Für eine Annotation bedeutet das, daß der Bezugspunkt möglicherweise bei späteren Ladevorgängen verschoben oder entfernt worden ist.

Bei Annotationssystemen, bei denen die Dokumente und die Annotationen vom gleichen Server verwaltet werden, kann dieses Problem umgangen werden, indem der Verfasser einer Änderung auch die Referenzpunkte der Annotationen anpaßt. Viel gravierender ist das Problem für Annotationssysteme, bei denen der Verfasser des Dokuments gar nicht weiß, daß Annotationen existieren und eine Veränderung diese beeinflußt.

### **3.3.2 Skalierungsaspekte**

Im WWW gibt es viele Millionen Benutzer. Diese Benutzer können alle auf die Dokumente eines Webservers zugreifen. Wenn auch alle diese Benutzer Annotationen zu einem Dokument hinzufügen können, kann man sich leicht vorstellen, daß dies zu einigen Problemen führt. Nach kurzer Zeit kann das Dokument mit Annotationen überfüllt sein, von denen viele einfach nicht interessant oder wegen verschiedener Sprachen nicht verständlich sind. Als Beispiel sei auf Newsgroups verwiesen, deren

Unübersichtlichkeit mit der Anzahl der Teilnehmer steigt.

Durch große Menge von Annotationen treten auch technische Probleme in den Annotationssystemen auf. Die Annotationen benötigen möglicherweise sehr viel Speicherplatz. Vor allem wenn multimediale Annotationen möglich sind, ist dies ein wichtiger Aspekt.

### **3.4 Zusammenfassung**

Um ein Annotationssystem sinnvoll verwenden zu können, muß es eine Mindestmenge an Funktionen unterstützen. Besonders die Benutzer- und Informationsverwaltung ist ein wichtiger Punkt. Außerdem sollten eine Vielzahl von Medien und Anwendungen in das System integrierbar sein.

Um diese Anforderungen zu erfüllen, müssen bei der Implementierung Details, wie Positionierung und Wiedergabe von Annotationen berücksichtigt werden. Mit der Einbindung von WWW-Dokumenten ergeben sich weitere Gesichtspunkte, wie Veränderbarkeit und Skalierung.



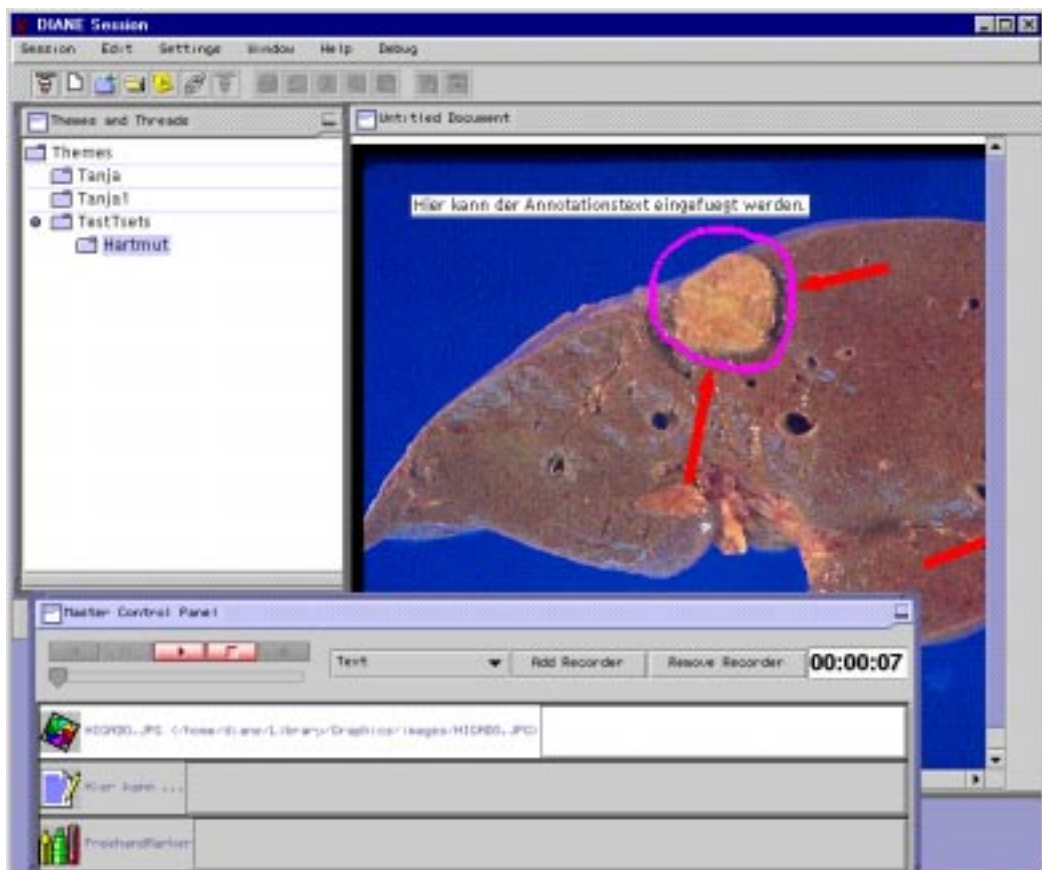
## 4.0 DIANE

DIANE (Design, Implementation and Operation of a Distributed Annotation Environment) ist ein Annotationssystem, das am IPVR der Universität Stuttgart entwickelt wird. Es erfüllt praktisch alle der oben genannten Anforderungen. Deshalb werden die in DIANE implementierten Konzepte auch als Grundlage für das hier entwickelte Annotationssystem für WWW-Dokumente verwendet.

### 4.1 DIANE Dokumente

In DIANE können multimediale Annotationen erstellt werden. Annotationen werden dabei als Dokumente behandelt. Dadurch können problemlos Annotationen zu Annotationen erstellt werden. Durch Referenzen werden semantische Beziehungen zwischen Dokumenten erstellt [Ben97b].

Die Benutzerschnittstelle ist in Abbildung 7 dargestellt.

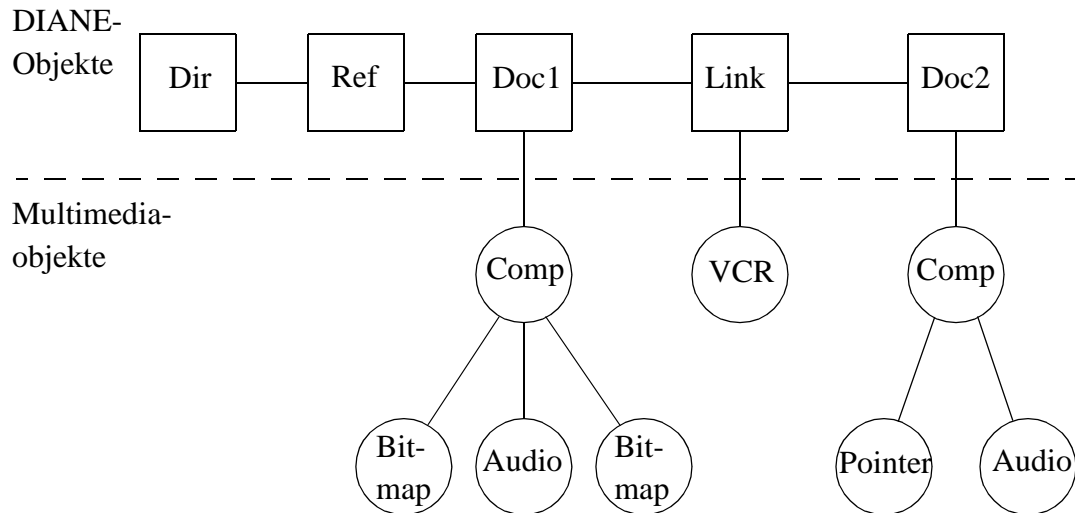


**Abbildung 7 : Die Benutzerschnittstelle von DIANE.** Das Basisdokument ist mit einem Marker und einer Textannotation versehen. Die Abspielkontrolle ist im unteren Teil des Fensters zu sehen. Dokumente und Annotationen werden im linken Teil des Fensters verwaltet.

Es ist ein Basisdokument, eine Textannotation und ein Marker dargestellt. Die Wiedergabe von Annotationen erfolgt mit der Annotationssteuerung im unteren Teil des

Fensters. Dokumente und Annotationen werden im linken Teil des Fensters verwaltet.

Abbildung 8 zeigt die interne Darstellung eines DIANE Dokuments. Das Modell veranschaulicht, wie Dokumente aus verschiedenen Medien zusammengesetzt und als DIANE Objekte verwaltet werden.



**Abbildung 8 : Das DIANE Dokumenten Modell.** Dokumente werden als DIANE Objekte in Verzeichnissen abgelegt. Semantische Beziehungen zwischen Basisdokument und Annotation werden mit einem Link beschrieben. Jedes Dokument ist aus mehreren Multimediaobjekten zusammengesetzt.

Die Abbildung zeigt die beiden Dokumente: das Basisdokument ('Doc1') und die Annotation ('Doc2'). Das Basisdokument ist über eine Referenz ('Ref') in einem Verzeichnis ('Dir') abgelegt. Es besteht aus der Komposition von zwei Bitmapgrafiken und einer Audioaufzeichnung. Die Kompositionskomponente ('Comp') speichert die Anordnung der einzelnen Medien auf dem Bildschirm und der Zeitachse. Über einen Verweis ('Link') ist die Annotation ('Doc2') an das erste Dokument gebunden. Dieser Link ist bidirektional, so daß beim Abspielen der Annotation auch das Basisdokument gefunden werden kann. Die Annotation besteht wiederum aus einer Aufzeichnung von Audio ('Audio') und Mauszeigerbewegungen ('Pointer'). Eine wichtige Komponente ist die Wiedergabesteuerung ('VCR'), die mit dem Link verbunden ist. Sie speichert, wie das Basisdokument während der Annotation wiedergegeben werden muß, damit das Dokument und die Annotation zeitlich zusammenpassen.

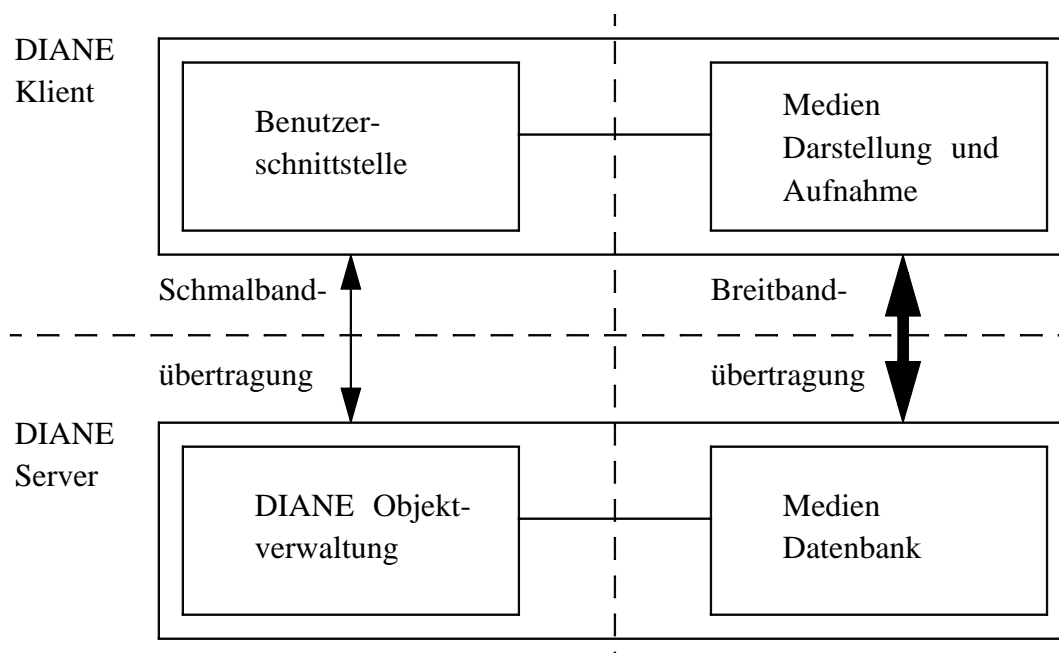
## 4.2 Systemarchitektur

DIANE ist ein Klient-Server-System. Der DIANE Server speichert die Dokumente und bearbeitet die Anfragen der verschiedenen DIANE Klienten. Auf einem Klienten-

tenrechner kann der Benutzer über die oben gezeigte Benutzerschnittstelle mit dem Server interagieren.

Das Terminal ist über eine verschlüsselte Verbindung mit dem Server verbunden. Dadurch wird die Geheimhaltung von Paßwörtern und Benutzeraktionen gewährleistet. Um aber den hohen Anforderungen an Bandbreite für Videoannotationen gerecht zu werden, existiert auch eine zusätzliche Breitbandverbindung zwischen dem Server und dem Klient.

Abbildung 9 zeigt schematisch diese Architektur.



**Abbildung 9 : DIANE Systemarchitektur.** Zwischen dem Klienten und dem Server gibt es zwei Kommunikationskanäle. Über den Schmalbandkanal werden Kontrolldaten übertragen. Für die umfangreichen Mediendaten wird der Breitbandkanal verwendet.

### 4.3 DIANE und WWW-Annotationssysteme

Ziel dieser Arbeit war die Entwicklung eines Annotationssystems für WWW-Dokumente, das mit DIANE zusammenarbeiten kann. Dadurch kann die Funktionalität beider Systeme gegenseitig verwendet werden.

Besonders die Aufzeichnungs- und Wiedergabefähigkeiten von DIANE sind nützlich. Auch die Benutzerverwaltung und das Dokumentenmodell kann für das hier entwickelte System verwendet werden.

DIANE unterstützt allerdings nicht die Verwendung von WWW-Dokumenten. Deshalb kann das Annotationssystem dieser Arbeit verwendet werden, um Funktionalität für WWW-Dokumente bereitzustellen.

Die einfachste Art der Kooperation beider Systeme kann erreicht werden, indem die Annotationen in einer gemeinsamen Datenbank abgelegt werden oder zumindest in

die jeweiligen Formate konvertiert werden können.

Viele der Konzepte, die in DIANE implementiert wurden, werden im Folgenden verwendet. Eine umfangreiche Beschreibung von DIANE ist in [Ben97c] gegeben.

## 5.0 Ansätze für WWW-Annotationssysteme

In diesem Kapitel wird analysiert, wie die Architektur eines Annotationssystem für WWW-Dokumente aussehen kann. Es wird die Funktion der verschiedenen Komponenten im WWW und ihr mögliches Zusammenspiel für ein Annotationssystem beschrieben. Die Probleme, die sich durch die Eigenschaften des WWW ergeben, wie Veränderbarkeit und Skalierung, werden analysiert. Das grundlegende Design von globalen und lokalen Annotationssystemen und die Funktionsweise eines Annotationsproxys werden detailliert erklärt.

### 5.1 WWW-Komponenten

Die Abläufe im WWW lassen sich am einfachsten durch das Konzept von Klient-Server-Architekturen beschreiben. Dabei wird das Internet als ein transparentes Netzwerk betrachtet. Komponenten unterhalb der Transportschicht, wie beispielsweise Bridges, Router und Gateways, werden nicht betrachtet.

#### 5.1.1 Klienten und Server

Die Server sind Diensteanbieter. Sie stellen Dokumente im WWW zur Verfügung. Prinzipiell sind Dokumente auf Servern für alle Klienten zugänglich. In Sonderfällen können sie auch explizit geschützt und nur autorisierten Benutzern zugänglich gemacht werden.

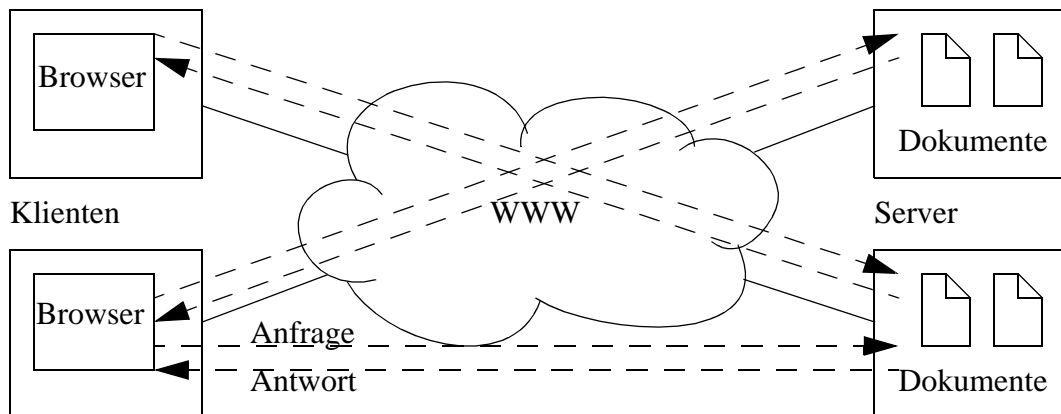
Die Klientenanwendung (Browser) kann Anfragen an verschiedene Server senden und Dokumente von diesen empfangen. Diese werden dann auf dem Klientenrechner dargestellt. Der Benutzer, der Informationen vom WWW gewinnen möchte, interagiert mit der Klientenanwendung, um die entsprechenden Dokumente gezeigt zu bekommen.

Um ein Dokument im gesamten weltweiten Netz eindeutig benennen zu können, wird ein URL (Uniform Resource Locator) verwendet. Im URL ist der Name des Servers mit optionaler Portnummer, der Pfad und Name des Dokuments, sowie das Protokoll mit dem das Dokument behandelt werden soll, enthalten. Ein Beispiel für einen URL ist 'http://www.uni-stuttgart.de/index.html'. Eine genauere Beschreibung von URL ist in [Ber94] gegeben.

Bei einer Anfrage (Request) werden Daten an einen Server gesendet, die dieser benötigt, um das gewünschte Dokument an den Klienten zu senden. Typischerweise enthält die Anfrage die Adresse des Senders und des Empfängers, den URL des Dokuments, sowie weitere Optionen, die genauer durch das verwendete Protokoll beschrieben sind.

Die Antwort (Response) des Servers enthält das Dokument, falls die Anfrage erfolgreich bearbeitet werden konnte. Auch hier werden Statuscodes und andere Verwaltungsdaten übertragen.

Ein einfaches Szenario von Dokumentanfragen ist in Abbildung 10 dargestellt.



**Abbildung 10 : Grundlegende WWW-Architektur.** Über das WWW sind Klienten und Server verbunden. Die Server stellen Dokumente bereit, die von den Klienten angefordert werden können. Die Dokumente werden vom Browser für den Benutzer dargestellt.

### 5.1.2 Proxyserver

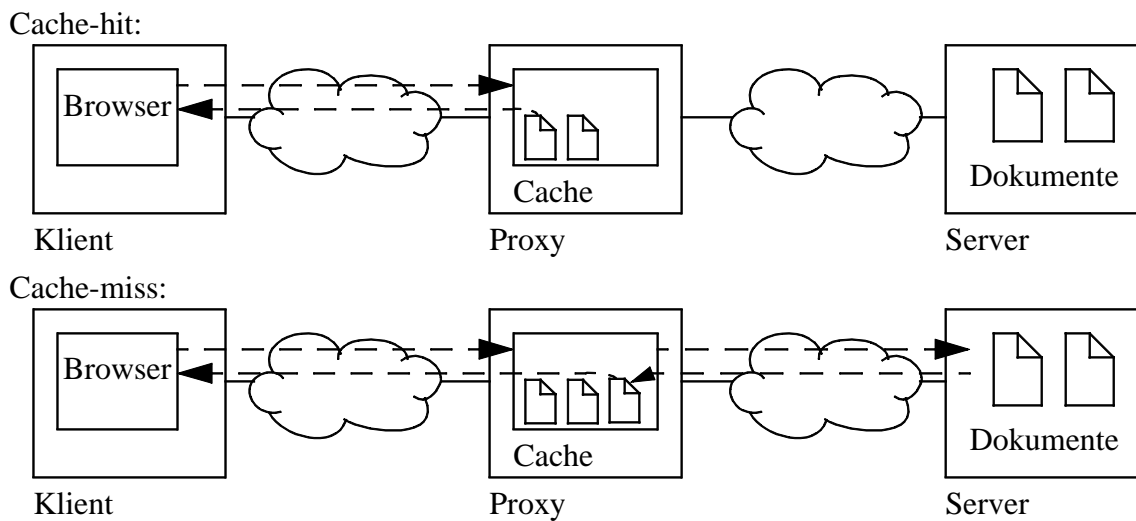
Da das WWW weltumspannend ist, können Klient und Server sehr weit auseinander liegen. Besonders bei transozeanischen Verbindungen ist Bandbreite kostbar, so daß es sinnvoll ist, häufig angefragte Dokumente an wichtigen Netzknotenpunkten zwischenspeichern. Zu diesem Zweck werden Proxyserver (Proxys) eingesetzt.

Ein Proxy nimmt die Anfrage eines Klienten entgegen und überprüft, ob das angeforderte Dokument im lokalen flüchtigen Zwischenspeicher (Cache) gespeichert ist. Falls das Dokument bereits vorhanden ist, wird es direkt an den Klienten geschickt. Eine Verbindung mit dem weit entfernten Server kann damit eingespart werden. Ist das Dokument nicht lokal vorhanden, wird es vom Proxy beim entsprechenden Server angefordert. Es wird dann im Cache abgelegt und an den Klienten übertragen.

Die Verwendung eines Proxys ist für den Benutzer und den Server völlig transparent. Die Klientensoftware muß nur insofern modifiziert werden, daß alle Anfragen an den Proxy statt an die jeweiligen Server geschickt werden. Praktisch alle kommerziellen Browser unterstützen diese Option.

Durch den Einsatz von Proxys werden allerdings mehr Verbindungen aufgebaut als bei direktem Serverzugriff. Bei hohen Cache-Trefferraten wird aber deutlich mehr Bandbreite eingespart, als durch den zusätzlichen Verbindungsauf- und -abbau verbraucht wird.

Abbildung 11 zeigt den Ablauf einer Dokumentanfrage bei Verwendung eines Proxys. Sowohl der Fall eines 'cache-hits' als auch eines 'cache-miss' ist dargestellt.



**Abbildung 11 : WWW-Architektur mit Proxyserver.** Im ersten Fall ist das angefragte Dokument im Cache des Proxys enthalten. Dadurch wird der Aufbau einer Verbindung zum Server eingespart. Im zweiten Fall ist das Dokument nicht enthalten und muß angefordert werden. Dann wird es in den Cache kopiert und zum Klienten übertragen.

Aufgrund des fast transparenten Verhaltens von Proxys können auch Proxyhierarchien verwendet werden. Dabei können Proxys verschiedener Kapazität an verschiedenen Stellen im Netzwerk eingesetzt werden. Beispielsweise werden sehr leistungsfähige Proxys an den Endpunkten von transozeanischen Verbindungen platziert. Kleinere Proxys können in lokalen Netzwerken verwendet werden, um den Zugriff auf das Internet zu verringern.

In diese Proxyhierarchie kann auch der Annotationsproxy, der in dieser Diplomarbeit entwickelt wurde, eingefügt werden.

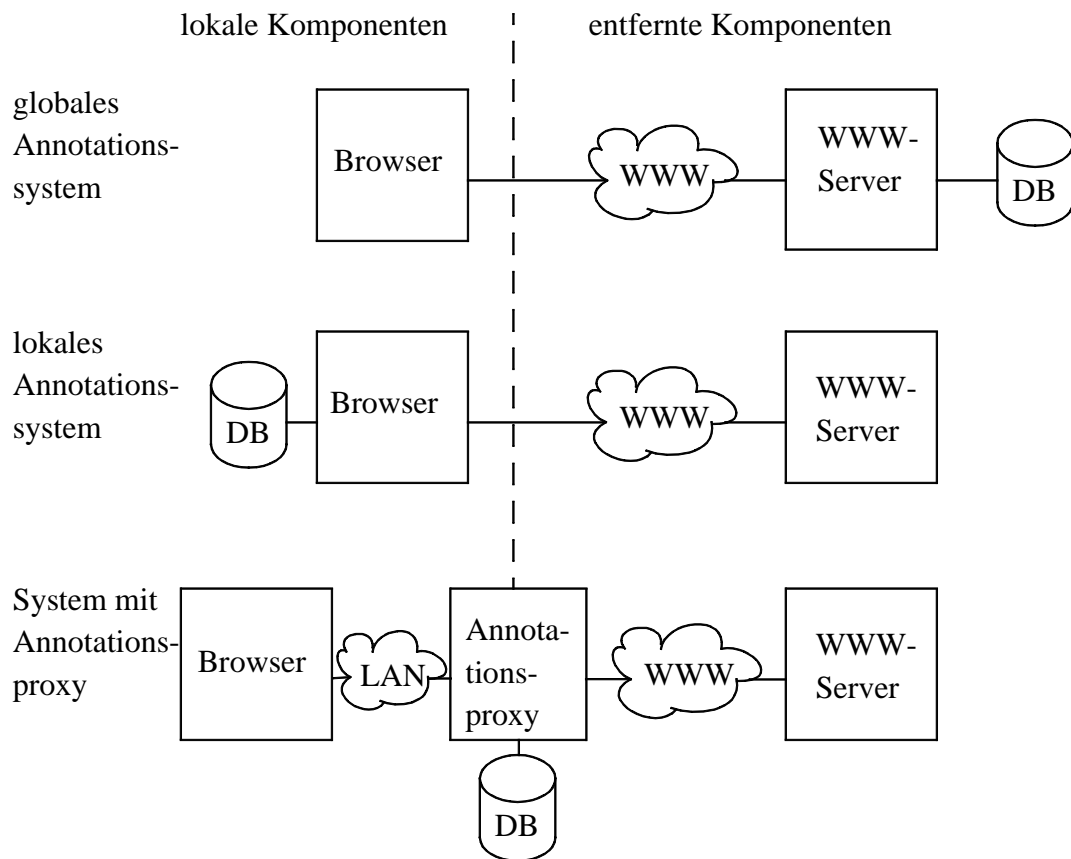
## 5.2 Architekturansätze

Um Annotationen für WWW-Dokumente zu erlauben, muß die Annotationsfunktionalität in das bisherige WWW so integriert werden, daß an der grundlegenden Klient-Server-Kommunikation nichts geändert wird. Wenn annotierte Dokumente wie gewöhnliche WWW-Dokumente behandelt werden können, sind weniger Modifikationen an bestehenden Systemen erforderlich. Damit läßt sich dann auch das Annotationssystem einfacher in den WWW-Alltag integrieren.

Deshalb können für das Annotationssystem nur diejenigen Komponenten modifiziert werden, die sowieso im WWW verwendet werden. Das sind erstens die WWW-Server, die WWW-Dokumente verwalten. Zweitens ist es das Netzwerk, das die Daten des Dokuments zum Benutzerrechner überträgt. Drittens ist das der Browser, der das Dokument darstellt. Viertens wird in manchen Systemen auch ein Proxy verwendet. Fünftens ist schließlich die Positionierung der Datenbank von Bedeutung.

Das Zusammenspiel zwischen dem Server und dem Browser über das Netzwerk ist durch die Architektur des Internets vorgegeben. Der einzige Freiheitsgrad besteht also in der Platzierung der Annotationsdatenbank und in der Verwendung eines Proxys. Abbildung 12 zeigt die drei möglichen Platzierungen.

Beim globalen Annotationssystem werden die Annotationen zusammen mit dem WWW-Dokument auf der Serverseite verwaltet. Beim lokalen Annotationssystem sind die Annotationsdaten beim Benutzer gespeichert. Der Server bleibt in diesem Fall unverändert. Bei der Verwendung eines Annotationsproxy werden die Annotationsdaten auf dem Proxy gespeichert. Sowohl der Server als auch der Browser bleiben unverändert.



**Abbildung 12 : Architekturen von Annotationssystemen.** Die Annotationsdatenbank (DB) befindet sich beim globalen Annotationssystem auf der Seite des WWW-Servers. Beim lokalen Annotationssystem werden die Annotationen auf der Klientenseite verwaltet. Der Annotationsproxy mit der Annotationsdatenbank kann als lokale oder entfernte Komponente angesehen werden.

Diese drei Ansätze werden in diesem Kapitel genauer betrachtet und ihre Vor- und Nachteile analysiert.

## **5.3 Globale Annotationssysteme**

Als erste der drei Architekturen wird nun das globale Annotationssystem betrachtet. Es zeichnet sich dadurch aus, daß die Annotationen aus Sicht des Benutzers entfernt gespeichert sind. Ein naheliegender Ansatz ist, die Annotationen zusammen mit dem originalen WWW-Dokument zu verwalten. Diese Methode kann zu Skalierungsproblemen führen. Um diesen zu entgehen, können die Annotationen auch verteilt bei ihren jeweiligen Verfassern gespeichert werden.

Globale Annotationssysteme können die Folgen, die sich durch Veränderungen in WWW-Dokumenten ergeben, wie sie in Kapitel 3.3.1 beschrieben sind, einfach umgehen. Alle Annotationen zu einem Dokument sind für den Autor des Dokuments und den Server sichtbar. Bei einer Änderung durch den Autor kann dieser prüfen, ob Referenzpunkte von existierenden Annotationen betroffen sind. In diesem Fall kann er sie so anpassen, daß die Annotationen weiterhin sinnvoll verwendet werden können. Bei dynamisch erzeugten Dokumenten kann der Server versuchen, die Referenzpunkte der Annotationen automatisch anzupassen.

### **5.3.1 Schreibzugriff auf Basisdokument**

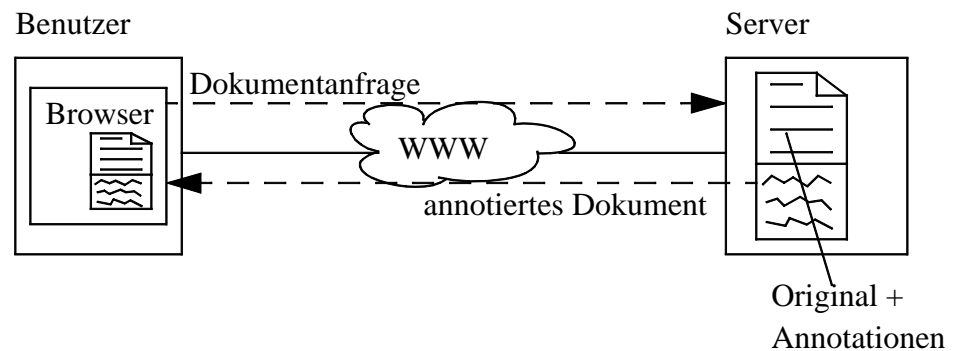
Die absolut einfachste Möglichkeit, wie eine Annotation zu einem Dokument hinzugefügt werden kann, ist, diese direkt in das Basisdokument einzufügen. Im Internet läßt sich das dadurch realisieren, daß der Schreibzugriff auf die Datei freigegeben wird, in der das Dokument abgelegt ist. Dieses Modell ist schematisch in Abbildung 13 dargestellt.

Im Falle einer Anfrage für das Dokument, sendet der Server, wie bei jeder anderen Anfrage, die Daten aus der Datei. Die Annotationen werden also automatisch mitübertragen. Wenn ein Benutzer eine Annotation hinzufügen möchte, schreibt er diese einfach in die Datei des Basisdokuments.

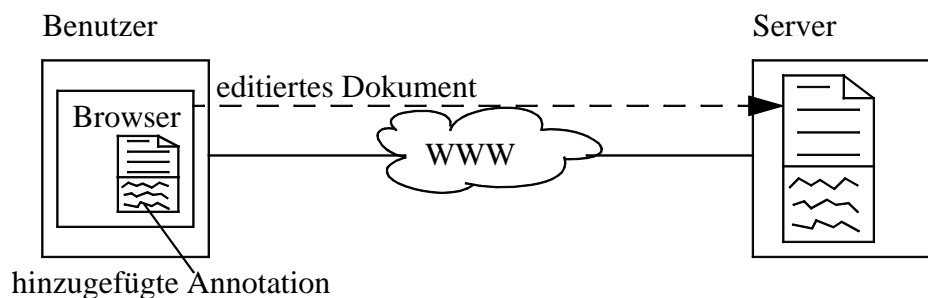
Der offensichtliche Vorteil dieses Systems besteht darin, daß keine zusätzliche Funktionalität im Server implementiert werden muß. Auch der Browser muß nur entfernte Dokumente editieren und wieder beim Server speichern können.

Allerdings ergeben sich eine Vielzahl von Nachteilen. Zunächst einmal wird das Basisdokument verändert, und der ursprüngliche Inhalt kann von Annotationen nicht unterschieden werden. Durch den beliebigen Schreibzugriff kann also der Inhalt des Basisdokuments verfälscht werden, bestehende Annotationen verändert oder gelöscht und neue Annotationen an beliebiger Stelle im Dokument hinzugefügt werden. Um dieses System sinnvoll verwenden zu können, ist eine Kooperation von allen Benutzern, die Annotationen hinzufügen wollen, erforderlich. Das ist lediglich in kleinen, gut organisierten Gruppen möglich. Als Lösung für offene Gruppen oder gar das 'ganze Internet' ist die Variante nicht einsetzbar.

Anfrage eines WWW-Dokuments



Hinzufügen einer Annotation



**Abbildung 13 : Einfaches System für globale Annotationen.** Das Original und die Annotationen sind nicht unterscheidbar, da sie in der gleichen Datei abgelegt sind. Deshalb sendet der Server bei einer Anfrage automatisch das Original zusammen mit den Annotationen. Neue Annotationen werden hinzugefügt indem sie direkt in die Datei geschrieben werden.

Außerdem treten einige weitere Probleme auf. Wenn mehrere Benutzer lokale Kopien des Dokuments gleichzeitig editieren, können Annotationen durch gegenseitiges Überschreiben verworfen werden. Dieses Problem kann gelöst werden, wird aber hier nicht weiter betrachtet.

Da die Annotationen nicht vom Dokument unterschieden werden können, ist es auch nicht möglich, diese in verschiedenen Präsentationsschichten darzustellen.

Vorteile:

- sehr einfach zu implementieren
- keine Veränderungen bei Server oder Klient nötig

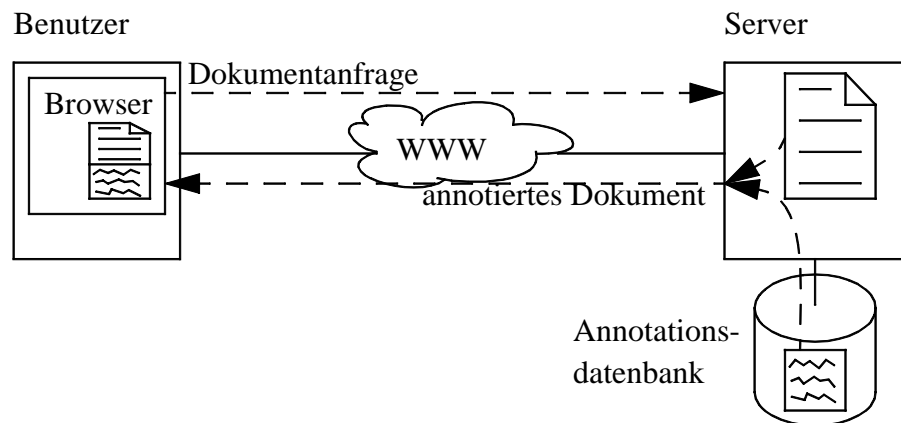
Nachteile:

- keine Trennung von Original und Annotation
- Veränderung des Originals möglich
- Kooperation der Teilnehmer erforderlich, um Übersichtlichkeit zu erhalten

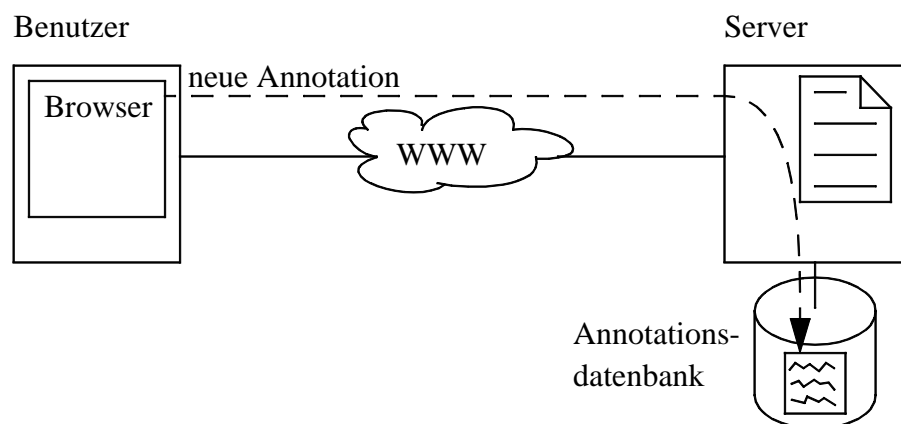
### 5.3.2 Annotationssystem auf Server

Eine wichtige Änderung, um das oben vorgestellte System zu verbessern, besteht darin, die Annotationen vom Original zu trennen. Dadurch bleibt das Original unverändert, und die Annotationen können unabhängig davon verwaltet werden. Natürlich muß dann beachtet werden, daß bei jeder Übertragung des Dokuments auch die dazugehörigen Annotationen mitgesendet werden. Dazu muß bei jeder Anfrage zunächst einmal überprüft werden, ob es überhaupt Annotationen zu dem entsprechenden Dokument gibt. Um diesen Vorgang möglichst einfach zu gestalten, wird im Folgenden eine Datenbank verwendet, in der die Annotationen gespeichert sind. Diese Datenbank ermöglicht einfache Anfragen über die Existenz von Annotationen zu Dokumenten. Im einfachsten Fall kann sie durch eine zusätzliche Datei auf dem WWW-Server implementiert werden. Ein solches System ist in Abbildung 14 dargestellt.

Anfrage eines WWW-Dokuments



Hinzufügen einer Annotation



**Abbildung 14 : Server mit Annotationsdatenbank.** Bei einer Anfrage wird überprüft, ob es Annotationen für dieses Dokument in der Datenbank gibt. Die Annotationen werden mit dem Original kombiniert und zum Klienten übertragen. Eine neue Annotation wird zum Server übertragen und dort in die Datenbank gespeichert.

Durch die Verwendung der Datenbank lassen sich einige Probleme des einfachen Systems lösen. So kann mit einem Autorisierungsverfahren sichergestellt werden, daß Annotationen nur vom jeweiligen Verfasser oder anderen berechtigten Nutzern modifiziert oder gelöscht wird. Außerdem können keine Überschreibinkonsistenzen mehr auftreten, da Annotationen einfach nur hinzugefügt werden, ohne andere Teile des Dokuments oder der Annotationen zu überschreiben.

Ein weiterer Vorteil einer Datenbank liegt darin, daß die Annotationen in verschiedene Sichten für verschiedene Benutzergruppen unterteilt werden können. Somit ist es möglich sensitive Daten vor dem Zugriff durch alle WWW-Benutzer zu schützen. Allerdings muß auch gleichzeitig angemerkt werden, daß natürlich der Verwalter des Servers, meist der Autor des Basisdokuments, alle Annotationen sehen kann. Somit erfordert eine Geheimhaltung die Kooperation dieser Person.

Die Annotationen können natürlich verschiedene Datenformate umfassen. Die Datenbank speichert einfach die zusätzlichen Dateien. Diese Daten können dann direkt auf die erste Anfrage hin übertragen werden oder verzögert über Referenzen angesprochen werden.

Die Nachteile dieses Modells bestehen darin, daß der Webserver modifiziert werden muß, um mit der Datenbank zusammenzuarbeiten. Dies führt zu einer reduzierten Leistung des Servers, da für jede WWW-Anfrage überprüft werden muß, ob es Annotationen gibt, die dann übertragen werden müssen. Dieses Problem kann dadurch reduziert werden, daß die WWW-Dokumente nicht automatisch mit Annotationen versehen werden, sondern nur auf Anfrage.

Dafür muß auf der Klientenseite keine zusätzliche Funktionalität implementiert werden, da die Annotationen in einfachen HTML-Formularen erstellt und zum Server übertragen werden können.

Vor allem mit der Möglichkeit, Annotationen mit umfangreichen Datenmengen, zum Beispiel Grafiken oder Audio, erstellen zu können, ergibt sich ein neues Problem. Der Server muß in der Lage sein, diese großen Datenmengen zu speichern und zu übertragen.

Vorteile:

- Trennung von Original und Annotation
- verschiedene Sichten auf Dokument und Annotation möglich

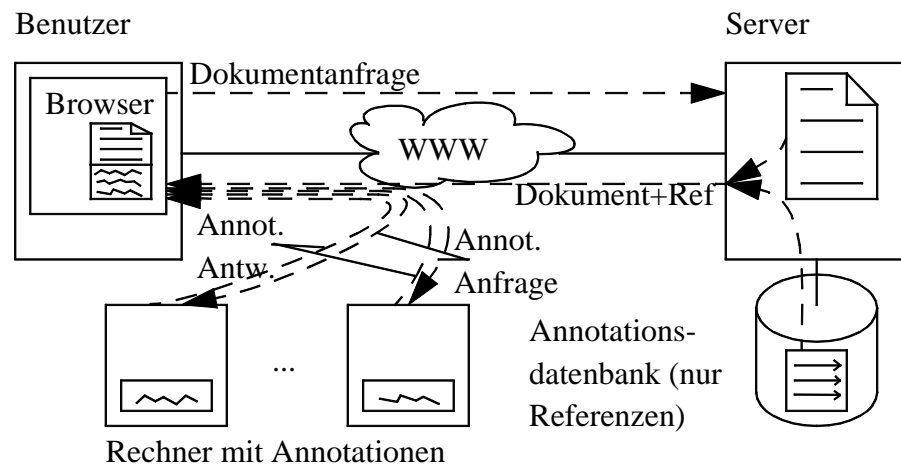
Nachteile:

- WWW-Server muß Annotationen unterstützen
- WWW-Server muß bei jeder Anfrage auch Annotationsdatenbank abfragen
- große Datenmengen belasten WWW-Server

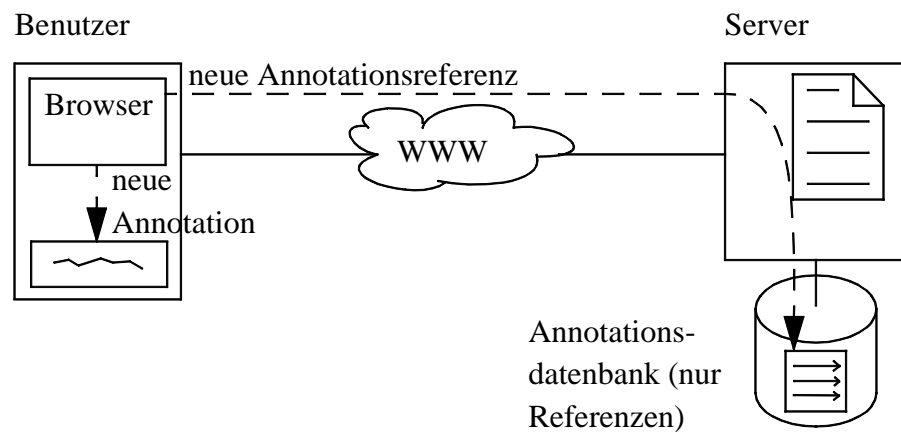
### 5.3.3 Annotationen bei Verfasser

Um eine Überlastung des Servers durch umfangreiche Annotationsdaten zu vermeiden, kann ein Modell entworfen werden, bei dem die Daten beim Verfasser der jeweiligen Annotation abgelegt werden. Anstatt der gesamten Annotation wird nur eine Referenz auf die Annotation in der Datenbank gespeichert. Diese Architektur ist in Abbildung 15 dargestellt.

Anfrage eines WWW-Dokuments



Hinzufügen einer Annotation



**Abbildung 15 : Annotationssystem mit Referenzen und verteilter Speicherung der Annotationsdaten.** Der Server des Basisdokuments fügt nur die Referenzen zu den Annotationen zum Dokument hinzu. Der Klient muß dann Anfragen an alle Rechner schicken, von denen Annotationsdaten angefordert werden sollen. Bei einer neuen Annotation wird nur die Referenz zum Server übertragen, die Annotationsdaten werden lokal gespeichert und global verfügbar gemacht.

Der entscheidende Vorteil dieses Systems liegt darin, daß der Server nun nicht mehr die ganzen Annotationsdaten speichern und übertragen muß. Statt dessen kann der Klient bei Bedarf Anfragen an die Knoten im Netzwerk senden, die eine Annotation gespeichert haben. Dadurch wird aber die Anzahl der Anfragen im Netz insgesamt

deutlich gesteigert. Das kann zu einer höheren Netzbelastung führen, als wenn alle Daten direkt vom Server zum Klient übertragen werden.

Der Browser auf der Benutzerseite sollte auch in der Lage sein, automatisch alle Referenzen zu laden (Warm Linking). Sonst muß der Benutzer jede einzelne Annotationsreferenz anwählen und diese explizit laden.

Es ist natürlich auch möglich, daß bei einer Anfrage die verteilten Annotationen vom Server geladen und dann mit dem Dokument auf einmal zum Klienten übertragen werden. Dadurch wird das Problem des Warm Linking gelöst, und der Speicherplatzbedarf am Server immer noch minimiert. Allerdings verdoppelt sich dadurch der Netzwerkverkehr beim Server, denn alle Daten müssen zum Server hin übertragen werden, nur um sofort zum Klienten weitergeschickt zu werden.

Ein anderes Problem ergibt sich dadurch, daß nun jeder Klient, der eine Annotation erstellt hat, über das Netzwerk erreichbar sein sollte, um seine Annotation bereitzustellen. Das bedeutet, daß jeder Klient auch als Server agieren können muß. Dieses Problem kann dadurch vermieden werden, daß ein Klient, der nicht zum Server werden möchte, die Daten an andere Annotationsserver zur Speicherung übertragen kann. Die Verwaltung dieser Server wird aber komplex, vor allem wenn sichergestellt werden soll, daß kein Server übermäßig belastet wird.

Vorteile:

- nur geringer Speicherplatzbedarf auf WWW-Server, da nur Referenzen gespeichert werden

Nachteile:

- viele Anfragen an unterschiedliche Server
- jeder Rechner, auf dem Annotationen verfaßt wurden, muß als Server funktionieren

## 5.4 Lokale Annotationssysteme

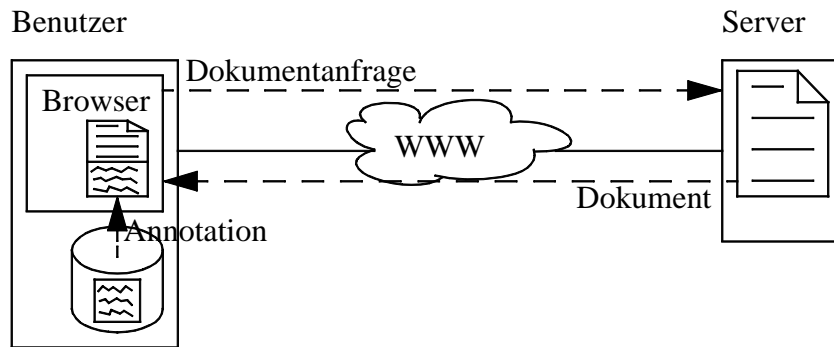
Als Alternative wird nun ein lokales Annotationssystem betrachtet, bei dem die Annotationen des Benutzers im lokalen System gespeichert werden. Weder der Server, auf dem das Basisdokument gespeichert ist, noch irgendein anderer Webteilnehmer wissen etwas von der Annotation. Dieser Ansatz ist insofern vorteilhaft, da die WWW-Server nicht mit neuer Funktionalität versehen werden müssen.

Einen einfachen Ansatz für ein lokales Annotationssystem, bei dem die Annotationsdatenbank mit dem Browser verbunden ist, stellt Abbildung 16 dar.

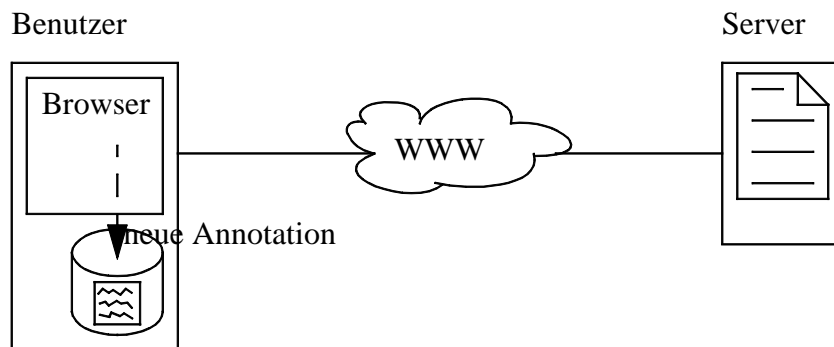
Die Annotationen zu WWW-Dokumenten sind in einer lokalen Datenbank auf dem System des Benutzers gespeichert. Wenn ein Dokument über das WWW geladen wird, überprüft der Browser, ob eine Annotation zu der entsprechenden Seite in der Datenbank existiert. Falls es eine Annotation gibt, wird diese zusammen mit dem geladenen Original dargestellt. Für den WWW-Server ist der gesamte Vorgang vom

Laden eines gewöhnlichen Dokuments nicht unterscheidbar. Deshalb kann dieses System mit allen WWW-Dokumenten funktionieren, die auf jedem beliebigen Server gespeichert sind.

#### Anfrage eines WWW-Dokuments



#### Hinzufügen einer Annotation



**Abbildung 16 : Lokales Annotationssystem mit Funktionalität im Browser.** Bei einer Anfrage wird das Dokument vom Server ohne irgendwelche Zusätze übertragen. Der Browser des Benutzers sucht in der Datenbank, ob irgendwelche Annotationen für das entsprechende Dokument existieren und stellt diese gegebenenfalls dar. Um eine Annotation hinzuzufügen, werden die Annotationsdaten einfach in der Datenbank gespeichert.

Wenn eine neue Annotation erstellt wird, dann muß der Browser diese nur in der lokalen Datenbank ablegen. Beim nächsten Ladevorgang wird dann die Annotation zusammen mit dem Dokument dargestellt.

Der große Vorteil dieses Systems ist, daß der Server unverändert bleibt. Allerdings treten nun einige Probleme auf, die bei globalen Systemen nicht vorhanden waren. Statt einer Modifikation des Servers muß nun der Browser so angepaßt werden, daß er bei jeder neuen Webseite einen Datenbankaufruf durchführt, um nach Annotationen zu suchen.

Ein weiterer Nachteil ist, daß die Annotation nur für die jeweiligen lokalen Benutzer sichtbar sind. Somit ist zwar eine Geheimhaltung leicht möglich, aber Kooperation und Gedankenaustausch zwischen verschiedenen Benutzern dieses Annotationssystems ist nicht möglich.

Daraus ergeben sich auch große Probleme, die bei Veränderungen in WWW-Dokumenten auftreten. Der Autor eines Dokuments weiß nicht, daß Annotationen zu seinem Dokument existieren. Wenn er Änderungen vornimmt, kann er also auch keine Rücksicht auf eventuelle Referenzpunkte nehmen. Deshalb kann es in einem lokalen Annotationssystem sehr leicht passieren, daß Annotationen ins Leere zeigen oder keinen Sinn ergeben.

Vorteile:

- WWW-Server müssen keine Annotationsfunktionen unterstützen

Nachteile:

- Browser muß an Annotationsdatenbank gekoppelt werden
- Annotationen sind nur für einen einzelnen Benutzer sichtbar

## 5.5 Annotationsproxy

Der Annotationsproxy ist ein Mittelweg zwischen globalen Annotationen, und Annotation, die nur für einen Benutzer zugänglich sind. Hierbei wird die Annotationsfunktionalität an einen Proxy gekoppelt, über den der Benutzerklient auf das Internet zugreift. Dadurch sind die Annotationen für alle Benutzer zugänglich, die diesen Proxy verwenden. Abbildung 17 zeigt ein solches System.

Der Browser sendet alle WWW-Anfragen an den Proxy, der diese dann an das WWW weiterleitet. Der Proxy ist mit der Annotationsdatenbank verbunden, in der nach Annotationen für das Dokument gesucht wird. Der Proxy setzt dann das Dokument und die Annotationen zusammen und sendet dies an den Browser des Benutzers.

Eine neue Annotation wird einfach an den Proxy zur Speicherung gesendet. Die Annotation wird beim nächsten Laden des WWW-Dokuments für alle beteiligten Benutzer sichtbar.

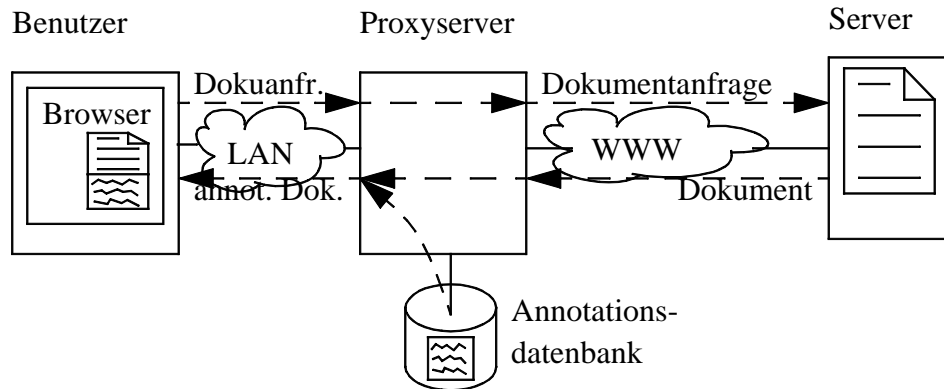
Hinsichtlich der Skalierung ergeben sich bei einem Annotationsproxy nicht die Probleme des globalen Annotationssystems, solange die Verwendung des Proxys auf eine Gruppe autorisierter Benutzer beschränkt ist. Bei Veränderungen in den WWW-Dokumenten zeigt das System aber die gleichen Probleme wie das lokale Annotationssystem, da auch hier die Annotationen für die WWW-Server unsichtbar sind.

Der entscheidende Vorteil dieses Systems liegt darin, daß das Annotieren des Dokuments fast völlig transparent für den Browser und den WWW-Server ist. Der Browser schickt seine Anfragen an den Proxy statt direkt ins WWW. Da praktisch alle gängigen Browser so konfiguriert werden können, daß sie einen Proxy verwenden, ist keine zusätzliche Software auf Benutzerseite nötig.

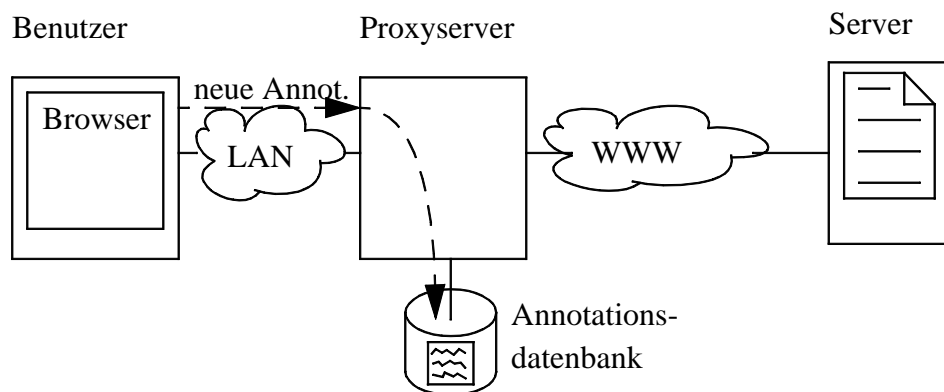
Der WWW-Server bekommt vom Proxy eine gewöhnliche WWW-Anfrage, auf die er ganz normal antwortet. Die ganze Annotationsfunktionalität ist auf den Proxy

reduziert. Der Browser muß nur die Annotationsdaten, die er vom Proxy bekommt, darstellen können.

Anfrage eines WWW-Dokuments



Hinzufügen einer Annotation



**Abbildung 17 : Annotationssystem mit Proxyserver.** Beim Laden eines Dokuments sendet der Klient eine Anfrage an den Proxy, der diese zum Server über das WWW weiterleitet. Der Proxy setzt dann das Dokument und die Annotationen zusammen und sendet alles an den Klienten. Beim Hinzufügen einer neuen Annotation wird diese in die Annotationsdatenbank am Proxy eingefügt.

Ein weiterer Vorteil dieses Systems ist, daß die Annotationen nun von einer Gruppe von Benutzern verwendet werden können. Alle Benutzer, die an das lokale Netzwerk angeschlossen sind und den Proxy verwenden, können die Annotationen sehen. Es ist natürlich auch möglich, ein Autorisierungsverfahren auf dem Proxy zu implementieren, um den Zugriff auf Annotation einzuschränken.

Die Verwendung eines Proxyserver in einem Netzwerk ist durchaus üblich. Die wichtigste Verwendung ist als Cache oder als Firewall. Solche Proxys können einfach modifiziert werden, um Annotationen zu unterstützen. In Topologien, wo es keinen Proxyserver gibt, kann dieser auch auf einem Rechner als Softwarekomponente implementiert werden.

Vorteile:

- weder Browser noch WWW-Server müssen verändert werden
- Annotationen können von einer Gruppe von Benutzern verwendet werden

Nachteile:

- WWW-Zugriffe erfolgen immer über den Proxy

## **5.6 Zusammenfassung**

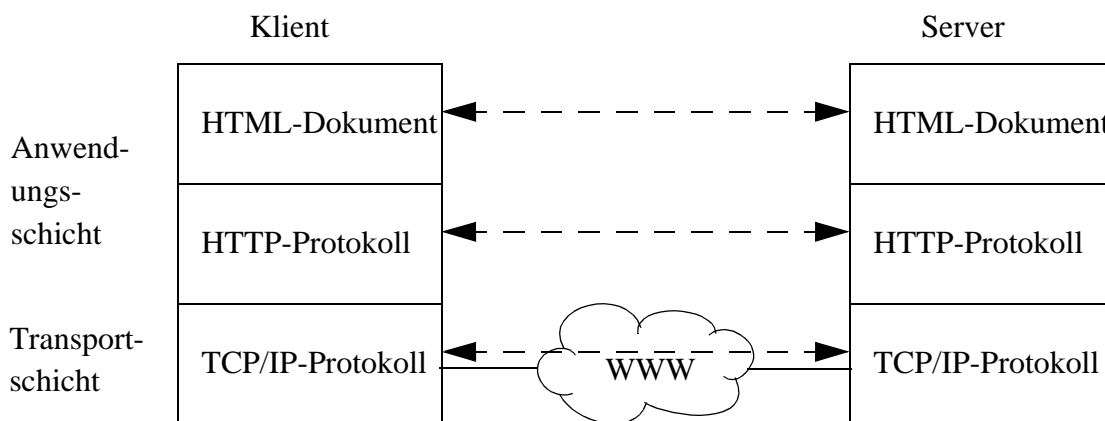
Die problemlose Integration eines Annotationssystems in das WWW setzt voraus, daß dieses mit allen existierenden Komponenten des WWW zusammenarbeitet. Deshalb ist eine minimale Veränderung auf Seiten des Benutzerklienten und des WWW-Servers wünschenswert. Der Annotationsproxy zeigt sich als vorteilhafte Architektur gegenüber globalen oder lokalen Annotationssystemen. Diese Architektur wurde als Grundlage für den restlichen Teil dieser Arbeit verwendet und auch implementiert.

## 6.0 WWW-Protokolle

Dieses Kapitel beschreibt die Protokolle und Datenformate, die im WWW verwendet werden. Von den dadurch gewonnenen Erkenntnissen wird bei der Implementierung Gebrauch gemacht. Es wird das HTTP-Protokoll beschrieben, mit dem WWW-Dokumente übertragen werden, sowie das HTML-Format, in dem diese verfaßt sind. Es wird kurz auf die TCP/IP-Verbindungen, Skripte, Java und Dynamic HTML eingegangen.

### 6.1 Schichtenmodell

Das Zusammenwirken der Protokolle im WWW ist in Abbildung 18 als Schichtenmodell dargestellt. Die Sicherungs- und Bitübertragungsschicht werden nicht betrachtet und als transparent angenommen.



**Abbildung 18 : Schichtenmodell für WWW-Dokumentübertragung.** Über eine TCP/IP-Verbindung zwischen Klient und Server werden die HTTP-Anfragen und -Antworten und damit auch das HTML-Dokument übertragen.

Zunächst wird eine TCP/IP-Verbindung zwischen dem Klienten und dem Server aufgebaut. Dann wird die im HTTP-Format formulierte Anfrage übertragen. Der Server generiert eine HTTP-Antwort, die das Dokument enthält. Die Dokumente sind in HTML verfaßt, damit eine Darstellung auf praktisch allen Arten von Klientengeräten möglich ist.

### 6.2 TCP/IP

Das Transmission Control Protocol (TCP) ist ein verbindungsorientiertes Protokoll, das einen fehlerfreien Kommunikationskanal bereitstellt. Mögliche Übertragungsfehler aus der Vermittlungsschicht, wo das Internet Protocol (IP) verwendet wird, werden in der Transportschicht durch TCP erkannt und behoben.

Für den Verbindungsaufbau benutzt der Klient die IP-Adresse des Servers, die in

dem URL enthalten ist. Falls keine Portnummer explizit spezifiziert ist, wird der für HTTP-Verbindungen übliche Port 80 verwendet.

Der Auf- und Abbau einer TCP/IP-Verbindung ist mit einigem Aufwand verbunden. So wird die Verbindung mit einem three-way-handshake geöffnet und mit einem weiteren geschlossen. Außerdem müssen einige Zustandsdaten, wie die aktuelle Paketnummer und die optimale Datenübertragungsrate, verwaltet werden. Deshalb empfiehlt es sich TCP/IP-Verbindungen zwischen Klient und Server nach der Übertragung des ersten Dokuments noch einige Zeit am Leben zu erhalten, falls weitere Dokumente, wie beispielsweise zum Dokument gehörige Grafiken, angefordert werden. Diese Vorgehensweise wird auch in [Fie98] empfohlen.

### 6.3 HTTP

Das Hypertext Transfer Protocol (HTTP) ist ein Protokoll zur Anfrage und Übertragung von Dokumenten. Es können verschiedene Optionen, wie Zeichensätze und Sprache, ausgehandelt werden. Weitere Metainformationen werden mit dem Dokument gesendet, um eine optimale Darstellung der Daten zu ermöglichen.

Die hier angegebenen Informationen beziehen sich auf den aktuellen Internet Draft der HTTP Working Group für HTTP/1.1 vom Februar 1998 [Fie98]. Dieser Entwurf wird noch weiterentwickelt, aber grundlegende Änderungen sind nicht zu erwarten. Informationen zum älteren, aber zur Zeit am meisten verwendeten HTTP/1.0, sind in [Ber96] gegeben.

#### 6.3.1 Nachrichtentypen

Eine HTTP-Nachricht besteht aus einem Kopf (Header), der Metainformationen enthält, und einem optionalen Rumpf (Body), der Nutzdaten enthält. Sowohl Anfrage als auch Antwort enthalten immer einen Kopf. Welche Nachricht auch einen Rumpf hat, hängt von der Übertragungsrichtung des Dokuments ab, und ob die Anfrage eine Dokumentübertragung vorsieht.

Unter HTTP kann der Klient durch eine Anfrage verschiedene Aktionen an einem Server auslösen. Es stehen die folgenden Anfragetypen zur Verfügung:

- GET: Anfrage für die Übertragung eines Dokuments
- HEAD: Wie GET, aber nur Übertragung der Metainformationen des Antwortkopfs
- POST: Hinzufügen des Dokuments im Anfragerumpf zum genannten URL
- PUT: Erzeugen eines neuen URL mit dem Dokument im Anfragerumpf
- DELETE: Löschen eines Dokuments
- OPTIONS: Anfragen der verfügbaren Kommunikationsoptionen
- TRACE: Anfrage für Loopbacktests

- CONNECT: Reserviert für SSL-Tunneling

### 6.3.2 Anfrageoptionen

HTTP-Nachrichten enthalten Metadaten, die die Kommunikationsoptionen spezifizieren. Beispiele für mögliche Spezifikationen sind:

- Zeichensatz
- Sprache
- Codierung des Dokuments
- Codierung des Rumpfes
- auf Klientenseite darstellbare Medien

Bei einer Anfrage kann der Klient die gewünschten Optionen angeben und diese mit Qualitätswerten versehen. Damit kann der Server versuchen, das Dokument in der für den Klienten am besten geeigneten Form zu übertragen.

Als weitere Funktionalität werden von HTTP auch Zugriffsauthentifikation, Cachinginformationen und Sicherheitsfunktionen unterstützt.

### 6.3.3 Serverantworten

Der Server bearbeitet die Anfrage des Klienten und sendet eine Antwort zurück. Die Antwort des Servers enthält zunächst einen Statuscode, der angibt, ob die Anfrage durchgeführt werden konnte, oder ob eventuell Fehler aufgetreten sind. Außerdem wird das Format des zurückgesendeten Dokuments, dessen Größe und andere Meta-information angegeben. Im Rumpf wird das Dokument gesendet.

Die fünf Kategorien der dreistelligen Statuscodes und einige Beispiele sind im Folgenden angegeben:

- 1xx: Information, keine Daten
- 2xx: Erfolg, Daten folgen, zum Beispiel '200 Document Follows' oder '201 Created'
- 3xx: Weiterleitung, Klient muß agieren, zum Beispiel '301 Moved Permanently' oder '307 Temporary Redirect'
- 4xx: Klientfehler, Anfrage unzulässig, zum Beispiel '400 Bad Request' oder '403 Forbidden'
- 5xx: Serverfehler, Server kann Anfrage nicht bearbeiten, zum Beispiel '501 Not Implemented' oder '503 Service Unavailable'

### 6.3.4 Beispiel

Ein Beispiel für eine Anfrage und die Antwort in HTTP/1.0 für das Dokument 'index.html' ist hier gezeigt. Die Anfrage lautet:

```
GET /index.html HTTP/1.0
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Accept-Language: en
Accept-Charset: iso-8859-1, *
```

Die 'Accept'-Zeile gibt dem Server an, welche Dokumententypen der Klient verwenden kann. Hier werden Grafiken im GIF-, Bitmap und JPEG-Format angefordert. Das Zeichen '\*'/\*' bedeutet, daß auch alle anderen Formate gesendet werden können. Außerdem wird das Dokument in Englisch angefordert, und der Zeichensatz sollte 'iso-8859-1' sein.

Die Antwort des Servers lautet:

```
HTTP/1.0 200 Document Follows
Content-Length: 454
Content-Type: text/html
Date: Thu, 04 Jun 1998 20:07:35 GMT
Last-Modified: Tue, 21 Apr 1998 14:04:18 GMT
```

Der Statuscode '200 Document Follows' gibt an, daß die Anfrage gültig war und der Server das Dokument übertragen wird. Über die Längenangabe ('Content-Length'), weiß der Klient wie viele Datenbytes aus dem TCP/IP-Kanal gelesen werden sollen. Das ist besonders dann von Bedeutung, wenn mehrere Anfragen und Antworten über die gleiche TCP/IP-Verbindung gesendet werden. Die Angabe des Dokumententyps ('Content-Type') ermöglicht dem Browser die korrekte Darstellung der erhaltenen Daten.

## 6.4 HTML

Die Hypertext Markup Language (HTML) ist eine Auszeichnungssprache für Textdokumente. HTML beschreibt die logischen Komponenten eines Dokuments, wie beispielsweise Überschriften oder Tabellen. Der Browser stellt das Dokument entsprechend formatiert dar. Ein wesentlicher Aspekt dabei ist, daß Dokumente auf praktisch allen Endgeräten dargestellt werden können. Dazu zählen Terminals mit verschiedenen graphischen Displays, Mobiltelefone und auch Geräte mit Sprach-ein- und -ausgabe [Rag95].

Die weitaus wichtigste Eigenschaft von HTML ist die Bereitstellung von Hyperlinks. Damit können Querverweise auf beliebige andere Dokumente im Internet erstellt werden. Bei der Darstellung in Browsern kann die entsprechende Referenz durch einen einfachen Mausklick geladen werden. So können auch verschiedene Medien, wie Grafiken oder Animationen in den Text eingebunden werden. Damit ist die Navigation durch große Mengen von Information leicht möglich, und darin liegt auch die große Beliebtheit des WWW begründet.

Es ist zu bemerken, daß HTML ein Klartextprotokoll ist. Das ursprüngliche Dokument wird nur um Klartextmarkierungen erweitert, so daß es in einem einfachen Texteditor lesbar bleibt. Damit können Korrekturen und Modifikationen leicht von Hand durchgeführt werden, ohne daß ein komplexer oder proprietärer Editor benötigt wird. Das vereinfacht auch das Ergänzen von Annotationen zu einem Dokument. Davon macht der in dieser Arbeit entwickelte Annotationsproxy Gebrauch.

### 6.4.1 Komponentenbeschreibung

HTML ist gemäß der Standard Generalized Markup Language (SGML, [ISO8879]) definiert. Jedes HTML Dokument besteht aus einem Kopf und einem Rumpf. Der Kopf beschreibt globale Eigenschaften des Dokuments. Der Rumpf enthält das Dokument an sich und zusätzliche Markierungen, die logische Strukturen beschreiben.

Eine Markierung für ein logisches Element besteht aus drei Komponenten: der Anfangsmarkierung, dem Inhalt des Elements und der Endmarkierung. Die Anfangsmarkierung legt den Typ der Markierung fest und ist durch eckige Klammern eingeschlossen, die Endmarkierung entspricht der Anfangsmarkierung, nur enthält sie ein zusätzliches '/'. Beispielsweise sieht die Beschreibung des Titels eines Dokuments so aus:

```
<TITLE> Benutzererzeugte Annotationen </TITLE>
```

### 6.4.2 HTML-Kopf

Der Kopfteil des HTML-Dokuments ist von der <HEAD>-Markierung eingerahmt. In ihm können folgende Attribute festgelegt werden:

- Titel des Dokuments
- Autor des Dokuments
- Schlüsselworte für Suchmaschinen
- HTTP-Optionen, wie Dokumententyp, Sprache und Zeichensatz
- Darstellungsinformationen (Style Sheets)
- weitere Informationen, wie Erstellungsdaten, Copyrights oder bibliographische Daten

Abgesehen vom Titel und Style Sheets werden die Informationen in einer HTML-Markierung folgenden Typs abgelegt:

```
<META NAME="Name des Attributs" CONTENT="Wert des Attributs">
```

HTTP-Optionen werden durch folgende Markierung gesetzt:

```
<META HTTP-EQUIV="Name der HTTP Option"
  CONTENT="Wert der Option">
```

Style Sheets enthalten Formatierungsinformationen für den Browser, wie Schriftarten und Farbgebung. Diese werden durch folgende Markierung beschrieben:

```
<STYLE TYPE="text/css">
  Formatierungseinheit { Formatierbefehle }
</STYLE>
```

Das folgende Beispiel zeigt einen vollständigen Kopfteil:

```
<HEAD>
  <META HTTP-EQUIV="Content-Type" CONTENT="text/html">
  <META NAME="Author" CONTENT="Tilman Wolf">
  <TITLE> Beispieldokument </TITLE>
  <STYLE TYPE="text/css">
    BODY {background: white; color: black}
  </STYLE>
</HEAD>
```

### 6.4.3 HTML-Rumpf

Der Körper des HTML-Dokuments ist von der <BODY>-Markierung umgeben und enthält das Dokument an sich. Gliedernde Markierungen gibt es für folgende logische Elemente:

- Absätze, Zeilenumbrüche, Listen, Tabellen und Überschriften verschiedener Tiefe
- Zeichenformatierungen, wie kursiv, fett, hoch-, tiefgestellt und Schriftgröße
- Hyperlinks
- Einbindung anderer Medien und Applets
- Formulare zur interaktiven Dateneingaben und JavaScript-Programme
- relative Anordnung mehrere Dokumente (Frames)

HTML 4.0 umfaßt sehr umfangreiche Funktionalität. Die Beschreibung aller Details würde den Rahmen dieser Arbeit deutlich sprengen. Deshalb sei auf die HTML 4.0 Spezifikation [Rag97] verwiesen.

Zum Abschluß sei noch ein Beispiel für ein komplettes HTML-Dokument gegeben:

```
<HTML>
  <HEAD>
    <META HTTP-EQUIV="Content-Type"
      CONTENT="text/html; charset=iso-8859-1">
    <META NAME="Author" CONTENT="Tilman Wolf">
    <TITLE> Beispieldokument </TITLE>
    <STYLE TYPE="text/css">
      BODY {background: white; color: black}
    </STYLE>
  </HEAD>
  <BODY>
    Das ist ein Beispiel fuer
    <A HREF="referenz.html"> einen Hyperlink </A> <BR>
    und eine Grafik
    <IMG SRC="picture.jpg" HEIGHT=200 WIDTH=400>
  </BODY>
</HTML>
```

## 6.5 Weitere WWW-Protokolle

Es werden nun noch einige Protokolle und Datenformate erwähnt, die im WWW verwendet werden und für die Implementierung des Annotationsproxys von Bedeutung sind.

### 6.5.1 CGI-Skripte

Das Common Gateway Interface (CGI) ist eine Möglichkeit, ausführbare Programme im WWW bereitzustellen. Sie werden von HTML-Dokumenten aus aufgerufen. Die CGI-Programme erzeugen schließlich selbst HTML-Code und senden diesen an einen WWW-Browser.

Häufig werden CGI-Programme zusammen mit HTML-Formularen verwendet. Der Benutzer füllt beispielsweise das HTML-Formular aus, die eingetragenen Daten werden an den Server übertragen und dieser führt das CGI-Programm aus. Die vom CGI-Programm erzeugten HTML-Daten werden an den Browser gesendet und als Antwort dargestellt.

Häufig sind die CGI-Programme in Perl oder C geschrieben. Für den Klienten stellen CGI-Programme kein Sicherheitsproblem dar, da nur die resultierenden HTML-Daten übertragen werden. Der Server wird allerdings belastet, da dort Rechenzeit benötigt wird. Außerdem können sich Sicherheitsprobleme ergeben, weil CGI-Skripte praktisch unbeschränkten Zugriff auf den Serverrechner haben.

CGI-Programme sind auch im Bezug auf WWW-Annotationen problematisch. Bei jedem Aufruf können unterschiedliche HTML-Daten erzeugt werden. Bezugspunkte für Annotationen können somit verschoben werden oder verloren gehen.

### 6.5.2 Java-Applets

Eine immer häufiger angewandte Methode, ausführbare Programme im WWW zur Verfügung zu stellen, ist in Form von Java-Applets. Diese sind praktisch auf allen Plattformen ausführbar, da der übertragene Bytecode auf der Java Virtual Machine im Browser interpretiert wird. Das Applet wird also auf dem Klientenrechner ausgeführt, und der Server wird nicht belastet.

Hinsichtlich der Sicherheit sind Java-Applets sehr restriktiv. Die Applets haben praktisch keinen Zugriff auf den Klientenrechner, wie beispielsweise die Festplatte. So braucht der Benutzer keine Sorge zu haben, daß er bösartige Applets vom WWW auf seinen Rechner lädt.

Es ist sehr schwierig Annotationen für Applets zu erstellen. Ein Applet ist dynamisch und kann bei verschiedenen Eingaben oder Interaktionen durch den Benutzer völlig unterschiedlich aussehen. Deshalb kann kaum entschieden werden, an welcher Position im Applet und zu welchem Zeitpunkt eine Annotation Sinn macht. Allerdings ist die Größe eines Applets im HTML-Dokument vorgegeben, so daß

zumindest die Koordinaten in einem Applet zur Positionierung zuverlässig verwendet werden können.

### 6.5.3 JavaScript

Falls in einem HTML-Dokument nur einfach Funktionen, wie kleine Berechnungen oder Pop-up-Meldungen, implementiert werden sollen, kann ein JavaScript in das HTML-Dokument eingefügt werden.

JavaScript stellt Variablen, Schleifen, Funktionen, Objekte, Methoden und Ereignisbehandlung bereit. JavaScript ist einem Java-Applet insofern überlegen, daß die Ausführung in den meisten Browsern wesentlich schneller abläuft. Außerdem muß keine zusätzliche Appletdatei übertragen werden, da der JavaScript-Code direkt im HTML-Dokument enthalten ist.

Besonders die Ereignisbehandlung läßt sich einfach mit JavaScript implementieren.

Hinsichtlich Annotationen ist auch JavaScript ein Problem. Praktisch überall im HTML-Dokument können dynamische Komponenten enthalten sein. Diese können aber durch die `<SCRIPT>`-Markierung erkannt werden. So kann man beispielsweise Annotationen auf diese Komponenten erkennen und gesondert behandeln.

### 6.5.4 Dynamic HTML

Dynamic HTML (DHTML) ist eine Erweiterung des klassischen HTML. Die grundsätzliche Idee besteht darin, daß alle Elemente einer HTML-Seite dynamisch ansprechbar und auch veränderbar sind. Neue Konzepte in DHTML sind:

- Formatierungsbeschreibungen für HTML-Daten (Cascading Style Sheets, CSS)
- beliebige Schriftarten (Dynamic Fonts)
- umfangreiche Ereignisbehandlung
- Ebenenobjekte (Layers)

Zur Zeit gibt es verschiedene proprietäre DHTML-Implementierungen. Zum einen gibt es im Netscape Navigator 4.0 dynamische Ebenenobjekte. Diese lassen sich frei gestalten und positionieren. Der Internet Explorer 4.0 von Microsoft implementiert ein umfangreicheres Objektmodell, bei dem jede HTML-Komponente ansprechbar ist. Allerdings ist der MS Internet Explorer nicht auf so vielen Plattformen verfügbar wie der Netscape Navigator.

Für die Realisierung des Annotationsproxy sind besonders die Ebenenobjekte hilfreich. Annotationen, die auf verschiedenen Ebenen positioniert sind, können somit ein- und ausgeblendet werden. Durch bewegbare Ebenen können animierte Annotationen, wie Mauszeigerbewegungen, wiedergegeben werden.

Zwar ist DHTML für die Darstellung von Annotationen hilfreich, aber das Annotieren von DHTML Basisdokumenten ist eine große Herausforderung. Praktisch

alle Komponenten des Dokuments können dynamisch verändert werden. Damit wird es äußerst schwierig, feste Referenzpunkte zu finden.

### **6.5.5 SMIL**

Die Synchronized Multimedia Integration Language (SMIL) ist ein Ansatz, Medien synchronisiert wiederzugeben. Synchronisation ist dann von Bedeutung, wenn mehrere Medien mit zeitlicher Erstreckung, wie beispielsweise Annotationen, in einem HTML-Dokument enthalten sind. In SMIL ist es möglich, die räumliche und zeitliche Position aller Medien festzulegen [Hos98].

Annotationen zu SMIL-Dokumenten können in die bereits existierenden Zeit- und Raumachsen eingebunden werden. Dadurch können die Annotationen genau nach Wunsch ihres Autors wiedergegeben werden.

Zur Zeit gibt es keine verfügbare Implementierung eines SMIL-Interpreters. Deshalb wird die Implementierung von synchronisierten Annotationen bis zur Verfügbarkeit eines SMIL-Interpreters zurückgestellt.



## 7.0 Spezifikation

In diesem Kapitel sind nun die konkreten Anforderungen aufgelistet, nach denen der Annotationsproxy dieser Arbeit entwickelt wurde. Zunächst sind die allgemeinen Systemspezifikationen, dann die Spezifikationen für die Annotationen und die Datenverwaltung angegeben. Schließlich wird durch Ablaufdiagramme die Interaktion zwischen den einzelnen Klassen gezeigt.

### 7.1 Systemspezifikation

Folgende grundlegende Eigenschaften muß der Annotationsproxy erfüllen:

- Der Annotationsproxy arbeitet als transparenter Proxyserver, der Annotationen für HTML-Dokumente verwaltet und bei Anfragen hinzufügt.
- Der Annotationsproxy ist über eine TCP/IP-Verbindung mit dem HTTP/1.0-Protokoll ansprechbar.
- Die Annotationen können von einem herkömmlichen Browser (beispielsweise Netscape Navigator 4.0) dargestellt werden.
- Neue Annotationen können mit einem herkömmlichen Browser verfaßt werden.
- Die Annotationen werden in einer Datenbank gespeichert und verwaltet. Der Zugriff auf die Datenbank findet über die JDBC/ODBC-Schnittstelle statt.
- Der Annotationsproxy ist in Java 1.1 implementiert.

### 7.2 Spezifikation für Annotationen

Folgende Funktionalität müssen die Annotationen aufweisen:

- Die Annotationen können aus folgenden Medien bestehen: Text, Hyperlink, Marker oder Mauszeigerbewegung.
- Annotationen können zum Originaldokument oder zu anderen Annotationen hinzugefügt werden.
- Textannotationen können durch HTML-Markierungen formatiert werden.
- Mauszeiger können durch verschiedene Symbole dargestellt werden.
- Marker können durch verschiedene Symbole dargestellt werden.
- Annotationen können durch den Benutzer verdeckt oder sichtbar gemacht werden.
- Animierte Annotationen können durch den Benutzer gestartet werden.

### 7.3 Spezifikation für Datenverwaltung

Folgende Funktionalität muß durch die Datenverwaltung gegeben sein:

- Zu jeder Annotation wird gespeichert: Bezugsdokument, Autor, Sichtbarkeit, Zeitpunkt der Erstellung.
- Jeder Benutzer ist einer Gruppe zugeordnet.
- Es gibt eine Benutzerverwaltung, so daß ein Benutzer alternativ seine Annotationen, die Annotationen seiner Gruppe oder alle sichtbaren Annotationen ansehen kann.
- Es ist möglich, daß ein Benutzer nur Annotationen, die nach einem bestimmten Zeitpunkt erstellt wurden, betrachten kann.
- Es ist möglich eine Liste aller WWW-Dokumente zu erhalten, zu denen Annotationen existieren.
- Annotationen können gelöscht werden, sofern der Benutzer gemäß der Sichtbarkeitsregeln das Recht dazu hat.

### 7.4 Klassendesign

Der Annotationsproxy soll durch die folgenden Klassen implementiert werden:

- die Klasse `ProxyServer`, die HTTP-Anfragen aus dem Netz entgegennimmt und die entsprechenden Antworten sendet
- die Klasse `ProxyDatabase`, die alle Datenbankzugriffe durchführt
- die Klasse `HTMLGenerator`, die aus den Annotationsdaten HTML-Code erzeugt

Im Folgenden wird genauer auf die Funktionalität dieser Klassen eingegangen. Für die vollständige Auflistung der Methoden sei auf die JavaDoc Dokumentation verwiesen, die mit der Implementierung des Proxys verfügbar ist.

#### 7.4.1 Klasse `ProxyServer`

Die Klasse `ProxyServer` implementiert die Grundfunktionalität eines Proxyserver. Sie nimmt HTTP-Anfragen aus dem Netz entgegen und sendet die entsprechenden Antworten. Bei jeder Anfrage werden folgende Schritte durchgeführt:

- Entgegennehmen der Anfrage
- Extrahieren und Überprüfen des Benutzernamens und Paßworts
- Entscheiden welches Dokument auf Anfrage gesendet wird
- bei dynamisch erzeugten Dokumenten Annotationsdaten von `ProxyDatabase` anfordern und `HTMLGenerator` zur Generierung des HTML-Codes aufrufen

- bei Dokumenten, die aus dem WWW geladen werden, Kontaktieren des entsprechenden WWW-Servers und Anfragen des Dokuments
- Antwort mit erhaltenem Dokument senden

Die Klasse `ProxyServer` enthält das Hauptprogramm des Annotationsproxys. Um mehrere Anfragen nebeneinander bearbeiten zu können, ist diese Klasse durch Threads implementiert.

## 7.4.2 Klasse `ProxyDatabase`

Die Klasse `ProxyDatabase` stellt die Schnittstelle des Annotationsproxys zur Datenbank dar. Alle Datenbankabfragen werden von dieser Klasse durchgeführt. Bei jedem Aufruf wird eine SQL-Abfrage erzeugt, die über die JDBC/ODBC-Schnittstelle an die Datenbank gesendet wird. Die folgenden Methoden werden zur Annotationsverwaltung durch die Klasse bereitgestellt:

- `checkUser()`, überprüft Name und Paßwort
- `getAnnotationsForSite()`, gibt Annotationsdaten für einen URL zurück
- `addAnnotation()`, fügt eine Annotation zu einem URL hinzu
- `deleteAnnotation()`, löscht eine Annotation

Außerdem gibt es Methoden zur Verwaltung der Datenbank:

- `createTables()`, erzeugt alle Tabellen, die vom Annotationsproxy in der Datenbank benötigt werden
- `addUser()`, erzeugt einen neuen Benutzer
- `deleteUser()`, löscht einen Benutzer
- `changePassword()`, ändert das Paßwort eines Benutzers
- `addGroup()`, erzeugt eine neue Benutzergruppe
- `deleteGroup()`, löscht eine Benutzergruppe

In manchen Fällen besteht der Rückgabewert einer Methode aus mehreren Datensätzen, wie beispielsweise bei `getAnnotationsForSite()`. Dann wird ein Vektor mit den entsprechenden Datensätzen zurückgegeben.

## 7.4.3 Klasse `HTMLGenerator`

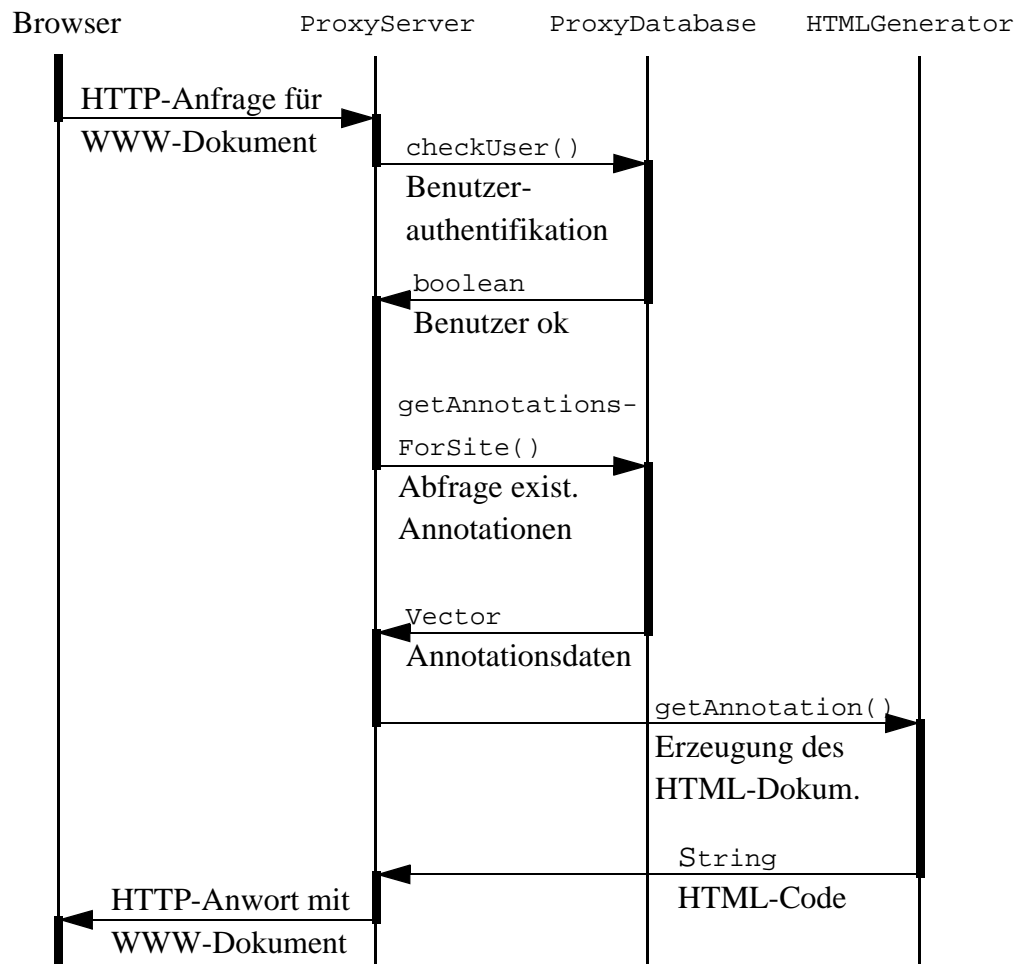
Die Klasse `HTMLGenerator` erzeugt den HTML-Code, der dynamisch vom Annotationsproxy erstellt wird. Die Erzeugung geschieht auf der Basis der Annotationsdaten, die von `ProxyDatabase` bereitgestellt werden. Die folgenden Methoden erzeugen die Dokumente, die im Dokumentenmodell in Kapitel 8.1 und 8.2 beschrieben sind:

- `getOriginal()`, erzeugt das Rahmengerüst

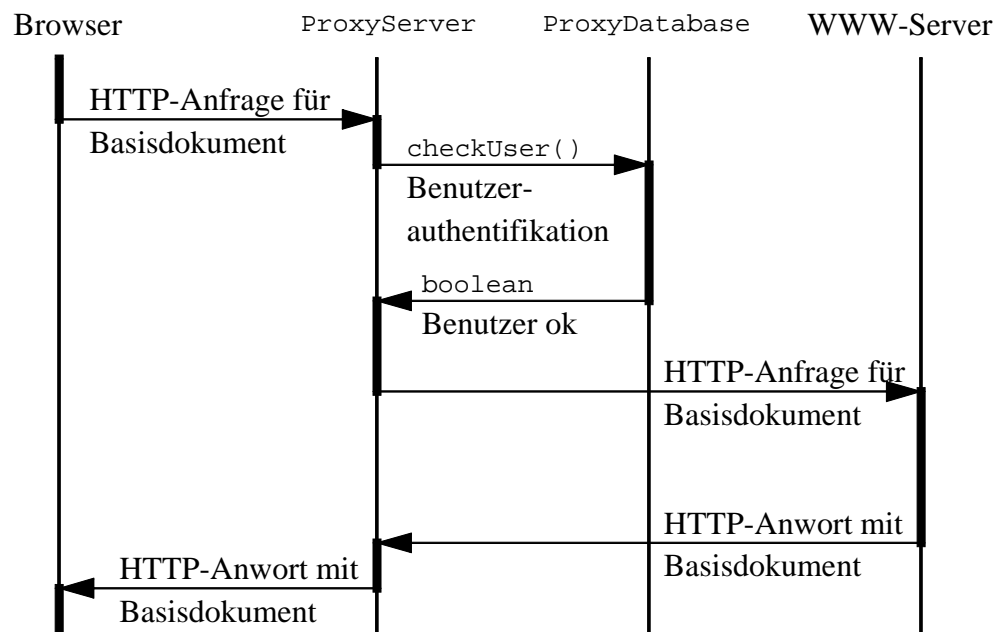
- `getControl()`, erzeugt Kontrollrahmen
- `getAnnotation()`, erzeugt Annotationsschichten
- `getRecorder()`, erzeugt Fenster zur Eingabe neuer Annotationen

## 7.5 Ablaufdiagramme

Die folgenden Ablaufdiagramme zeigen die Interaktion zwischen den Klassen des Annotationsproxy und den anderen WWW-Komponenten. Es werden zwei Fälle gezeigt. In Abbildung 19 ist eine Anfrage für ein Dokument dargestellt, das dynamisch vom Annotationsproxy erzeugt wird. Abbildung 20 zeigt den Fall, wo das Basisdokument angefragt wird, um es unverändert in das Dokumentenmodell aus Kapitel 8.1 zu integrieren. Die Methodenaufrufe sind durch ihren Methodennamen und die Semantik des Aufrufs dargestellt. Der Rückgabewert ist durch den Datentyp und seinen Inhalt beschrieben.



**Abbildung 19 : Ablaufdiagramm für dynamisch erzeugtes Dokument.** Das Ablaufdiagramm zeigt die Interaktion zwischen den Klassen des Annotationsproxy bei einer Anfrage für ein dynamisch erzeugtes Dokument. Bei Methodenaufrufen ist der Name der Methode und ihre Bedeutung gegeben. Der Rückgabewert ist durch den Java Datentyp und seinen Inhalt beschrieben.



**Abbildung 20 : Ablaufdiagramm für Basisdokument.** Das Diagramm zeigt die Interaktionen bei der Anforderung des Basisdokuments, das unverändert in das Dokumentenmodell integriert wird. Nach der Benutzerauthentifikation fordert der Annotationsproxy das Dokument vom WWW-Server an und sendet es an den Browser.



## 8.0 Implementierung

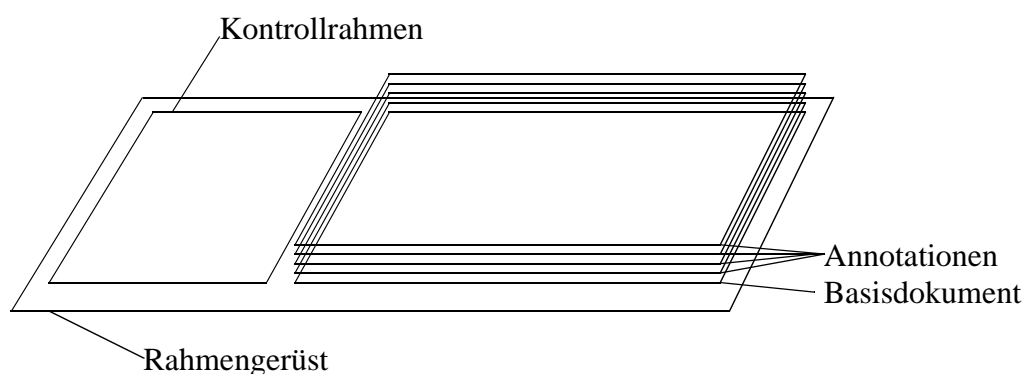
Dieses Kapitel beschreibt einige Implementierungsaspekte des Annotationssystems, das im Rahmen dieser Arbeit entwickelt wurde. Zunächst wird gezeigt, wie ein HTML-Dokument um Annotationen erweitert und im Browser dargestellt wird. Dann werden die Vorgänge im Annotationsproxy beim Laden eines annotierten WWW-Dokuments und beim Erstellen einer neuen Annotation erläutert. Schließlich wird noch das Datenbankmodell dargestellt.

### 8.1 Dokumentenmodell

Um ein annotiertes WWW-Dokument im Benutzerbrowser darzustellen, werden HTML-Rahmen und das Schichtenkonzept von Dynamic HTML verwendet. Die Implementierung von DHTML beim Netscape Navigator und beim Microsoft Explorer unterscheiden sich erheblich. Da der Netscape Navigator für mehr Plattformen erhältlich ist, wurde er als Basis für diese Arbeit ausgewählt. Das heißt, daß sich alle weiteren Angaben auf den Netscape Navigator 4.05 beziehen.

Zu jedem Dokument wird ein Steuerungsfeld für die Kontrolle der Annotationen hinzugefügt. Realisiert wird das durch die Aufteilung des Browserfensters in zwei Rahmen. Im linken ist die Annotationssteuerung zu sehen, im rechten das Basisdokument und die Annotationen. Im Annotationsrahmen ist jede Annotation und das Basisdokument in einer eigenen Schicht dargestellt. Die Verwendung von Schichten ermöglicht das einfache Ein- und Ausblenden oder Bewegen des jeweiligen Dokuments.

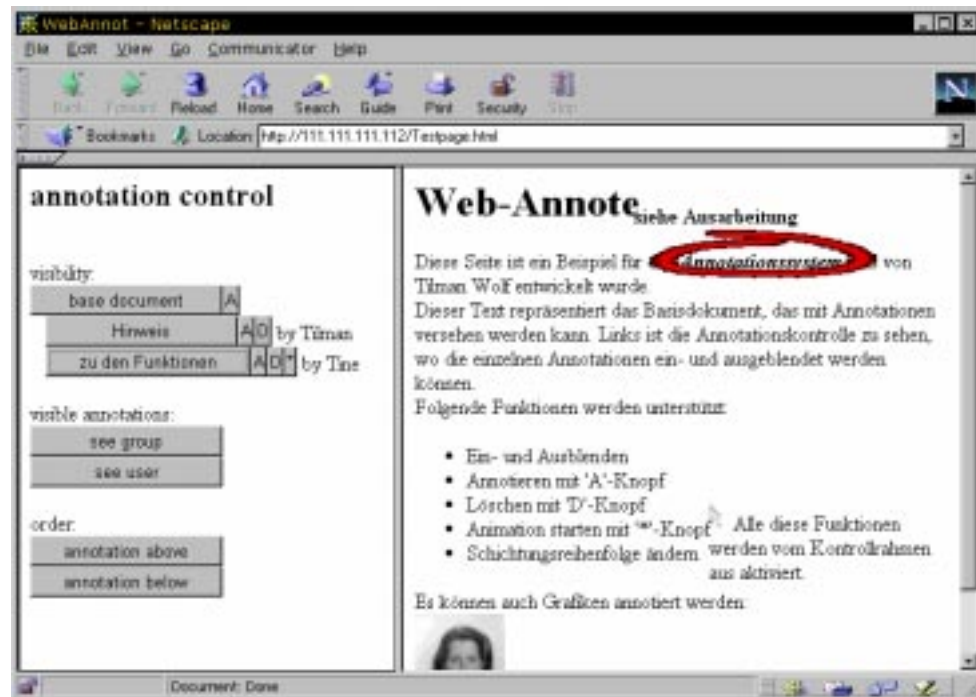
Abbildung 21 zeigt den schematischen Aufbau eines annotierten Dokuments.



**Abbildung 21 : Darstellung eines annotierten WWW-Dokuments.** Das Browserfenster ist in zwei Teile aufgeteilt. Der Kontrollrahmen enthält Steuerelemente für die Annotationen. Der andere Rahmen ist aus mehreren Schichten zusammengesetzt. Das Basisdokument und jede Annotationen wird in einer eigenen Schicht dargestellt.

## 8.2 Darstellung von annotierten WWW-Dokumenten

Wie das oben beschriebene Dokumentenmodell im Browser dargestellt wird, ist in Abbildung 22 zu sehen. Der rechte Rahmen enthält das Dokument und die Annotationen. Im linken Rahmen können Annotationen sichtbar gemacht oder neue Annotationen hinzugefügt werden.



**Abbildung 22 : Darstellung des annotierten WWW-Dokuments im Browser.** Das Basisdokument mit zwei Annotationen ist im rechten Rahmen des Browserfensters zu sehen. Links können die jeweiligen Annotationen sichtbar gemacht oder neue Annotationen hinzugefügt werden können.

### 8.2.1 Kontrollrahmen

Im Kontrollrahmen stehen dem Benutzer folgende Funktionen zur Verfügung:

- Ein- und Ausblenden der Annotationen und des Basisdokuments
- Hinzufügen einer neuen Annotation
- Löschen einer Annotation
- Starten der Animation einer Annotation
- Ändern der Schichtungsreihenfolge um Sichtbarkeit zu verbessern

Die Hierarchie der Annotationen wird dadurch dargestellt, daß die Titelknöpfe der Annotationen entsprechend der Hierarchiestufe eingerückt werden. Eine neue Annotation wird immer eine Stufe tiefer eingefügt als das Dokument, auf das sie sich bezieht.

Das Ein- und Ausblenden wird dadurch erreicht, daß der Benutzer auf den Knopf mit dem Titel der Annotation klickt. Wenn die Annotation sichtbar gemacht wird, springt der Dokumentenrahmen automatisch an die Stelle der Annotation. Die Möglichkeit Dokumente auszublenden ist wichtig, da es beispielsweise vorkommen kann, daß sich zwei Verweise überlappen. Der Browser kann beim Anklicken dann nicht entscheiden, welcher der beiden gemeint ist. Durch Ausblenden kann der nicht gewünschte Verweis entfernt werden.

Hinter dem Titel jeder Annotation ist ein 'A'-Knopf vorhanden, mit dem eine neue Annotation zum jeweiligen Dokument hinzugefügt werden kann.

Wenn der Benutzer entsprechend seiner Zugriffsrechte auch eine Annotation löschen kann, gibt es auch einen 'D'-Knopf. Falls eine Annotation vom Benutzer nicht gelöscht werden darf, fehlt der Knopf.

Bei animierten Annotationen gibt es auch einen '\*'-Knopf. Beim Anklicken wird die entsprechende Annotation sichtbar gemacht und bewegt sich.

Schließlich gibt es noch Knöpfe, mit denen der Benutzer auswählen kann, ob er alle Annotationen, die seiner Gruppe oder nur seine eigenen sehen möchte. Außerdem kann er sich das Basisdokument entweder als oberste oder unterste Schicht anzeigen lassen. Dadurch kann der Benutzer beeinflussen, wie die Dokumente angeordnet werden, und sie optimal darstellen lassen.

## 8.2.2 Dokumentenrahmen

In diesem Rahmen wird das Dokument zusammen mit den Annotationen dargestellt. Die Position einer Annotation wird über die x- und y-Koordinate relativ zur linken oberen Ecke des Basisdokuments definiert. Das setzt aber voraus, daß das Basisdokument immer mit gleicher Länge und Breite dargestellt wird. Deshalb ist die Breite des Dokumentenrahmens systemweit einheitlich festgelegt.

Beim Erzeugen einer neuen Annotation kann mit der Maus die Position festgelegt werden. Genauso wird bei animierten Annotationen die Mauszeigerbewegung im Dokumentenrahmen aufgezeichnet.

## 8.2.3 Erzeugen neuer Annotationen

Der Benutzer kann eine neue Annotation erzeugen, indem er den 'A'-Knopf des entsprechenden Basisdokuments anklickt. Dann kann er die Position der Annotation im Dokumentenfenster wählen. Bei animierten Annotationen wird der Bewegungsablauf der Maus aufgenommen. Der Annotationstext wird schließlich in einem entsprechenden Fenster eingegeben. Abbildung 23 zeigt das Eingabefenster.

Das HTML-Formular enthält Eingabefelder für den Titel und den Text der Annotation. Der Annotationstext kann mit HTML-Markierungen formatiert werden. Wie die Annotation dann im Browser dargestellt wird, kann mit der Vorschau angesehen

werden. Auch die Breite der Annotation kann verändert werden, um eine optimale Präsentation der Annotation zu ermöglichen. Die Zugriffsrechte auf die Annotation bezüglich der Sichtbarkeit und Löschbarkeit können auch festgelegt werden.



**Abbildung 23 : Eingabefenster für neue Annotationen.** Der Titel und der Text der Annotation werden in das HTML-Formular eingetragen. Der Text kann mit HTML-Befehlen formatiert werden. Es können auch verschiedene Marker und Zeiger verwendet werden. Außerdem können die Zugriffsrechte und die Breite der Annotation festgelegt werden. Mit der Vorschau ist es möglich zu sehen, wie die komplette Annotation vom Browser dargestellt wird.

Am Ende der Eingabe werden dann alle Daten an den Proxy übertragen und in der Annotationsdatenbank gespeichert.

### 8.3 Abläufe im Proxy

Der Annotationsproxy erfüllt folgende Funktionen :

- Generieren der HTML-Dokumente, die Annotationen und Steuerungselemente enthalten
- Anfragen der Basisdokumente von den Webservern
- Unterscheiden zwischen Anfragen für annotierbare WWW-Dokumente und nicht annotierbare Daten
- Verwalten der Annotationsdatenbank

### 8.3.1 Anfragen eines WWW-Dokuments

Wenn der Proxy eine Anfrage für ein normales HTML-Dokument erhält, werden die folgenden Abläufe in Gang gesetzt.

Der Proxy sendet zunächst das Rahmengerüst, das Verweise auf den Kontrollrahmen und den Rahmen mit den Annotationen und dem Basisdokument enthält. Diese Verweise werden dann vom Browser automatisch nachgeladen. Das bedeutet, daß der Proxy weitere Anfragen erhält, die er aber von gewöhnlichen Anfragen unterscheiden können muß, um nicht wieder das Rahmengerüst zu senden. Die Unterscheidung kann durch Verwendung eines speziellen URL erreicht werden, wie es unten beschrieben wird.

Für das angefragte WWW-Dokument wird eine entsprechende Anfrage an die Annotationsdatenbank gesendet. Abhängig von den Wünschen des Benutzers werden dann die Annotationen zusammengestellt. So können entweder nur die Annotationen des Benutzers, die der Gruppe oder alle ausgewählt werden.

In den Kontrollrahmen wird dynamisch der HTML-Code für die Knöpfe und JavaScript-Funktionen eingefügt und an den Browser übertragen. In den Dokumentenrahmen wird das Basisdokument praktisch unverändert übernommen. Außerdem werden die Schichten mit den Annotationstexten generiert und hinzugefügt.

Damit sind das Dokument und die Annotationen übertragen und können komplett vom Browser dargestellt werden.

### 8.3.2 Spezielle URL

Wie bereits erwähnt wurde, muß der Proxy zwischen den initialen Anfragen für ein Dokument und den Anfragen für den Kontroll- und Dokumentenrahmen unterscheiden können. Deshalb wird ein spezielles Schema für diese URL verwendet.

Die initialen Anfragen werden mit dem URL des Basisdokuments gesendet. Eine Änderung dieses URL ist schon deshalb nicht möglich, weil Verweise von anderen Dokumenten keinen speziellen URL verwenden können.

Anfragen für Kontroll- und Annotationsdaten werden dann direkt an den Proxy gesendet. Das heißt, daß die Serverkomponente des URL die IP-Adresse des Proxy ist. Die Pfadkomponente enthält die Angabe, welches Dokument angefragt wird und auf welches Basisdokument sich die Anfrage bezieht.

Als Beispiel wird der spezielle URL für einen Kontrollrahmen angegeben. Das Basisdokuments hat den URL 'http://www.webserver.com/document.html', der Annotationsproxy hat die IP-Adresse 'www.proxy.com' und der Kontrollrahmen wird durch die Pfadkomponente '.../control/...' identifiziert:

'http://www.proxy.com/control/www.webserver.com/document.html'

Der konstruierte URL ist eindeutig, da ja schon der URL des Basisdokuments ein-

deutig war. Die Verwendung der IP-Adresse des Webservers in der Pfadkomponente ist zulässig. Auch die Verwendung einer möglichen Portnummer, die durch ':' abgetrennt ist, entspricht den Vorgaben für URL.

Entsprechend wird der URL für den Annotationsrahmen mit '.../annotation/...', der für das Eingabefenster für neue Annotationen mit '.../recorder/...' und so weiter konstruiert.

Die Verwendung von diesen speziellen URL hat auch Vorteile für die Benutzerverwaltung. Das HTTP-Protokoll unterstützt eine einfache Authentifikation. Dabei wird beim Zugriff auf geschützte Dokumente ein Benutzername und ein Paßwort verlangt. Der Browser überträgt dann automatisch diese Angaben mit jeder Anfrage an den entsprechenden Server. Da alle speziellen URL mit der IP-Adresse des Proxys beginnen, muß der Benutzer nur beim ersten Zugriff auf den Annotationsproxy seinen Namen und das Paßwort angeben.

### 8.4 Datenbank

Der Annotationsproxy ist für die Verwendung mit einer relationalen Datenbank ausgelegt. Der Zugriff erfolgt über die ODBC (Open Database Connectivity) Schnittstelle. Diese wird von der JDBC (Java Database Connectivity) Schnittstelle angesprochen, so daß ein einfacher Zugriff mit Java möglich ist. Alle Datenzugriffe werden mit SQL (Structured Query Language) durchgeführt.

Die folgenden Tabellen zeigen das relationale Datenmodell, das für die Datenverwaltung des Annotationsproxy verwendet wird. Es ist der Name der jeweiligen Tabelle und ihrer Spalten gegeben. Die Primärschlüssel sind mit 'P' versehen. Unter dem Namen jeder Spalte ist auch der SQL-Datentyp vermerkt.

Die Tabelle 'AnnotationTable' enthält alle Annotationen. Jede Annotation wird bei der Erstellung mit einer eindeutigen Nummer versehen. Der URL der Annotation ist in den Server- und Pfadteil getrennt. Die Spalte 'RefersTo' enthält die Nummer der Annotation, auf die sich die jeweilige Annotation bezieht. Falls sich die Annotation auf das Basisdokument bezieht, wird hier eine Null eingetragen. Außerdem werden der Name des Autors, die Sichtbarkeits- und Löscharkeitsrechte und der Zeitpunkt der Erstellung gespeichert. Die übrigen Spalten sind für die Darstellung der Annotation wichtig. Die Spalten 'Animation' und 'Data' enthalten die Animationsdaten und den Annotationstext an sich. Die Größe von 32 Kilobyte sollte ausreichend für praktisch alle Text-, Mauszeiger- und Markerannotationen sein. Bei Bedarf kann dieser Wert auch vergrößert werden.

**Tabelle 2: AnnotationTable**

AnnID (P) INT	Server CHAR(50)	Path VARCHAR (200)	RefersTo INT	User VARCHAR (8)	Visible CHAR(1)	Deletible CHAR(1)
Time TIME- STAMP	Title VARCHAR (20)	Xpos INT	Ypos INT	Width INT	Animation VARCHAR (32768)	Data VARCHAR (32768)

Die Tabelle 'UserTable' speichert die Daten aller Benutzer des Annotationsproxys. Neben dem Loginnamen, der Gruppenzugehörigkeit und dem Paßwort, werden auch der vollständige Name und die Emailadresse gespeichert, um Rückfragen an den Autor einer Annotation zu ermöglichen. Der Zeitpunkt des letzten Logins wird vermerkt, um beispielsweise nur die Annotationen anzuzeigen, die seit der letzten Benutzung erstellt worden sind.

**Tabelle 3: UserTable**

User (P) VARCHAR (8)	Group VARCHAR (8)	Password VARCHAR (20)	RealName VARCHAR (50)	Email VARCHAR (50)	LastLogin TIME- STAMP

Die Tabelle 'GroupTable' enthält eine informelle Beschreibung aller Gruppen. Damit kann die Gruppenzugehörigkeit eines neuen Benutzers einfach ausgewählt werden.

**Tabelle 4: GroupTable**

Group (P) VARCHAR(8)	Description VARCHAR(50)

## 8.5 Integration weiterer Medientypen

Das Annotationssystem ist nicht auf die bisher implementierten Medientypen für Annotationen beschränkt. Wenn neben Text, Verweisen, Markern und Mauszeigerbewegungen ein weiteres Medium benötigt wird, kann es auf folgende Weise in das

System integriert werden:

- Anpassen der Datenbank, so daß Medientyp verwaltet werden kann
- Entwicklung einer Aufnahmesoftware für dieses Medium, die auch die Daten an den Server zur Speicherung übertragen kann
- Vergeben einer speziellen URL für dieses Medium, so daß die Klasse `Proxy-Server` erkennen kann, wenn Daten von diesem Typ angefordert werden
- Anpassung der Klasse `HTMLGenerator`, so daß HTML-Code für die Einbindung des Mediums ins Dokumentenmodell generiert werden kann

## 8.6 Bewertung

Das Annotationssystem, das in dieser Arbeit entwickelt wurde, ist ein voll funktionsfähiges System. Es erfüllt alle Anforderungen, die in Kapitel 3 herausgearbeitet wurden:

- Die geforderte einfache Integration in die Arbeitsumgebung ist dadurch gegeben, daß der Benutzer keine Software installieren muß. Die einzige Modifikation ist das Verändern der Proxyangaben im Browser.
- Das System ist multiuserfähig und erlaubt verschiedene Sichten auf die gespeicherten Annotationen. Dadurch ist eine Kooperation durch gemeinsame Nutzung von Information möglich.
- Das Annotationssystem unterstützt multimediale Annotationen. Die hier implementierten Text, Hyperlink, Marker und Mauszeigerbewegungen sind nur Beispiele für die Medien, die verwendet werden können.
- Die Annotationen und das Basisdokument können klar unterschieden werden. Die semantische Beziehung zwischen den einzelnen Dokumenten ist im Kontrollrahmen ersichtlich.
- Die benutzerdefinierte Wiedergabe ist insofern implementiert, daß der Benutzer animierte Annotationen zu einem beliebigen Zeitpunkt starten kann.

Es gibt einige Erweiterungen, deren Implementierung den Rahmen dieser Arbeit sprengen würden. Sie sind aber für den praktischen Einsatz dieses Annotationssystems von Bedeutung:

- Unterstützung des Microsoft Explorers und Erweiterung der annotierbaren Dokumente auf HTML-Seiten, die selbst Rahmen und Schichten verwenden
- Implementierung einer Schnittstelle zum DIANE System
- Implementierung weiterer Medientypen für Annotationen, eventuell unter Verwendung von DIANE
- Implementierung einer graphischen Benutzerschnittstelle für den Annotationsproxy

## **9.0 Zusammenfassung und Ausblick**

Zum Abschluß ist in diesem Kapitel eine Zusammenfassung der Arbeit und ein Ausblick auf weitere Forschungsthemen gegeben.

### **9.1 Zusammenfassung**

Ziel dieser Arbeit war die Entwicklung eines Annotationssystems für WWW-Dokumente. Existierende Annotationssysteme wurden auf Funktionalität und Nutzbarkeit hin untersucht. Ihre Vor- und Nachteile wurden als Grundlage für die Spezifikation eines neuen Systems verwendet.

Besondere Bedeutung wurde der einfachen Integration des Systems in das WWW und in die Arbeitsumgebung des Benutzers zugemessen. Deshalb wurde ein Proxyserver entwickelt, der die Annotationsfunktion für andere Internetkomponenten transparent implementiert. Auf diese Weise können die Annotationen ohne spezielle Software im Browser des Benutzers dargestellt und verfaßt werden.

Die Implementierung basiert auf der Verwendung von bewährten Internetprotokollen und Datenformaten, wie HTTP, HTML, JavaScript und Dynamic HTML. Der entstandene Prototyp ist voll funktionsfähig und zeigt die Leistungsfähigkeit des entwickelten Annotationssystems.

### **9.2 Ausblick**

Auf der Basis dieser Arbeit sollten in Zukunft noch einige Themen untersucht werden, um das entwickelte System weiter zu verbessern.

Interessant und wichtig ist die automatische Behandlung von Veränderungen in Dokumenten und Annotationen. Eine Lösung ist besonders für die oft kurzlebigen Informationen auf dem WWW von Bedeutung.

Außerdem ist die Zusammenfassung der Annotationen in einem gemeinsamen Schema für Position und Zeit hilfreich. Nur so können zeitliche Abhängigkeiten zwischen Annotationen dargestellt werden. Dies läßt sich durch eine Integration in das DIANE System erreichen, bei dem solche Funktionen bereits implementiert sind. Ein anderer Ansatz ist die Verwendung von SMIL-Dokumenten.

Schließlich sollte das System von Benutzern getestet werden. Nur so können zusätzliche Benutzeranforderungen gefunden werden, damit eine erfolgreiche Verwendung des Annotationssystems im Arbeitsalltag gewährleistet ist.



## **Anhang A : Annotationssysteme im WWW**

Die folgenden WWW-Adressen enthalten Informationen über die verschiedenen Annotationssysteme.

### **In dieser Arbeit beschriebene Systeme:**

Annotator: <http://www.futuresoft.com/>

ComMentor: <http://hci.stanford.edu/commentor/>

CoNote: <http://dri.cornell.edu/pub/davis/Annotation/annotation.html>

Mole: <http://www.cs.strath.ac.uk/~dave/Mole/>

Futplex: <http://gewis.win.tue.nl/applications/futplex/>

HypeNews: <http://www.hypernews.org/HyperNews/get/hypernews.html>

### **Weitere Annotationssysteme:**

FRESCO: <http://www-pcd.stanford.edu/FRESCO/>

W3 Document Annotator: <http://eliot.uni.ch:8085/docs/wda-article.html>

NetQ: <http://netq.rowland.org>

Menno Bot: <http://www.evitech.fi/~mennob/password.pl/annotationssetup>

Active Annotations: <http://www.eurecom.fr/~hummes/papers/Web4Groups/>



## Anhang B : Bibliographie

- [Ber94] Berners-Lee, T., Masinter, L., McCahill, M. (1994). RFC 1738: Uniform Resource Locators.
- [Ber96] Berners-Lee, T., Fielding, R., Frystyk H. (1996). RFC 1945: Hypertext Transfer Protocol - HTTP/1.0, Network Working Group.
- [Fie98] Fielding, R., Gettys, J., Mogul, J. C., Frystyk, H., Masinter, L., Leach, R., Berners-Lee, T. (1998). Hypertext Transfer Protocol - HTTP/1.1, Working Draft, HTTP Working Group.
- [Ben97a] Benz, H., Fischer, S., Mecklenburg, R., Wenger, A. (1997). Application Output Recording for Instant Authoring in a Distributed Multimedia Annotation Environment. IPVR, Universität Stuttgart.
- [Ben97b] Benz, H., Fischer, S., Dermler, G., Mecklenburg, R. (1997). DIANE - Hypermedia Documents in a Distributed Annotation Environment. IPVR, Universität Stuttgart.
- [Ben97c] Benz, H., Fischer, S., Mecklenburg, R. (1997). Architecture and Implementation of a Distributed Multimedia Annotation Environment: Practical Experiences using Java. IPVR, Universität Stuttgart.
- [Hau98] Hauser, J. (1998). Modellierung von Annotationen auf veränderlichen Hypermediadokumenten. Diplomarbeit, Fakultät Informatik, Universität Stuttgart.
- [Hos98] Hoschka, P., Bugaj, S., Bulterman, D., Hardman, L., Jansen, J., Lanphier, R., Layaida, N., Marsh, J., Rao, A., Rutledge, L., ten Kate, W., van Ossenbruggen, J., Vernick, M., Yu J. (1998). Synchronized Multimedia Integration Language, Working Draft, W3C.
- [ISO8879] International Standard 8879:1986. Standard Generalized Markup Language (SGML). International Organization for Standardization.
- [Rag95] Raggett, D. (1995). HyperText Markup Language Specification Version 3.0, W3C.
- [Rag97] Raggett, D., Le Hors, A., Jacobs, I. (1997). HTML 4.0 Specification, W3C.



## **Anhang C : Erklärung**

Ich versichere, daß ich diese Arbeit  
selbständig verfaßt und  
nur die angegebenen Hilfsmittel  
verwendet habe.

---

Datum, Unterschrift