Suchdienste im World-Wide Web - Anforderungen und Perspektiven

Ute Masermann
Medizinische Universität zu Lübeck
Institut für Informationssysteme
Osterweide 8
23562 Lübeck
email: masermann@informatik.mu-luebeck.de

Zusammenfassung

Durch das World-Wide Web (kurz WWW oder Web) ist es auf einfache Art möglich, Zugang zu Informationen zu erhalten, die über das ganze Internet verteilt sind. Durch das große und schnell wachsende Informationsangebot im WWW hat es sich jedoch als notwendig erwiesen, über die reine Navigation hinaus Suchwerkzeuge zur Verfügung zu stellen.

Ziel dieses Beitrags ist es, nach einer Einführung in die Problematik Anforderungen an Suchdienste zu formulieren. Anschließend werden bestehende Suchdienste vorgestellt und in Hinblick auf diese Anforderungen untersucht. Basierend auf diesen Ergebnissen sollen zuletzt Perspektiven für zukünftige Entwicklungen aufgezeigt werden.

1 Einführung

Das World-Wide Web ist ein Internet-Dienst, der mit Hilfe eindeutiger URLs (Uniform Resource Locators) Dateien auf den WWW-Servern lokalisiert und diese dem Benutzer zur Verfügung stellt. Bei diesen Dateien handelt es sich überwiegend um strukturierte Dokumente, die in der Sprache HTML (Hypertext Markup Language) verfaßt sind. Neben den Strukturierungselementen wie Markierung von Listen, Überschriften oder bestimmter Textattribute wie „hervorheben" besteht eine wichtige Eigenschaft von HTML darin, sogenannte *Hyperlinks* zu definieren. Hyperlinks werden im Browser (dem WWW-Client) sichtbar gemacht und verweisen auf andere Dokumente. Diese können an beliebiger Stelle im Internet liegen und werden durch Anwählen im Browser (meistens durch Mausclick) geladen und angezeigt bzw. je nach Datentyp wie Audio/Video verarbeitet. Sie können ihrerseits wieder Hyperlinks enthalten, die weiterverfolgt werden können, so daß man auf diese Art und Weise durch das Internet navigieren kann.

Außer der Eigenschaft, durch statische Verknüpfungen bereits vorhandene Dokumente zu laden, bietet das WWW die Alternative, durch Hyperlinks CGI (Common Gateway Interface)–Programme auf dem Server ausführen zu lassen. Diese Programme bekommen (z.B. interaktiv über HTML-Formulare) Parameter und Daten zugeführt und können beliebige Daten, insbesondere auch HTML-Dokumente zurückgeben. So können dynamisch HTML-Seiten (inklusive Hyperlinks) generiert werden, die aktuelle Informationen wie Statistiken enthalten oder die Daten aus anderen Informationssystemen für das Web aufbereiten.

Um gezielt auf Informationen zuzugreifen, muß demnach der URL der gewünschten Seite bekannt sein oder ein Startdokument, das durch Links mit den interessanten Informationen verbunden ist (und das Wissen, daß dies der Fall ist), so daß das gewünschte Dokument durch Navigation erreicht werden kann. Da durch die starke Expansion des WWW und seine dezentrale Verwaltung keine der beiden Bedingungen zufriedenstellend erfüllt werden kann, müssen geeignete Suchdienste entworfen werden.

Die Anforderungen an solche Dienste werden im nächsten Abschnitt formuliert. Danach werden verschiedene Suchwerkzeuge vorgestellt, wobei es sich um Suchmaschinen, Datenbankanbindungen ans WWW und neuentwickelte WWW-Anfragesprachen handelt. Im letzten Abschnitt werden nach einer Zusammenfassung Perspektiven aufgezeigt.

2 Anforderungen an Suchdienste

Zunächst stellt sich die Frage, was es bedeutet, das „gesamte“ WWW durchsuchen zu wollen. Handelt es sich um sämtliche im Internet verfügbaren Dokumente? Sind es darüber hinaus diejenigen, die auf Servern liegen, die kurzfristig ausgefallen sind? Sind es solche, die *momentan* von einem Startpunkt aus zu erreichen sind, wobei hier Ausfälle nicht nur einzelner Server sondern von Teilnetzen zu berücksichtigen sind? Da das WWW zu groß ist, um zu jedem Suchzeitpunkt auf sämtliche Dokumente zugreifen zu können, müssen hier bei der Realisierung eines konkreten Suchdiensts sicher Abstriche gemacht werden. Trotzdem soll diese Anforderung in ihrer weitesten Interpretation aufgenommen werden, d.h. eine WWW-Suche soll alle Dokumenten auf allen Servern berücksichtigen. In diese Problematik fällt auch die Suche nach dynamisch erzeugten Web-Seiten, die u.U. nicht dauerhaft auf den Servern liegen, aber dennoch Informationen darstellen, die über das WWW zur Verfügung stehen.

Selbstverständlich sollen keine veralteten URLs, sondern aktuell vorhandene zurückgegeben werden. Deshalb muß die dynamische Serverentwicklung (Umstrukturierung/Umbenennung von Dokumenten) berücksichtigt werden. Was heute noch in einer bestimmten Stelle im Dokumentbaum eines Servers zu finden ist, kann morgen schon nicht mehr vorhanden sein bzw. kann zwar die URL als solche weiterhin existieren, ihr Inhalt kann sich aber vollständig geändert haben. Aus diesem Grund wird die Forderung nach konsistenten Ergebnissen aufgestellt.

Die Ergebnisse sollen insofern korrekt sein, als daß sie die für den Benutzer relevante Informationen enthalten. Das bedeutet beispielsweise, daß ein Dokument nur dann als relevant zum Thema „Datenbanken“ eingestuft wird, wenn es nicht nur eine gleichlautende fettgedruckte Überschrift enthält, sondern wenn auch der Inhalt des Dokuments sich um dieses

Thema dreht. Konkret handelt es sich um die Frage: Wie kann „Relevanz“ im Kontext von HTML definiert werden? Auch wenn diese Frage von den einzelnen Suchdiensten unterschiedlich beantwortet wird und es noch keine allgemeingültige Definition für Relevanz eines Dokuments bezüglich einer Anfrage gibt, soll diese Forderung mit aufgenommen werden, da ein intuitives Verständnis vorausgesetzt werden kann.

Da es sich beim WWW um ein dezentrales System handelt und die einzelnen Server autark sind, soll von einem Suchdienst möglichst wenig in diese Autarkheit eingegriffen werden. Daraus folgt, daß z.B. das Hypertext-Transfer-Protokoll nicht geändert werden soll oder keine zusätzliche HTML-Markierungen in eine Vielzahl der Dokumente eingefügt müssen.

Daneben sollen noch Anforderungen wie „schnelle Antwort“, „leichte Bedienbarkeit“, „geringe Netzbelastung“ erfüllt werden, auf die aber im folgenden nicht näher eingegangen wird.

In Kurzform handelt es sich demnach um folgende Anforderungen, auf die die vorgestellten Dienste untersucht werden sollen:

1. Suche im „gesamten“ WWW,
2. Lieferung von konsistenten Ergebnissen,
3. Relevanz der Ergebnisse sowie
4. wenig Eingriffe in die dezentrale und autarke Organisation des WWW.

3 Suchmaschinen

Als erster Suchdienst werden die sehr populären Suchmaschinen (oder auch *Indexserver*) beschrieben. Dabei handelt es sich um *Robots*, die – ausgehend von einem Startdokument – automatisch die von diesem Dokument weiterführenden Hyperlinks durchsuchen. Die gefundenen Dokumente werden analysiert und katalogisiert. Dabei gibt es sowohl Unterschiede bei der Suche nach Dokumenten als auch bei deren Analyse. Zum einen werden möglichst viele Server besucht und jeweils nur wenige Dokumente registriert, zum anderen werden nur wenige Server besucht, deren Dokumentbaum aber vollständig katalogisiert. Somit werden von unterschiedlichen Maschinen – *WebCrawler*, *Lycos* oder *JumpStation*[1] seien als Beispiele genannt – verschiedene Subnetze des WWW katalogisiert. Diese Kataloge werden regelmäßig aktualisiert. Auch beim Aktualisieren gibt es verschiedene Vorgehensweisen, die vom völligen Neuerstellen des Katalogs bis zum Prüfen, ob die betreffenden URLs sich geändert haben, reichen.

Bei der Analyse der Dokumente werden beispielsweise lediglich die Titel der HTML-Seiten und Schlagworte extrahiert. Eine andere Strategie besteht darin, sämtliche Schlagworte des Textes aufzunehmen oder solche, die durch HTML-Markierungen hervorgehoben sind. Auch die Auswertung der Hyperlinks kann Bestandteil einer Analyse sein. Eine Darstellung von Analysealgorithmen ist in [10] zu finden.

[1] Die URLs lauten:
`http://webcrawler.com`, `http://lycos.cs.cmu.edu` bzw. `http://js.stir.ac.uk/jsbin/jsii`

Beim Nutzen einer solchen Suchmaschine wird der Katalog, auf dem er basiert, nach den gewünschten Schlagworten durchsucht. Es werden alle URLs zurückgeliefert, in denen diese Schlagworte – gegebenenfalls durch UND, ODER oder andere einfache Verknüpfungen kombiniert – vorkommen. Manche Dienste bewerten diese URLs danach, wie oft die Schlagworte im Titel oder im Text vorkommen und geben sie in dieser Reihenfolge an. Für eine genauere Beschreibung sei auf [8] verwiesen.

Die genannten Systeme liefern bei der gleichen Anfrage unterschiedliche URL-Listen zurück, da sie – wie oben erwähnt – verschiedene Subnetze durchsucht haben und damit nicht auf den gleichen Katalogen basieren. Oft sind diese Listen u.U. zu lang, um sie als Suchergebnis effizient nutzen zu können, auch wenn durch Angabe vieler Schlagworte versucht wurde, die Suche einzugrenzen. Ebenso ist nicht zu erkennen, ob einzelne Dokumente zueinander durch Hyperlinks in Beziehung stehen.

Über einfache Suchmaschinen hinaus geht u.a. der *MetaCrawler*[2] [9]. Er dient als allgemeine Schnittstelle zu den einzelnen Indexservern, die unterschiedliche Oberflächen haben, und wertet die gemeinsamen Ergebnisse aus, um qualitativ bessere URL-Listen zu erzeugen. Auch durch Entwicklung spezieller Maschinen für bestimmte Anforderungen (z.B. Suchen persönlicher Homepages[3]) wird versucht, die Suche individueller zu gestalten.

Insgesamt zeigt sich, daß bei Suchmaschinen die Angabe der Suchkriterien rein durch Schlagworte zu schwach ist, um aus dem großen Angebot die richtige Information herauszufiltern. Viele WWW-Server sind außerdem in keinem der Indexserver registriert, da sie z.B. nicht stark genug mit anderen „verlinkt“ sind und deshalb nicht von den Robots erfaßt werden. Darüber hinaus sind einige URLs der Kataloge veraltet, da die Server nur in gewissen Abständen aktualisiert werden und das WWW sich zu dynamisch verhält, um sie wirklich aktuell halten zu können. Die dynamisch erzeugten Seiten (auf die z.B. nur durch eine CGI-Schnittstelle zugegriffen werden kann) können durch die Suchmaschinen nicht mit erfaßt werden, so daß diese Informationen, die beispielsweise aus Unternehmensdatenbanken stammen, nicht in die Suche integriert werden können.

4 Anbindung von Datenbanksystemen an das WWW

Da das WWW als riesige verteilte Datenbank interpretiert werden kann und für Datenbanken mächtigere Anfragesprachen existieren als es eine reine Schlagwortsuche sein kann, die im vorigen Abschnitt beschrieben wurde, bietet es sich an, WWW-Server mit Datenbanken zu verwalten und über Datenbankmechanismen Suchmöglichkeiten zur Verfügung zu stellen. Bei Datenbanksystemen ist durch das Schema eine stärkere Semantik vorgegeben als es bei unstrukturierten Daten oder auch HTML-Dokumenten der Fall ist. Deshalb liefert eine Suche in einer Datenbank qualitativ bessere Ergebnisse zurück als eine reine Schlagwortsuche in Katalogen von Indexservern.

Aus diesen Gründen ist es von Vorteil, eine Datenbank mit einer CGI-Schnittstelle als WWW-Server zu betreiben, wie es verschiedene Anbieter kommerzieller Datenbanksysteme bereits tun [1], [7]. So können alle Informationen, die in der Datenbank enthalten sind,

[2]URL: `http://metacrawler.cs.washington.edu`
[3]Ahoy! The Homepage Finder: `http://www.cs.washington.edu/research/ahoy`

dynamisch als HTML-Dokumente dargestellt werden. Die Benutzerschnittstelle im WWW-Browser kann wie eine 4GL-Schnittstelle für Abfragen benutzt werden.

Der Vorteil eines solchen WWW-Servers liegt nicht nur in den guten Suchmöglichkeiten, sondern auch in der einfachen Pflege der Daten. Bei einem Datenbanksystem wird es durch die Definition von Integritätsbedingungen und das zentrale Halten der Information vermieden, daß Links „ins Leere“ zeigen.

Der Nachteil ist hier aber offensichtlich: Es werden lediglich Informationen *eines* Servers verwaltet, der Rest des Internets kann so nicht durchsucht werden. Beispielsweise ist eine solche Lösung für eine einzelne Firma als „Intranet“ sicher sinnvoll, aber um Informationen im gesamten WWW zu suchen, genügt es nicht, einzelne – ansonsten autarke – Server mit einem Datenbanksystem zu verwalten. Die Alternative, von allen Servern zu fordern, daß sie mit einem (wie auch immer gearteten) Datenbanksystem arbeiten und das Schema als Schnittstelle für andere offen darlegen, ist nicht durchzusetzen.

5 WWW-Anfragesprachen

Da sowohl Indexserver als auch einzelne Datenbankanbindungen nicht als Suchwerkzeug ausreichen, werden Ansätze entwickelt, die versuchen, das Web selbst als Datenbank zu behandeln und mit entsprechenden Sprachen zu durchsuchen. Eine solche Herangehensweise bietet sich an, da Datenbankanfragesprachen zumindest für relationale Datenbanksysteme gut erforscht sind und somit eine Grundlage für dieses neue Anwendungsgebiet bieten können.

In diesem Abschnitt werden zunächst drei Ansätze vorgestellt, die das Web mit deklarativen Sprachen durchsuchen, danach werden zwei Ideen beschrieben, die durch Spezifikation des Inhalts und der Benutzerschnittstelle von Informationsquellen eine Suche im WWW realisieren.

Als erstes sei hier W3QS (World-Wide Web Query System) von Konopnicki und Shmueli [2] genannt. Sie versuchen in ihrem System, das WWW mit einer SQL-ähnlichen Sprache (W3QL) zu durchsuchen und entsprechende Listen von URLs als Ergebnis zu liefern. In ihrem System wird ausgenutzt, daß die meisten Dateien im WWW semistrukturierte Daten enthalten (z.B. HTML, LaTeX). Ohne die vollständige Sprache zu beschreiben, soll an einem Beispiel das Prinzip von W3QL dargestellt werden. Es sollen alle URLs herausgesucht werden, die von der Seite mit der URL `‘‘http://www.informatik.uni-trier.de/~ley/db’’` aus durch einen direkten Link verbunden sind und in ihrem Titel das Wort „SIGMOD“ enthalten:

```
SELECT n2
FROM n1,l1,n2
WHERE n1.url= ‘‘http://www.informatik.uni-trier.de/~ley/db’’
AND n2.title LIKE ‘‘%SIGMOD%’’;
```

Analog zur From-Klausel eines SQL-Ausdrucks wird hier angegeben, von welchen Knoten und über welche Links im Netz die Webseiten ausgewertet werden sollen. Das Web wird dabei als DAG interpretiert, wobei n_i einen Knoten (= eine Webseite) und l_i eine Kante (=

ein Hypertextlink) darstellt. Somit werden „Knoten" und „Kante" in W3QL einer Relation in SQL gleichgesetzt. Im Beispiel beginnt die Suche bei `n1` und geht über einen Link `l1` zu einer Seite `n2`. In der Where-Bedingung wird die Struktur der betreffenden HTML-Seite ausgewertet, indem `n1` durch seine URL spezifiziert wird und der Titel von `n2` „SIGMOD" enthalten muß. Alle URLs, die diese Bedingung erüllen, werden als Ergebnismenge ausgegeben.

Statt eine bestimmte „Start-URL" anzugeben, können auch die Kataloge der Indexserver genutzt werden, so daß der Suchraum vergrößert wird und der Benutzer keine URL angeben muß. Für die Nutzung der Indexserver wird der Mechanismus des automatischen Formularausfüllens zur Verfügung gestellt, der Fill-Out-Forms – die Benutzerschnittstelle dieser Server – ohne Benutzerinteraktion vervollständigt. Weiterhin können mit UNIX-Kommandos und -Programmen Listen aus dem zurückgelieferten Ergebnis erzeugt werden, die zudem periodisch aktualisiert werden. Diese Ergebnisse werden als Views bezeichnet, wie sie aus den Datenbanken bekannt sind, da sie eine Sicht auf das WWW bieten.

Mit der Auswertung der Struktur innerhalb eines Dokuments bei einer Suche geht dieser Ansatz deutlich über eine reine Schlagwortsuche hinaus. Durch die DAG-Notation in der From-Klausel ist es zudem möglich, auch die Hypertextstruktur des WWW auszuwerten. Als Sprache für Endbenutzer ist sie allerdings zu komplex und zu technisch orientiert, so daß zumindest eine Schnittstelle entworfen werden muß, die dem Anwender wie eine 4GL-Oberfläche zur Verfügung steht.

Mendelzon, Mihaila und Milo beschreiben in [6] WebSQL. Hier wird ein einfaches Datenbankschema für das Web definiert, das jeweils für *Document* und *Link* eine Relation vorsieht. An diese Relationen können mit einer SQL-ähnlichen Sprache Anfragen gestellt werden. Auch hier soll ein Beispiel das Prinzip erläutern (die Fragestellung ist ähnlich der obigen, nur soll zusätzlich zur URL auch der Titel in der Ergebnismenge angezeigt werden):

```
SELECT d.url, d.title
FROM Document d
SUCH THAT ‘‘http://www.informatik.uni-trier.de/~ley/db’’= ->|=> d
WHERE d.title CONTAINS ‘‘SIGMOD’’
```

Der erste Teil des Select-Statements ist Standard-SQL und kann sich lediglich auf die beiden bekannten Relationen beziehen. In der From-Klausel kann eingeschränkt werden, auf welchen Dokumenten der Relation die Suche ausgeführt werden soll, indem URLs angegeben werden und ggf. andere Kriterien, die zutreffen müssen. Auch in der Where-Klausel können die Bedingungen in dieser Form spezifiziert werden, wobei sich die Kriterien in beiden Klauseln auf reine Schlagwortsuche im gesamten HTML-Dokument beschränken. Neu ist die Formulierung „SUCH THAT", mit der der „Domain", in dem gesucht werden soll, eingeschränkt werden kann.

Der Schwerpunkt von WebSQL liegt in der Definition einer Notation für die Angabe von Pfaden. Diese kann in der WebSQL-Frage verwendet werden, um die „Suchtiefe" einzuschränken, z.B. kann angegeben werden, nur lokale URLs zurückzugeben bzw. solche, die durch globale Links zu erreichen sind (d.h. es handelt sich um Dokumente, die auf einem anderen Server liegen). Im Beispiel werden alle URLs, die durch einen lokalen oder einen globalen Link (`->` bzw. `=>`) mit der Startseite verbunden sind, zur Auswertung herangezogen. Diese Notation kann durch die Verwendung regulärer Ausdrücke komplexe Pfade beschreiben (z.B. gibt „`->->(=>*)`“ alle Pfade an, die durch zwei lokale Links und anschließend beliebig viele globale Links beschrieben sind). Mit der Pfadnotation ist auch eine Beurteilung der „Kosten“ solcher Anfragen möglich, d.h. es wird ausgewertet, wie lang die Pfade sind, die für eine bestimmte Anfrage durchsucht werden müssen.

Auch in diesem Ansatz wird die Suche von einem oder mehreren bekannten Startdokumenten gestartet, die a priori bekannt sein müssen.

In [3] wird von Lakshmanan et al. WebLog, eine logikbasierte Sprache vorgestellt, die zum Ziel hat, ähnlich wie W3QS, durch eine Suche ein Teil des Netzes als View darzustellen, wobei in WebLog nicht nur eine einfache URL-Liste erzeugt wird, sondern noch weitere Informationen aus den aufgelisteten Dokumenten in diese Liste mit aufgenommen werden. Die Suche an sich läuft über komplette HTML-Dokumente, wobei bedeutungstragende Einheiten identifiziert werden (anhand unterschiedlicher HTML-Tags), die ausgewertet werden. Mit WebLog wird ausgehend von einem Startdokument spezifiziert, welche Eigenschaften von dort wegführende Links erfüllen sollen.

Auch hier soll an einem Beispiel das Prinzip der Sprache dargestellt werden:

```
sigmod_urls[title-->'SIGMOD pages accessible from Ley-Homepage',
           hlink-->>L, occurs-->>T]
<-- http://www.informatik.uni-trier.de/~ley/db [hlink -->> L],
    href(L,U), U[title->T], substring(T, 'SIGMOD')
```

In „sigmod_urls.html“ wird eine Liste von URLs mit deren Titeln aufgebaut, die die im weiteren definierten Bedingungen erfüllen. Die Variable L wird mit den Hyperlinks belegt, die von der Ley-URL ausgehen. Mit dem Builtin-Prädikat *href* werden alle URLs U identifiziert, die durch diese Links erreicht werden. Anschließend werden diejenigen Titel an die Variable T gebunden, die „SIGMOD“ beinhalten. In der Ergebnis-HTML-Seite (die als Titel „SIGMOD pages accessible from Ley-Homepage“ hat) werden die betreffenden Links mit ihren Titeln aufgeführt.

Es werden Builtin-Prädikate und atomare Formeln (z.B. `<url> [<attr>--><val>]`) zur Verfügung gestellt. Zudem gibt es die Möglichkeit, das Ergebnis als HTML-Seite mit verschiedenen durch die Suche gewonnenen Informationen darzustellen, somit ist durch WebLog ein gutes Suchinstrument gegeben. Leider arbeitet WebLog lediglich von einem Dokument aus und nicht über das gesamte WWW, ansonsten ist die Mächtigkeit ähnlich der beiden oben beschriebenen Ansätze.

In [4] wird von Levy, Rajaraman und Ordille das „Information Manifold“ System (IM) vorgestellt, dessen Ziel es ist, komplexe Fragen (d.h. über Schlagwortsuche hinausgehende) an heterogene Quellen (z.B. WWW-Server) zu stellen. Dafür wird eine Sprache zur Beschreibung solcher Quellen spezifiziert. Sie basiert auf Relationen, Klassen und Subklassenbeziehungen und beschreibt einerseits den Inhalt der Quellen und andererseits die Benutzerschnittstelle, d.h welche Informationen bei einer Anfrage angegeben werden können bzw. müssen und welche Informationen zurückgegeben werden. Insgesamt existiert ein „World-View“, in dem alle Beschreibungen gehalten werden und in den neue Beschreibungen und Änderungen integriert werden. An ihn wird letztendlich die Anfrage gestellt.

Auch das Problem der unvollständigen Quellen (Bsp.: Eine Quelle über Datenbankliteratur enthält Datenbankzeitschriften, aber keine Informationen über Konferenzen) wird diskutiert.

Interessant wird die Auswertung einer Anfrage. Dazu wird ein Ausführungsplan erstellt, der – ausgehend von den Quellenbeschreibungen im „World-View“ – ermittelt, welche Quellen überhaupt relevant sind und wie die gewünschte Information aufgrund der Inhalts- und Schnittstelleninformation gewonnen werden kann.

In [5] realisieren Liu, Pu und Lee die gleiche Idee mittels eines „Distributed Interoperable Object Model“ (DIOM). Damit soll es Benutzern ermöglicht werden, auf heterogene Informationsquellen zuzugreifen, ohne dabei ein globales Schema entwerfen und pflegen zu müssen, wie es sonst bei verteilten Datenbanken der Fall ist. Dafür werden eine Beschreibungs- und Anfragesprache (IDL bzw. IQL) beschrieben, die genau diese Unabhängigkeit realisieren soll. Die Anfragen werden dann analysiert und bearbeitet, wobei dieser Ansatz den Schwerpunkt auf die Schnittstellenabstraktion und die Anfrageanalyse legt.

Weder in [4] noch in [5] wird jedoch erläutert, wie Informationen aus dem WWW auf solche Quellenbeschreibungen abgebildet werden können. Es dürfte schwierig sein, zu jedem Server eine aktuelle Informations- und Schnittstellenbeschreibung vorzuhalten.

6 Zusammenfassung und Perspektiven

Im schnell expandierenden WWW wird es immer schwerer, gezielt nach bestimmten Informationen zu suchen. Der ursprüngliche Ansatz, lediglich durch navigierenden Zugriff in diesem Hypertextsystem Informationen zu finden, wird zu ineffizient. Mit den viel genutzten Suchmaschinen ist eine gute Unterstützung gegeben, jedoch zeigen sich wie oben beschrieben auch Nachteile. Datenbanksysteme als WWW-Server einzusetzen, hat den entscheidenden Nachteil, daß sie nur für eine lokale Suche sinnvoll sind. Die in Abschnitt 5 beschriebenen Ansätze stellen eine sinnvolle Kombination von Indexservern und Navigation dar, erfüllen aber ebenfalls noch nicht alle in Abschnitt 2 auf Seite 3 gestellten Anforderungen an Suchwerkzeuge. In Tabelle 1 ist kurz zusammengestellt, welcher der beschriebenen Dienste die eingangs definierten Anforderungen wie erfüllt. Die Systeme W3QS, WebSQL und WebLog bzw. IM und DIOM sind wegen Ihrer Ähnlichkeit zusammengefaßt worden.

Keiner der Dienste erfüllt alle Anforderungen optimal, so daß Anwender beim Nutzen der Dienste je nach ihren eigenen Anforderungen Einschränkungen hinnehmen müssen. Zum jetzigen Zeitpunkt scheint es auch nicht möglich, ein Suchwerkzeug zur Verfügung zu stellen, das allen Ansprüchen genügt.

Als Hauptproblem kristallisiert sich heraus, daß die WWW-Dokumente zwar meist durch HTML strukturiert sind, daraus aber nicht klar ist, zu welchem Thema sie gehören bzw. welche Teile als Schlagwort/Suchkriterium spezifiziert werden können. Durch HTML wird lediglich das Layout eines Dokuments, aber nicht seine Semantik festgelegt, obwohl es dort sicher Zusammenhänge gibt.

Eine Erleichterung wäre eine HTML-Erweiterung um semantische Informationen, so daß

	Anforderungen			
	1. „gesamtes“ WWW	2. Konsistenz	3. Relevanz	4. Eingriffe in WWW-Org.
Suchmaschinen	große Teile des WWW durch Katalogisieren möglichst vieler Dokumente	abhängig von dem Aktualisierungsrhythmus und -verfahren	abhängig vom Analysealgorithmus, der meist auf Schlagwortsuche basiert, deshalb ist die Relevanz oft fraglich	keine Eingriffe nötig
DB-Anbindungen	beschränkt auf *einen* WWW-Server	konsistent	durch das DB-Schema ist Relevanz gegeben	Es muß eine DB-Benutzerschnittstelle zur Verfügung gestellt werden.
W3QL, WebSQL und WebLog	abhängig von den Start-URLs, jedoch durch Navigation flexibler als Start ausschließlich von Katalogen (bei WebLog: Einschränkung – nur von *einem* Dokument ausgehend	konsistent, da direkt während der Suche auf die untersuchten Dokumente zugegriffen wird	kaum mehr als Schlagwortsuche, dadurch ist die Relevanz fraglich	keine Eingriffe nötig
IM/DIOM	abhängig von der Zahl der (manuell) erfaßten Quellen	konsistent, falls Beschreibungen und Serverseiten in einem gepflegt werden	hohe Relevanz, da durch die Beschreibungen die Semantik der Quellen festgelegt wird	Die Quellenbeschreibungen müssen für jede Information erstellt und aktuell gehalten werden, was über die übliche Pflege von HTML-Seiten hinaus geht.

Tabelle 1: Zusammenfassung der Eigenschaften verschiedener Suchdienste

beim Erstellen von HTML-Dokumenten gezielt Schlüsselwörter, die bei einer Suche berücksichtigt werden, angegeben werden können. Ähnlich wie im BibTeX-Format sind auch Schlüssel-Wert-Paare denkbar, um eine – den Attributen in relationalen Datenbanken ähnliche – Semantik zu erreichen.

Ein großes Manko der meisten bekannten Werkzeuge besteht weiterhin darin, daß lediglich die textuellen Bestandteile des WWW durchsucht werden. Mittlerweile sind jedoch Bilder und andere Medien Teile dieses Informationsangebots, die ebenfalls berücksichtigt werden müssen. Gerade dieser Punkt wird mit Zunahme der WWW-Benutzer und auch Anbieter mehr und mehr Priorität erhalten.

Da sich das WWW noch in der Entwicklung befindet, wird es sicher Änderungen/Erweiterungen bei Protokollen oder bei der Serververwaltung geben, so daß es durch solche Änderungen vielleicht möglich wird, sowohl die drei ersten Anforderungen besser zu erfüllen als auch nach multimedialen Informationen zu suchen.

Literatur

[1] *Illustra Products.* `http://www.illustra.com/products.html`

[2] D. Konopnicki und O. Shmueli. W3QS: A Query System for the World-Wide Web. In *Proceedings of the 21st International Conference on Very Large Data Bases (VLDB)*, S. 54–64, 1995.

[3] L. V. S. Lakshmanan, F. Sadri, und I. N. Subramanian. A Declarative Language for Querying and Restructuring the Web. In *Proc. of 6th. International Workshop on Research Issues in Data Engineering, RIDE '96*, 1996.

[4] A. Y. Levy, A. Rajaraman, und J. J. Ordille. Querying Heterogeneous Information Sources Using Source Descriptions. In *Proceedings of the 22st International Conference on Very Large Data Bases (VLDB)*, S. 251–262, 1996.

[5] L. Liu, C. Ying, und Y. Lee. An Adaptive Approach to Query Mediation Across Heterogeneous Information Sources. In *International Conference on Cooperative Information Systems (CoopIS)*, 1996.

[6] A. O. Mendelzon, G. A. Mihaila, und T. Milo. *Querying the World Wide Web.* `ftp://db.toronto.edu/pub/papers/websql.ps.Z`

[7] *O2Web: Building a WWW service on top of the O2 database system.* `http://www.o2tech.fr`

[8] F. Ramm. *Recherchieren und Publizieren im World Wide Web.* Vieweg, 1995.

[9] E. Selberg und O. Etzioni. Multi-Service Search and Comparison Using the MetaCrawler. In *Fourth International World Wide Web Conference*, 1995.

[10] B. Yuwono und D. L. Lee. Search and Ranking Algorithms for Locating Resources on the World Wide Web. In *Proceedings of the Twelfth International Conference on Data Engineering (ICDE)*, S. 164–171, 1996.