Zeitschrift: Comtec : Informations- und Telekommunikationstechnologie = information and telecommunication technology

Herausgeber: Swisscom

Band: 74 (1996)

Heft: 5

Artikel: Jetzt giltet es die Informationsaufbereitung zu verbessern

Autor: Johner, Rudolf / Liechti, Beat

DOI: https://doi.org/10.5169/seals-876767

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 15.07.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

DATENBANKEN IM WEB

JETZT GILT ES DIE INFORMATIONS-AUFBEREITUNG ZU VERBESSERN

In den letzten Jahren wurden speziell von Dienstleistungsunternehmen vermehrt Applikationen auf der Basis von On-line-Systemen realisiert. Das Ziel dieser Applikationen ist es, das Produkte- und Dienstleistungsangebot einem bestimmten Publikum aktuell und attraktiv anzubieten und den Informationsaustausch zwischen Anbietern und Konsumenten zu vereinfachen. Während sich die ersten On-line-Systeme nicht erwartungsgemäss entwickelten, erreichte der World Wide Web – das Paradepferd des weltumspannenden Internet – innert kürzester Zeit gigantische Benutzer- und Volumenzahlen. Der vorliegende Bericht beleuchtet die Probleme der Informationsaufbereitung und zeigt mögliche Lösungen auf. Der technische Teil des Berichts beschreibt die Entwicklung und die Implementierung eines Datenbankgateways für den Web. Abgerundet wird der Bericht mit dem Ausblick auf mögliche konkrete Einsätze der vorgestellten Lösung.

RUDOLF JOHNER UND BEAT LIECHTI, BERN

Ein Erfolgsfaktor des World Wide Web (Web) ist die Art der Informationsaufbereitung: Auf einem Web-Server werden Seiten mit Texten und Bildern – auch zeitsensitive Daten wie Ton, Video und Animationen sind möglich – eingerichtet und miteinander verknüpft. Der Web-Server ist eine Komponente im Web und kann weltweit erreicht werden. Das Informationsangebot eines Web-Servers führt meistens über ein Porträt der Firma oder Institution zu detaillierteren Angaben über Produkte und Dienstleistungen. Um Datenredundanz und all die damit verbundenen Probleme zu vermeiden, sollten die Web-Seiten keine Daten enthalten, welche bereits in firmeninternen Datenbanken vorliegen und unterhalten werden.

GRUNDLAGEN

World Wide Web

Die Zuwachszahlen, die Präsenz in den Medien und Buchhandlungen sowie ausgefallene Ideen wie Internet-Kaffees zeugen von der Anziehungskraft des Web auf Informationsbezüger wie auf Informationsanbieter. Tatsächlich glauben viele – frei nach Michael Gorbatschow: *«Wer zu spät kommt, den bestraft das Leben.»* –, etwas zu verpassen, wenn sie nicht mitmachen. Oftmals wird der Web als Informatik- und Medienrevolution bezeichnet. Dabei dürfen wir nicht übersehen, dass der Web einerseits auf bewährten, darunter auch auf alten Technologien und Methoden basiert, und anderseits primär durch Einfachheit besticht. Dies sichert ihm die Akzeptanz bei allen Akteuren (Konsumenten und Anbietern): das Einrichten eines Web-Servers sowie einer Homepage bedeutet einen Aufwand von ein paar Tagen; der Benutzer benötigt einen Web-Browser, einen Zugang zum Internet – und los geht's. Der Web hat auch eine grosse Bedeutung für die Informatikszene. Gerade in jüngster Zeit wird deutlich erkennbar, dass die Dominanz einzelner Firmen durch den Web aufgeweicht werden könnte, was durchaus als positiv zu werten wäre.

ELECTRONIC DELIVERY

Desktop Publishing

Das klassische Desktop Publishing umfasst die computergestützte Erstellung von Dokumenten in Papierform. Heute kann jedermann für wenig Geld mit einem PC, einem Textsystem und einem Laserdrucker eine Kleindruckerei betreiben. Desktop Publishing ist deshalb auch prinzipiell ein Störfaktor für effiziente Bürokommunikation. Gerade die Möglichkeiten, mit bestechenden Tools prunkvolle Dokumentationen zu erstellen, verursachen bei vielen Benutzern eine Denkweise, die Konzepten für unternehmensweite Dokumentenhaltung zuwiderläuft.

Electronic Viewing

Im Gegensatz zum Desktop Publishing werden mit Electronic Viewing Informationen nicht auf Papier herausgegeben, sondern am Bildschirm angezeigt; der Computer wird dabei als Medium auch zum Konsumieren von Informationen eingesetzt. In den Bereich Viewing lassen sich unter anderem die typischen Multimedia-Applikationen einordnen. Zahlreiche Hersteller bieten heute Viewing-Tools an. Viele ermöglichen die Präsentation elektronischer Bücher (z. B. Systemdokumentationen), andere haben bereits Konzepte mit Hypertextnavigation realisiert.

Electronic Distribution

Darunter wird der eigentliche Transport von Informationen verstanden. Multimedia-Applikationen zeichnen sich oft dadurch aus, dass die zugehörigen Informationen von CD-ROM stammen. Dieses Trägermedium hat auf dem Markt sicher grosse Bedeutung (z. B. bei CBT), muss aber aufgrund der statischen Informationen und der deshalb mangelnden Aktualität der Informationen relativiert werden. Gerade dies ist im Geschäftsbereich nebst andern Faktoren wie beispielsweise Interaktivität von entscheidender Bedeutung. Electronic Distribution verlangt deshalb nach On-line-Verbindungen. Anzustreben sind Viewing-Tools, welche über geeignete Übertragungsmedien auf Verwaltungssysteme zugreifen.

Vorbild World Wide Web

In eindrücklicher Weise demonstriert der World Wide Web (Web) die Umsetzung von Electronic Viewing und Distribution in die Praxis. Die Möglichkeit der nicht zielgerichteten Informationssuche (bekannt unter dem Begriff «Infosurfen») verzaubert Massen von Konsumenten und führte dazu, dass in kürzester Zeit ein riesiges Informationsangebot zur Verfügung stand. Das Gleichgewicht zwischen Konsumenten und Anbietern (frühere On-line-Dienste litten jeweils unter einem mangelhaften Informationsangebot) fordert grosse Unternehmer aus diversen Branchen, wie Verleger, Computerhersteller und Telecom-Anbieter, zum Mitmachen heraus. Ob durch deren Engagement die Qualität der Informationen und der angebotenen Dienstleistungen (z. B. Sicherheitsdienste für elektronische Märkte) zunimmt und ob ein kommerzialisierter Web weiterhin Erfolg hat, wird die Zukunft zeigen.

HTML

HTML steht für HyperText Markup Language. Es ist eine Weiterentwicklung bzw. Spezialisierung der viel älteren und von der ISO normierten Sprache SGML. Während SGML vorwiegend für umfassende technische Dokumentationen und Ablagen (z. B. im Flugzeugbau) verwendet wird, ist HTML grundsätzlich für «Electronic Viewing» (siehe Kasten «Electronic Delivery») ausgelegt.
Ein interessanter Aspekt von HTML ist der Vergleich zwischen digitalisierten Dokumenten und Dokumenten in Papierform. Die im Zusammenhang mit Dokumentenmanagement entstandene Idee der Verbunddokumente (Compound Documents) wurde nämlich mit dem Web bzw. dank HTML in die Praxis umgesetzt. Im Gegensatz zu elektronischen Abbildern von Dokumenten, das heisst einer 1 : 1-Übernahme aus der Papierform, liegt dem Konzept der Compound Documents die folgende, grundsätzlich neue Definition des Begriffs Dokument zugrunde:
«Ein Dokument ist die Darstellung von Informationen als Sammlung (Collection) während einer bestimmten Zeit.»
Erfolgt die Darstellung auf Papier, sind die dargestellten Informationen statisch, das heisst, Änderungen müssen durch Versionenvergabe kontrolliert werden; zudem können zeitkritische Daten (Ton, Video) nicht dargestellt werden. Erfolgt die Darstellung mit dem Computer und zudem losgelöst vom Textsystem (siehe Kasten «Das Medium Computer»), können neue Konzepte des Dokumentmanagements realisiert werden:

- Die Verwaltung der Informationen erfolgt unabhängig vom Verwendungszweck. Das Verwaltungssystem garantiert für Datenkonsistenz und -sicherheit.
- Der Zugriff auf Daten erfolgt mittels einer definierten Sprache (im Web ist dies HTML).

- Applikation (z. B. Document Viewer) und Verwaltungssystem sind durch eine definierte Schnittstelle getrennt (im Web sind dies die bekannten Internet-Protokolle auf verschiedenen Ebenen).
- Die Darstellung von zeitkritischen Daten (Ton, Video) ist möglich.
- Eine grundlegende Forderung für mobile Kommunikation, auf Daten zuzugreifen, wenn sie benötigt werden, ist erfüllt.

Globale Märkte

Führen globale Datennetze auch zu globalen Märkten? Das Beispiel Internet bejaht diese Frage, wirft aber gleichzeitig neue Fragezeichen auf, indem es uns vor zahlreiche, zum Teil gravierende Probleme stellt. Globale Datennetze sind längst nicht mehr eine rein technische Angelegenheit, sondern ein Thema, das alle Gesellschaftsbereiche tangiert und auch speziell die Politiker zum Handeln herausfordert.

Elektronische Märkte

Während der Markt grundsätzlich die gesellschaftlichen Koordinationsfunktionen zwischen allen Arten von volkswirtschaftlichen Aktivitäten umfasst, dienen elektronische Märkte als Drehscheiben für Markttransaktionen in allen Phasen. Sie sind die künftige Grundinfrastruktur für globale Märkte.
Datennetze schaffen Berührungspunkte für alle Teilnehmer (Akteure) und begünstigen im Vergleich zu konventionellen Marktinfrastrukturen die nötigen Koordinationsmechanismen beim Aufeinandertreffen von Angebot und Nachfrage:

- Angebote liegen aktuell und umfassend vor.
- Distanz und Zeitverschiebung zwischen den Teilnehmern sowie ihr Standort sind unbedeutend; die Welt wird zum Dorf.
- Geschäftsprozesse sind effizienter, Mittel und Techniken zur Produktivitätssteigerung wie beispielsweise Just-in-time finden Verwendung.
- Standards wie EDI/EDIFACT kommen zum Tragen.
- Die volle Digitalisierung vermeidet Medienbrüche und erlaubt den Einsatz zeitkritischer Daten (Ton, Video, Animation).

Die heranwachsende Flut von Informationen weckt zwangsläufig das Bedürfnis nach Orientierung und Strukturierung. Elektronische Märkte entwickeln inhärent das Potential für einen neuen Markt von Dienstleistungen wie beispielsweise Mehrwertdienste («value added services»). Es ist dies der grosse Wachstumsmarkt der Telekommunikation.

Die Rolle des Informationsanbieters

Ein interessanter Aspekt, den Datennetze in globalen Märkten darstellen, ist die Rolle des Informationsanbieters. Die meisten Web-Benutzer sind Konsumenten, das heisst, sie navigieren durch die Informationsangebote einzelner Web-Server, ohne selbst Informationen anzubieten. Eine Charakteristik des Web ist es jedoch, mit einfachsten Mitteln und für wenig Geld einen Server mit eigenen Seiten in ihn einzubinden. Ein Konsument kann also ohne grossen Aufwand in die Rolle des Informationsanbieters («broadcaster») wechseln. Es ist deshalb zu erwarten, dass künftig auch kleine und mittlere Unternehmen vermehrt auf dem Web präsent sein und so den elektronischen Markt entscheidend erweitern und beleben werden. Sie werden den Web als neuen Akquisitions- und Distributionskanal, als Werbeträger sowie als Medium zur Abwicklung neuer elektronischer Dienstleistungen nutzen.

Informationsaufbereitung

In der allgemeinen Web-Euphorie haben Zweifler und Kritiker einen schweren Stand. Die Autoren sind sich dessen bewusst und versuchen deshalb die im Rahmen der Problemkreise aufgeworfenen Schwachstellen beim Web objektiv zu behandeln, ohne dabei den Web als Ganzes in Frage zu stellen.

Datenbanken und Web-Server: ein Vergleich

Die effiziente Informationsbewirtschaftung ist für heutige Unternehmen unabdingbar. Relationale Datenbanken (RDBMS) sind das Kernstück aktueller Informationssysteme; sie erfüllen die Anforderungen an Leistung und Datenintegrität (Konsistenz, Sicherheit und Datenschutz). Den Markt dominieren zurzeit die Hersteller von RDBMS. Der Trend Richtung Objektorientierung ist auch bei Datenbanksystemen erkennbar, wobei aus Gründen der Systemleistung, Zuverlässigkeit und Marktverbreitung eine Ablösung von RDBMS durch objektorientierte Systeme höchstens langfristig denkbar ist. Zurzeit gilt das Augenmerk den zahlreichen Projekten, bei denen die Kopplung bzw. das Bindeglied zwischen objektorientierten Applikationen und RDBMS untersucht wird. Ansätze sind in ersten Entwicklungen beispielsweise von Datenbank-Brokers oder Persistence-Frameworks sichtbar.
Mit dem relationalen Modell wurde gleichzeitig das 3-Schema-Prinzip eingeführt. Dieses Prinzip beruht auf der Mehrsichtigkeit von Datenbeständen (intern, extern und konzeptionell) und gilt als Grundlage für die konsequente Trennung von Datenmanipulation und Datendefinition bzw. von den Inhalten und der Beschreibung einer Datenbank (Metadaten).
Ein weiteres Merkmal von RDBMS ist die Datenbanksprache SQL. Dieser Standard wird heute bei jedem namhaften RDBMS angewendet; SQL definiert den Zugriff (Manipulation und Definition) auf die Datenbestände.
Der Vergleich einer Datenbank mit einem Web-Server bezüglich Informationsaufbereitung ergibt folgendes Bild:

- Das Design von Datenbanken lehnt sich an das relationale Modell. Beziehungen zwischen Entitäten oder Tabellen werden in Applikationen dazu verwendet, je nach Bedürfnis einzelne Datenattribute zu einem Ganzen zusammenzufügen. Web-Server verwalten einzelne Web-Seiten, bestehend aus lose aneinandergereihten Datenblöcken. Innerhalb einer Seite können Links angebracht werden, über die ein Benutzer zu weiteren Web-Seiten navigieren kann. Ein eigentliches Datenmodell kennt der Web-Server nicht. Zusammenhänge zwischen den Daten oder den Datenblöcken sind nur mit den Links definiert und wiedererkennbar.
- Datenbanken verfolgen konsequent die Trennung zwischen Datenmanipulation und Datendefinition über definierte Sichten. Beim Web-Server sind diese Funktionen vermischt, das heisst, die Informa-

DAS MEDIUM COMPUTER

«The Paper-Moving Problem» lautet die Überschrift einer Broschüre der Firma, die PostScript entwickelte. Danach folgen Zahlen: Pro Geschäftstag werden weltweit 200 Mio Seiten Papier abgelegt und 600 Mio Dokumente erstellt, Büroleute schlagen sich während 60 % der Arbeitszeit mit Dokumenten herum, Führungsleute verschwenden drei Stunden pro Tag für Dokumentensuche. Das Durchschnittsdokument wird 19mal kopiert usw. Ob diese Zahlen exakt stimmen, sei dahingestellt. Sicher weisen sie auf uns bekannte Probleme hin:

- zu viele Informationen liegen in Dokumentenform vor
- die Verwaltung der Dokumente ist ungenügend
- die Dokumentenablagen sind redundant
- die Arbeit mit Dokumenten ist ineffizient

Vom Computer als «Mittel zur Informationsproduktion» . . .

Nur selten wird ein (umfassendes) Dokument am Bildschirm gelesen. Arbeiten mit dem Textsystem werden in den meisten Fällen mit dem Ausdruck auf Papier abgeschlossen; das heisst, Textsysteme sind primär für den Papierdruck und nicht zur Präsentation von Informationen bestimmt. Die Gründe dafür liegen einerseits bei den fehlenden Bedienungselementen und den störenden typographischen Hilfsinformationen sowie anderseits (und das ist das Entscheidende) beim Aufbau von Dokumenten. Da ein Dokument auf Papier von einem bestimmten Format gedruckt wird, ist der Autor zu einer thematischen Gliederung der Informationen gezwungen; die Informationselemente werden dabei sequentiell aneinandergereiht. Der Leser jedoch will oder kann oft nicht den Gedankengang des Autors exakt nachvollziehen, er liest nicht sequentiell; seine Denkweise ist assoziativ. Er überfliegt ein Fachbuch oder einen Bericht zuerst inhaltlich, um sich grob zu informieren. Je nach Interesse und vorhandenen Kenntnissen liest oder studiert er dann einzelne Passagen oder Details: bei bestimmten Stellen wird eine Markierung angebracht, vor- oder zurückgeblättert, weil vielleicht eine weiterführende Erklärung gesucht werden muss, usw. Hilfsmittel, wie Inhaltsverzeichnis und Index, helfen ihm, den thematischen Aufbau zu überblicken und liefern wichtige Referenzangaben, über die er rasch auf die gesuchten Stellen zugreifen kann.

Die Papierform erlaubt dem Leser also, Informationen seinen Bedürfnissen entsprechend zu verknüpfen. Mit dem Textsystem ist genau dies nicht möglich: die Informationen sind zum Drucken und nicht zum Lesen am Bildschirm aufbereitet. Das Textsystem ist zum Erstellen und Bearbeiten von Dokumenten, jedoch nicht zum Präsentieren gemacht.

Schon vor Jahren wurde wegen dieser Umstände nach Alternativen zum Textsystem gesucht. Unter dem Begriff Hypertext entwickelte sich eine neue Form der Textaufbereitung: Texte sollen in erster Linie nicht mehr gedruckt, sondern am Bildschirm gelesen werden können. Dazu werden Texte abschnittweise abgespeichert und referenziert. Der Leser kann sie nach seinen Bedürfnissen aufrufen und gliedern.

. . . zum Computer als «Medium»

Die Idee, den Computer als Medium einzusetzen, stammt aus dem Marktsegment Multimedia. Im Vordergrund stand primär das Bedürfnis, alle Informationsarten, die der Computer verarbeiten kann, also nicht nur diskrete (Text, Grafik, Bilder usw.), sondern auch zeitkritische (Film und Ton), zu präsentieren. Hypermedia (die Aufbereitung und Verwaltung von Informationen aller Art) sowie Multimedia (die interaktive Präsentation dieser Informationen) wurden ins Leben gerufen. Erst die heutige Generation von Arbeitsstationen erlaubt jedoch Multimedia-Applikationen. Benötigt werden genügend CPU- und Daten-I/O-Leistungen sowie eine graphische Benutzeroberfläche, welche neue Dialogformen und Bedienungselemente ermöglicht.

In Zukunft wird sich Multimedia als Alternative zur Erstellung von Dokumenten auf Papier aufdrängen. Im Heim- oder Unterhaltungsbereich ist die CD als Trägermedium mit sehr grosser Kapazität ein Ersatz für Bücher, Fachschriften usw.

Der Web ist ein Beispiel dafür, wie der Computer als Medium eingesetzt werden kann. Was ihn von Multimedia unterscheidet, ist einerseits die enge Kopplung mit dem Netzwerk und anderseits (als Gegensatz dazu) die Entkopplung zwischen Applikation und Daten. Der Web verdeutlicht auch das Zusammenwachsen von Computer und Datenkommunikation. Das Netzwerk beinhaltet die Informationen (Daten und Programme), der Computer dient als Verwaltungssystem beim Anbieter und als Medium beim Konsumenten.

tionen sind Inhalt, Beschreibung und Struktur zugleich.
- Die Metadaten einer Datenbank (die Datendefinitionen) umfassen auch Konsistenzregeln für einzelne Attribute, Tabellen und Beziehungen zwischen Attributen bzw. zwischen Tabellen; zudem umfassen sie Regeln bezüglich Zugriffsrechten. Diese Voraussetzungen für Datenintegrität sind bei einem Web-Server nicht oder nur ungenügend gegeben.

Informationen stellen oft ein wertvolles Gut dar. Es muss deshalb bei der Informationsaufbereitung alles getan werden, um ein Chaos zu vermeiden; speziell dann, wenn die Daten in hohem Grad strukturiert sind und ein selektiver Zugriff erwünscht wird.

Problemkreise

Datenredundanz

«Das Grundübel für mangelnde Aktualität und Verletzung der Datenintegrität.» Datenredundanz ist dann vorhanden, wenn Datenbestände physikalisch mehrfach vorhanden sind. Es gibt gute Gründe, Daten zu replizieren, das heisst Kopien davon anzulegen. Die Datenreplikation hat sich als Form für verteilte Datenbanken in der Praxis auch durchgesetzt. Werden Replikate manuell, das heisst nicht unter Kontrolle des Datenbanksystems, angelegt, müssen auch Änderungen am Original manuell übertragen bzw. nachgeführt werden. In der Praxis funktioniert dies in der Regel nicht. Manuell angelegte Kopien haben deshalb nur temporäre Gültigkeit.
Viele Unternehmen betreiben Datenbanken und unterhalten darin Informationen über ihre Produkte und Dienstleistungen. Der Web bietet nun die Möglichkeit, diese Informationen dem Kundenkreis des Unternehmens zu präsentieren. Werden auf dem Web-Server manuell Kopien von der existierenden Datenbank angelegt, führt dies zu unerwünschter Datenredundanz.

Aktualität

«Nur aktuelle Informationen sind akzeptabel.» Von früheren On-line-Diensten sind uns Zustände mit veralteten Informationen bekannt. Der Grund für die mangelnde Aktualität von Informationen eines On-line-Dienstes liegt meist darin, dass nicht mit dem Original, sondern mit Datenkopien gearbeitet wird, das heisst, der Server des On-line-Dienstes verwaltet seine eigenen Datenbestände und greift nicht auf die Originaldaten zu. Die Praxis zeigt, dass Kopien meist ungenügend unterhalten werden, das heisst, Änderungen am Original werden an der Kopie nicht nachgeführt.
Ein Web-Server kann einerseits Originaldaten enthalten, muss aber anderseits in der Lage sein, auf externe Datenbestände (z. B. eine Datenbank) zuzugreifen und so sein Informationsangebot dynamisch zu ergänzen.

Datenintegrität

Konsistenz
«Informationssysteme müssen widerspruchsfreie Daten enthalten.» Es sind Fälle bekannt, wo Firmen auf dem Web falsche Daten propagierten und deshalb behaftet wurden. Oft sind es Preisangaben für Produkte und Dienstleistungen, die falsch eingetragen oder nicht aktualisiert wurden. Gerade diese Daten unterliegen in der hektischen Geschäftswelt grosser Dynamik und verlangen deshalb nach ordentlicher Pflege, das heisst, sie müssen definierte Konsistenzregeln erfüllen. Bei einem Web-Server ist der Web-Master für die Datenkonsistenz besorgt, das heisst, er überprüft die Konsistenz jeweils aufgrund der Inhalte. Bei einer Datenbank sind die Konsistenzregeln Teil der beschreibenden Daten und nicht die Inhalte selbst. Eine gut aufgesetzte Datenbank verhindert weitestgehend das Auftreten von Inkonsistenzen.

Datensicherheit
«Informationen sind ein wertvolles Gut.» Ähnlich wie die Konsistenz einer Datenbank durch Konsistenzregeln gewährleistet wird, bestimmen Zugriffsregeln den selektiven Zugriff auf die enthaltenen Daten. Eine übergeordnete Autorität bestimmt bei einer Datenbank den Benutzerkreis und die jeweiligen Zugriffsrechte einzelner Benutzer bzw. von Benutzergruppen. Im Gegensatz dazu ist bei einem Web-Server in der Regel keine Feinabstimmung möglich. Es ist wiederum der Web-Master, der durch Wahl des Speicherorts gleichzeitig die Zugriffsrechte bestimmt.
Es ist bekannt, dass Datenschutzprobleme wie beispielsweise der Umgang mit Kreditkarteninformationen nicht bei der Übertragung, sondern bei der Speicherung der Daten zu suchen sind. Es ist (entgegen vielen Reportagen in den Massenmedien) viel einfacher, bei einem auf Files basierenden System, wie ein Web-Server es darstellt, via Insider an solche Daten zu kommen, als sie bei den Übertragungseinrichtungen herauszufiltern. Sensitive Daten gehören deshalb in die Kontrolle eines Datenbanksystems.

Datenschutz
«Der Schutz der Betroffenen.» Im Gegensatz zur Datensicherheit geht es dabei nicht um die Daten selbst, sondern um den Schutz der betroffenen Personen, über die Daten angelegt werden. Um Rechenschaft darüber abzugeben, welche Daten erfasst werden, benötigen die Verantwortlichen einen exakten Datenkatalog. Bei einer Datenbank ist dieser durch die Datendefinition bereits gegeben. Bei einem Web-Server sind die Informationen nicht so strukturiert, dass sie einem vordefinierten Datenkatalog entsprechen würden. Was gespeichert wird, ist wiederum durch die Inhalte selbst und nicht durch deren Beschreibung ersichtlich.

Lösungsmodelle

Informationsdesign

Es wäre falsch, den Web-Server nur noch als Zugangsknoten für Datenbanken einzusetzen. Wie in der Einleitung erwähnt, ist ja gerade die Informationsaufbereitung ein Erfolgsfaktor für den Web. Für die Diskussion der Lösungsmodelle gehen wir deshalb von folgendem Grundsatz aus:
«Datenbanken ergänzen das Informationsangebot eines Web-Servers.»
Eine exakte Abgrenzung darüber, welche Informationen entweder im Web-Server oder in der Datenbank unterhalten werden, ist nicht möglich. Es ist grundsätzlich eine Designentscheidung aufgrund der Informationsarten und der im Kapitel Problemkreise behandelten Punkte.
Kandidaten für den Web-Server sind:
- längere zusammenhängende Texte
- unstrukturierte Daten
- Dokumente, die als Ganzes eingebunden sind, beispielsweise PDF-Dateien

CO-HTML

CO-HTML definiert HTML-artige Konstrukte, um mit Composite Objects zu arbeiten und deren Werte in ein HTML-Dokument zu übernehmen.

Mengendeklaration

```
<!-- CO SET Mitarbeiter M -->
...
<!-- CO /SET -->
```

Mit einer Mengendeklaration wird eine Menge von bestimmten Composite Objects angefordert und mit einem Namen versehen. Im obigen Beispiel wird eine Menge von Mitarbeitern mit dem Namen M angefordert. Mit der Deklaration ist auch gleich die Instanzierung verbunden, die vom CO-Manager übernommen wird.

Iteration

```
<!-- CO ITER M -->
...
<!-- CO /ITER -->
```

Um durch eine Menge von Composite Objects zu iterieren, wird das Iterationskonstrukt benutzt. Im Beispiel wird durch die Menge M iteriert.

Wertreferenz

```
<!-- CO $M.Name -->
<!-- CO $M.Vorname -->
<!-- CO $M.Titel -->
```

Die Wertreferenz holt den Wert eines Felds des aktuellen Objekts einer Menge in das Dokument. Hier werden der Name, der Vorname und der Titel des aktuellen Objekts der Menge M in das Dokument übernommen.

Selektion

```
<!-- CO IF #M = «0» -->
...
<!-- CO ELSIF #M = «1» AND $M.Name = «Muster» -->
...
<!-- CO ELSE -->
...
<!-- CO /ITER -->
```

Mit der Selektion kann die Gestaltung eines HTML-Dokuments von gewissen Bedingungen abhängig gemacht werden. Ist beispielsweise eine Menge leer, so kann eine entsprechende Meldung in das Dokument aufgenommen werden. Enthält die Menge nur ein Element, kann das Objekt detailliert dargestellt werden. Enthält die Menge mehrere Elemente, so kann eine Liste der Elemente erzeugt werden (mit Iteration und Wertreferenz).

Einbezug von Dateien

```
<!-- CO INCL = «filename» -->
```

Um beispielsweise umfangreiche Dokumente in mehrere kleinere Dateien aufteilen zu können, ist dieser Befehl sehr nützlich. Er kann auch immer dann angewendet werden, wenn derselbe HTML-Code mehrmals in verschiedenen Dokumenten zur Anwendung kommt.

Vorwärts-/Rückwärtsreferenz

```
<!-- CO FREF = «filename» SINGLE M -->
...
<!-- CO /FREF -->
...
<!-- CO BREF -->
...
<!-- CO /BREF -->
```

Um eine aktive HTML-Referenz auf ein bestimmtes Objekt zu erzeugen, steht die Vorwärtsreferenz zur Verfügung. So kann ein einzelnes Composite Object aus einer Menge entnommen und an ein anderes HTML-Dokument übergeben werden.

Suchformulare

Um dem Web-Client die Möglichkeit zu geben, einen bestimmten Datensatz (bzw. ein bestimmtes Composite Object) zu suchen, stellt CO-HTML Suchformulare zur Verfügung. Da für eine Suche beliebige Kriterien und Jokerzeichen benutzt werden können, sind mit Suchformularen bereits recht interessante Applikationen möglich.

Editierformulare

Um via Web einen bestehenden Datensatz zu editieren oder zu löschen, oder einen neuen Datensatz zu erzeugen, können Editierformulare eingesetzt werden. Mit diesem Feature kann eine komplette Datenbankapplikation mit CO-HTML implementiert werden, so beispielsweise ein Reservationssystem für Räume oder Anfragen für Unterlagen und dergleichen mehr.

- statische Daten, das heisst Daten mit langer Gültigkeitsdauer
- öffentlich zugängliche Daten

Kandidaten für die Datenbank sind:
- Daten, die bereits im Rahmen von (In-house-)Applikationen erfasst und unterhalten werden
- strukturierte oder strukturierbare Daten
- Daten, die in Tabellenform vorliegen, das heisst, sie sind mengenorientiert
- dynamische Daten, das heisst Daten mit kurzer Gültigkeitsdauer
- sensitive Daten, das heisst, selektiver Zugriff ist erforderlich
- Referenzen auf binäre Dateien wie Dokumente, Bilder, Videos usw.

Systemdesign

In Bild 1 – Lösungsmodelle – sind vier Modelle prinzipmässig dargestellt:

Modell a
Der Web-Client (in der Regel ein Web-Browser, beispielsweise Netscape) kommuniziert mit einem Web-Server. Dieser holt bei Bedarf Daten aus der direkt gekoppelten Datenbank. Der Web-Server kennt die Datenbeschreibung und die Zugangsregeln für die spezifische Datenbank.

Modell b
Der Web-Client greift direkt auf die Datenbank zu. Die Lösung kann beispielsweise mit HotJava realisiert werden. Bei Bedarf holt sich der Web-Client vom Web-Server ein sehr spezifisches Applikationsprogramm, ein Java Applet, startet dieses lokal und kommuniziert so mit der Datenbank. Diese Form von verteilten Applikationen ist sehr zukunftsorientiert.

Modell c
Web-Server und Datenbank sind durch ein Gateway entkoppelt. Der Web-Server kennt die Datenbeschreibung bzw. die Zugangsregeln der Datenbank nicht zwingend.

Modell d
Der Web-Client greift via Gateway auf die Datenbank zu. Die Lösung ist sonst identisch mit Modell b.
Zur Beurteilung der vorgestellten Modelle definieren wir folgende Kriterien:
- Grad der Datenbankkopplung: Wie eng ist die Kopplung der Datenbank auf Netzwerk- und SQL-Ebene (Netzwerk- und Datenbankprotokolle)?
- Abstraktion der Datenbank: Entspricht die Sicht auf die Datenbank dem externen Schema oder existiert eine (z. B. objektorientierte und damit praxisgerechtere) Zwischenschicht?
- Benötigte Komponenten.
- Applikationsdesign: mögliche Bedienungselemente und Dialogformen.
- Gestaltungsfreiheit: Wie können Datenbankextrakte (Resultate von Abfragen) in Web-Seiten bzw. wie können Abfragestatements (z. B. SQL) in HTML eingebunden werden?

Tabelle 1 behandelt Vor- und Nachteile der diskutierten Modelle aufgrund der aufgestellten Kriterien.

Die Entkopplung der Datenbank

In den Modellen c und d haben wir die Datenbank durch den Einsatz eines Gateways vom Web-Server bzw. vom Web-Client entkoppelt. Diese zusätzliche Komponente betrachten wir aus folgenden Überlegungen als notwendig:

Modularität
Auf Netzwerkebene muss nur noch mit dem Gateway über ein gemeinsames Protokoll kommuniziert werden. Das Gateway stellt so ein protokollneutrales Bindeglied dar.
Aus Sicht Web-Server bzw. Web-Client sind die zahlreichen und sehr spezifischen Schnittstellen (API) von Datenbanksystemen unbedeutend, das heisst, der Web-Master bzw. der Entwickler eines Applets (z. B. Java) muss sich nicht darum kümmern. Das Gateway stellt ihnen eine universelle und auf die Möglichkeiten des Web angepasste Schnittstelle zur Verfügung.

Abstraktion der Datenbank
Programmierer haben oft Mühe mit dem Einbinden von Datenbankbefehlen in ihre Applikation. Sie verwenden in der Regel «embedded SQL» oder spezielle Middleware. Mit dem Aufkommen der objektorientierten Programmierung wurde ein neuer Ansatz gefunden, der die Datenbank über einen Persistence-Layer abstrahiert. Der Web-Master bzw. der Entwickler eines Applets (z. B. Java) steht vor densel-

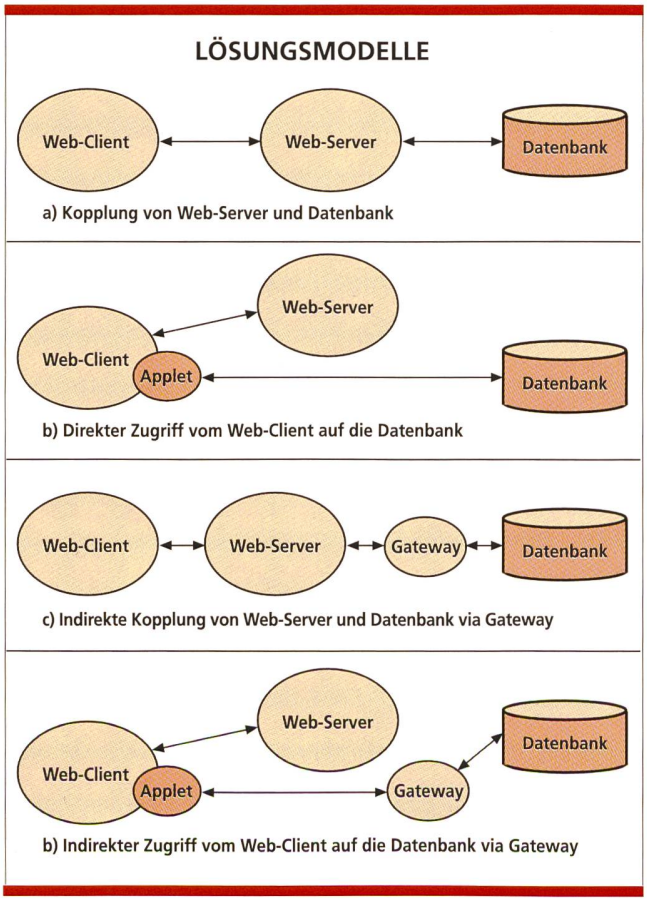


Bild 1.
Vier prinzipmässig dargestellte Lösungsmodelle.

	pro	kontra
Modell a	Benötigt wenig zusätzliche Komponenten	Keine Abstraktion der Datenbank aus Sicht Web-Server, mangelnde Gestaltungsfreiheit, eingeschränkte Bedienungselemente und Dialogformen
Modell b	Möglichkeit zur Einbindung einer Datenbankapplikation mit bekannten Bedienungselementen (z. B. von Windows) und Dialogformen	Keine Abstraktion der Datenbank aus Sicht Web-Client, noch nicht ausgereifte Technologie
Modell c	Entkopplung der Datenbank auf Netzwerk- und SQL-Ebene, mehr Gestaltungsfreiheit	Benötigt eine zusätzliche Komponente, eingeschränkte Bedienungselemente und Dialogformen
Modell d	Wie Modelle b und c zusammen	Benötigt eine zusätzliche Komponente, noch nicht ausgereifte Technologie

Vor- und Nachteile der Modelle beurteilt nach bestimmten Kriterien.

ben Problemen, wenn er direkt mit SQL bzw. der SQL-Ebene konfrontiert wird. Er wünscht deshalb auf «vorgefertigte Objekte», wie Records, Listen und Kombinationen davon, zuzugreifen. Mit Hilfe des Gateways können solche Objekte nach Bedarf definiert und den Clients zur Verfügung gestellt werden. Es ist dann Sache des Gateways, die Objekte zu instanzieren, das heisst, spezifische Datenbankbefehle auszuführen und die Objekte zu verwalten.

Gestaltungsfreiheit
Die einfachste Art, das Informationsangebot des Web-Servers mit der Datenbank zu ergänzen, wäre, nach Bedarf einfach zusätzliche Web-Seiten mit Inhalten der Datenbank zu generieren. In der Praxis würde diese Methode den Ansprüchen an das User-Interface nicht standhalten. Wir erweitern deshalb den im Kapitel Informationsdesign aufgestellten Grundsatz wie folgt:
«Einzelne Web-Seiten werden mit Inhalten der Datenbank ergänzt.»
Ein Web-Server ist standardmässig nicht in der Lage, Web-Seiten dynamisch zu generieren. Wir können dem Grundsatz deshalb nur nachkommen, wenn wir zusätzliche Funktionalität im Gateway implementieren.

Analyse existierender Lösungen

Das Common Gateway Interface (siehe Kasten «CGI») ist sehr vielseitig und geradezu prädestiniert, um eine Datenbank an den Web anzubinden. Praktisch alle existierenden Lösungen bauen auf dem CGI als Schnittstelle des Web-Servers zur Datenbank auf, denn es gibt eigentlich keine vergleichbare Alternative. Technisch gesehen, entspricht dies dem bereits erläuterten Modell c. Was jedoch die konzeptionelle Anbindung angeht, so gibt es unterschiedliche Abstraktionsstufen.

Festcodierte Scripts

Praktisch jeder Datenbankanbieter hat Bibliotheken im Angebot, die den Zugriff auf die Datenbank mit einer Programmiersprache ermöglichen. So kann eine Datenbankabfrage in ein Gateway Script verpackt werden, das die resultierende Tabelle in ein entsprechend formatiertes HTML-Dokument umwandelt. Diese relativ simpel zu implementierende Lösung hat mehrere Nachteile. So muss pro Relation ein Script erstellt werden, das Abfrage und Formatierung durchführt. Wird das Schema der Datenbank verändert, so müssen eventuell gleich mehrere Gateway Scripts umgeschrieben werden, was recht aufwendig sein kann. Der Autor des Scripts bildet hier die Brücke zwischen Datenbank und Web, er muss beide Welten kennen und sowohl SQL- als auch HTML-kundig sein.
Der Grad der Datenbankkopplung ist gross, ein solches Script ist für ein ganz bestimmtes Datenbankschema geschrieben worden und kann praktisch nicht portiert werden. Die Abstraktion der Datenbank bleibt dem relationalen Modell verhaftet. Die Gestaltungsfreiheit für das resultierende HTML-Dokument ist zwar gegeben, aber auch bei der kleinsten Änderung des Layouts muss das Script entsprechend angepasst werden. Es ist offensichtlich, dass ein solches System keine attraktive Lösung darstellt.

Scripts mit Metadaten

Um ein Gateway Script nicht nur für eine bestimmte Relation nutzen zu können, werden Metadaten spezifiziert, die Auskunft über das Datenbankschema geben. Mittels Parametrierung des Gateway Scripts kann eine bestimmte Relation abgefragt werden. Bei Änderungen des Datenbankschemas werden nur die Metadaten angepasst, was vom Aufwand her bedeutend einfacher ist, als in einem Script SQL-Befehle und Formatierung der Daten zu ändern.
Der Grad der Datenbankkopplung ist kleiner als bei der vorangegangenen Lösung, doch man bleibt dem bewährten, aber etwas starren relationalen Modell treu. Die Gestaltungsfreiheit ist hier nicht gegeben, das Gateway Script formatiert die resultierenden Tabellen auf genau eine Art und Weise.
Viele Lösungen von Dritten, die mit Metadaten arbeiten, bieten recht attraktive Möglichkeiten, so können etwa Beziehungen zwischen Tabellen

COMMON GATEWAY INTERFACE

Das Common Gateway Interface (CGI) wurde ursprünglich entworfen, um HTML-Formulare auszuwerten. Der Benutzer füllt ein HTML-Formular aus und startet mit dem sogenannten «Submit»-Knopf beim WWW-Server ein Gateway Script, das das Formular auswertet und je nach Auswertung eine Meldung generiert. Sind beispielsweise nicht alle benötigten Daten vom Benutzer angegeben worden, so kann das Script eine Fehlermeldung in HTML-Format generieren, die dann vom WWW-Server an den Benutzer weitergereicht wird. Mit dem CGI können aber nicht nur Formulare ausgewertet werden, sondern es können ganz allgemein dynamisch HTML-Seiten beliebigen Inhalts erzeugt werden. Diese Fähigkeit macht das CGI zu einem Dreh- und Angelpunkt bei vielen Web-Seiten, die mehr als nur einfach HTML bieten wollen.

Ein Gateway Script kann in jeder beliebigen Programmiersprache implementiert werden, da die Parameterübergabe entweder via Umgebungsvariablen oder via Standardeingabe (stdin) erfolgt. Die Rückgabe der resultierenden HTML-Seite an den aufrufenden WWW-Server erfolgt per Standardausgabe (stdout). Dabei kann ein Gateway Script nicht nur über ein Formular gestartet werden, sondern auch direct via Uniform Ressource Locator (URL), also genau gleich wie eine normale HTML-Seite. Eventuelle Parameter müssen dann Teil des URL sein.

mit HTML-Verknüpfungen realisiert werden oder es sind Konstrukte zum Suchen, Editieren und Löschen von Tupeln verfügbar. Nichtsdestotrotz bleiben diese Lösungen Stückwerk, vor allem was die Gestaltungsfreiheit in den HTML-Dokumenten anbelangt.

Zelig-HTML

Eine publizierte Lösung, nämlich Zelig von Carlos A. Varela und Caroline C. Hayes (Universität von Illinois), verwendet einen etwas anderen Ansatz. Anstatt aus den Daten komplette HTML-Dokumente zu generieren, wird ein bereits existierendes HTML-Dokument mit den Daten aus der Datenbank ergänzt. Um die Daten sinnvoll in das Dokument einzubeziehen, wird HTML von Zelig um einige Konstrukte erweitert, was als ZHTML bezeichnet wird. Mit den neuen Befehlen können Daten aus der Datenbank angefordert, durch die Daten iteriert und Werte aus der Datenbank in das HTML-Dokument übernommen werden.

Der einfachste und direkteste Weg, um ein ähnliches System aufzubauen, wäre etwa, SQL-Befehle in das HTML-Dokument einzubeziehen und dann auf die resultierende Tabelle mit geeigneten Befehlen zuzugreifen. Der Grad der Datenbankkopplung ist bei dieser Lösung sehr hoch, auch die Abstraktion der Datenbank ist nicht gegeben. Ganz abgesehen davon ist es mehr als nur ein Schönheitsfehler, HTML- und SQL-Befehle zu mischen.

Das Composite Object System

Das Composite Object System wurde im Rahmen eines Forschungsprojekts entwickelt. Gemäss Kapitel Lösungsmodelle basiert es auf dem Modell c: indirekte Kopplung von Web-Server und Datenbank via Gateway.

Konzeptionell ist das Composite Object System an das vorgestellte «Zelig-HTML» angelehnt, erfüllt aber zusätzlich die Forderungen nach Abstraktion der Datenbank und Gestaltungsfreiheit für HTML-Dokumente (Web-Seiten).

Abstraktion der Datenbank

Das Ziel der Abstraktion der Datenbank ist es, vom relationalen Modell zu einem objektorientierten Modell zu kommen, ohne dass die Datenbank umorganisiert werden muss. Warum ist dies sinnvoll? Das relationale Modell benutzt Fremdschlüsselbeziehungen, um mengenwertige Attribute abzubilden; ein Behelfskonstrukt, das den realen Sachverhalt nur unvollkommen abbildet. Ein objektorientiertes Modell entspricht viel mehr der Realität. Daher wird eine Schicht zwischen die Datenbank und die HTML gelegt, die eine objektorientierte Sicht auf die Daten ermöglicht (Bild 2). Die Schicht kann auch andere Ressourcen als eine relationale Datenbank abstrahieren, die Anwendung bemerkt davon nichts.

Die objektorientierte Abstraktion von relationalen Datenbanken ist zurzeit ein vieldiskutiertes Thema, denn rein objektorientierte Datenbanken sind noch nicht in der Lage, die Robustheit und Ausgereiftheit der relationalen Datenbanken zu bieten. Viele Fachleute glauben, dass sich in der nahen Zukunft «Middleware»-Produkte auf dem Markt durchsetzen werden, die zwar ein objektorientiertes Modell bieten, aber auf der bewährten relationalen Technik aufbauen. So können die konzeptionellen Vorteile des objektorientierten Ansatzes für neue Anwendungen genutzt werden und bestehende relationale Anwendungen weiterverwendet werden, ohne dass eine Migration der Daten nötig wird.

Composite Objects

Ein Seiteneffekt, der bei Normalisierungen von Datenbankschemata auftritt, ist, dass logisch zusammengehörende Daten auf mehrere Tabellen verteilt werden. Um diesem Effekt entgegenzuwirken, haben wir die sogenannten Composite Objects kreiert. Der Term «Composite» soll der Tatsache Ausdruck verleihen, dass diese Objekte aus mehreren (atomaren oder komplexen) Bausteinen zusammengesetzt sind. Damit ist auch schon das wichtigste über die Composite Objects gesagt: sie enthalten atomare Werte und geordnete Mengen von Composite Objects, wobei eine Menge von Composite Objects immer Composite Objects der gleichen Art enthält.

Im Idealfall entspricht ein Datentupel

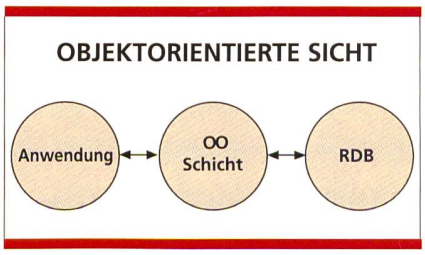


Bild 2. Objektorientierte Abstraktion von relationalen Datenbanken.

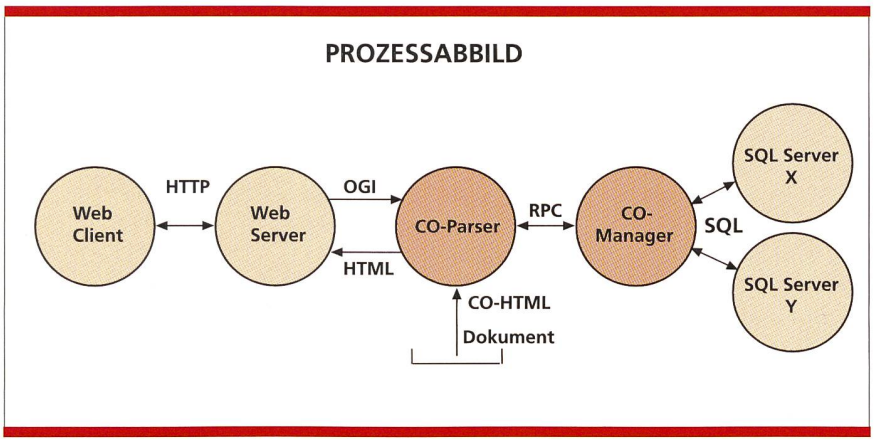


Bild 3. Prozessabbild des CO-Systems.

einer Relation genau einem Composite Object; in der Praxis ist es häufig so, das die Composite Objects aus Attributen von mehreren Tabellen zusammengesetzt sind. Dass ein Composite Object Mengen von Composite Objects enthalten kann, abstrahiert Fremdschlüsselbeziehungen auf einfache Weise. Das relativ simple Objektmodell der Composite Objects hat nicht den Anspruch, eine vollständige objektorientierte Abstraktion der Datenbank zu bieten, sondern es ist ein minimaler Ansatz, der auf jedes relationale Datenbankschema angewendet werden kann.

Composite Object HTML

Um mit den Composite Objects zu arbeiten, haben wir HTML um einige Konstrukte erweitert, die wesentlich einfacher zu handhaben sind als SQL-Befehle. Als Elemente von Composite Object HTML (CO-HTML) werden einerseits die Konstrukte der strukturierten Programmierung Sequenz, Selektion und Iteration verwendet, und anderseits werden einige weitere Konstrukte definiert (siehe Kasten «CO-HTML»).
CO-HTML benutzt benannte Mengen, um Composite Objects zu verwalten. Mit einer Mengendeklaration kann einer Menge von Composite Objects ein Name gegeben werden. Mit dem Iterationskonstrukt kann durch eine benannte Menge iteriert werden und mit der Wertreferenz ein oder mehrere Werte des aktuellen Objekts der Menge in das HTML-Dokument übernommen werden. Die Selektion erlaubt, auf gewisse Sachverhalte, wie zum Beispiel einer leeren Menge, angemessen zu reagieren.

Prozessabbild

Das Prozessabbild des CO-Systems ist aus Bild 3 ersichtlich. Die beiden Prozesse CO-Parser und CO-Manager sind die Teile des CO-Systems. Der CO-Manager ist ein Hintergrundprozess, der die Datenbank abstrahiert und der verantwortlich für die Verwaltung der Composite Objects ist. Er setzt sie anhand der Beschreibung aus den Daten der Datenbank zusammen und stellt sie dem CO-Parser zur Verfügung. Der CO-Parser ist ein Gateway Script, das ein referenziertes CO-HTML-Dokument auf CO-HTML-Befehle parst und bei Bedarf Composite Objects vom CO-Manager anfordert und deren Werte in das resultierende HTML-Dokument einfügt. Der CO-Manager hilft, mit Caching-Strategien das Laufzeitverhalten des Systems soweit zu verbessern, dass für den Web-Client kein Unterschied zu herkömmlichen Web-Seiten feststellbar ist.

Informationsdesign

Beteiligte Personen

Nur in den seltensten Fällen werden die doch recht verschiedenen Aufgabenbereiche Datenbanken und Web von einer einzigen Person betreut, insofern sind mindestens zwei Personen in den Aufbau eines datenbankbasierten Web-Systems involviert. Die erste Person ist der sogenannte «Webmaster», der das Web-System unterhält, und die zweite Person ist der Datenbankadministrator (DBA). Damit sich diese beiden Personen verständigen können, muss jeder einen Teil des Fachgebiets des anderen kennen. Mit dem Konzept der Composite Objects wird eine klare Schnittstelle zwischen Web und Datenbank definiert, die auch die Zusammenarbeit von Webmaster und DBA unterstützt.

Vorgehen

Zuerst wird festgelegt, welche Daten der Datenbank von Interesse für die Öffentlichkeit sind. Dann werden daraus die Composite Objects abgeleitet. Ein Composite Object fasst logisch zusammenhängende Attribute zusammen, dabei spielt es keine Rolle, ob die Attribute atomar oder komplex sind. Die Beschreibung der Composite Objects stellt die Schnittstelle zwischen Web und Datenbank dar. Damit ist der «schwierige» Teil des Systemaufbaus auch schon erledigt. Der Webmaster kann nun wie gewohnt HTML-Dokumente erzeugen und die Daten aus der Datenbank mittels CO-HTML-Befehlen in die Dokumente einfügen und mit allen Mitteln und Möglichkeiten, die HTML zur Verfügung stellt, gestalten. Dabei spielt es keine Rolle, welche Version von HTML benutzt wird, auch zukünftige Erweiterungen können später problemlos angewendet werden. Zum Schluss werden die benötigten Programmteile des Composite Object Systems installiert und gestartet.

Beispiel

Definition der Composite Objects

Anhand eines einfachen Beispiels sollen die Zusammenhänge weiter ver-

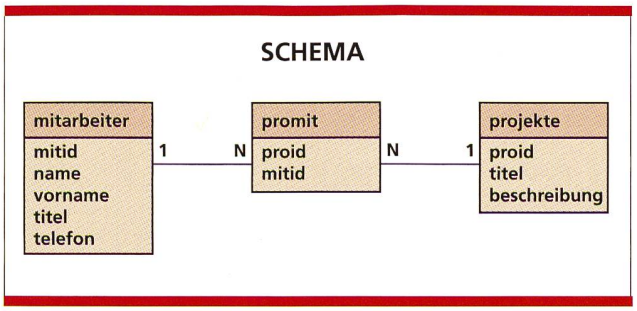


Bild 4. Datenbankschema.

deutlicht werden (Bild 4). Das Datenbankschema drückt folgenden Sachverhalt aus: Ein Mitarbeiter arbeitet in einem oder mehreren Projekten mit, und ein Projekt hat einen oder mehrere Mitarbeiter, also eine klassische N:M-Beziehung. Daraus werden nun die Composite Objects abgeleitet und mit einer für den CO-Manager verständlichen Syntax beschrieben (die fett gedruckten Wörter stammen aus dem Datenbankschema):

```
Mitarbeiter mitarbeiter
{
    string Id          mitid
    string Name        name
    string Vorname     vorname
    string Titel       titel
    string Telefon     telefon
    string Buero       buero
    Projekt            Projekte Id,
                       promit.mitid:proid,projekte.proid
}

Projekt projekte
{
    string Id                  proid
    string Titel               titel
    string Beschreibung        beschreibung
    string Leiter              Id leiter
    Mitarbeiter   Leiter       LeiterId, mitarbeiter.mitid
    Mitarbeiter   Mitglieder   Id,promit.proid:mitid,
                               mitarbeiter.mitid
}
```

Das Composite Object Mitarbeiter basiert auf der Tabelle **mitarbeiter.** Die erste Spalte drückt aus, wie der Wert im HTML-Dokument dargestellt wird. Normalerweise ist dies immer string, nur bei Bildern oder Tondokumenten (BLOBS) wird beispielsweise imggif oder sndaif spezifiziert. Enthält das Feld keinen atomaren Wert, sondern eine Menge von Composite Objects, so wird in der ersten Spalte der Name des Composite Objects angegeben. Die zweite Spalte enthält den CO-HTML-Alias für das Feld. Mit diesem beliebigen Alias wird das Feld in einem CO-HTML-Dokument referenziert. Die dritte Spalte enthält für atomare Werte den Namen des Attributs und für nichtatomare Werte eine Liste mit Schlüsseln und Tabellen, die beschreibt, wie auf die Menge zugegriffen werden muss.

Jedes Composite Object Mitarbeiter enthält alle Projekte, an denen der Mitarbeiter gerade arbeitet, und jedes Composite Object Projekt enthält alle Mitglieder, die an dem Projekt mitarbeiten. Die Instanzierung der Composite Objects ist durch die Daten in der Datenbank bereits gegeben, nun kann mit CO-HTML auf die Objekte zugegriffen werden.

Zugriff auf die Composite Objects mit CO-HTML

Für ein gegebenes Projekt mit der Nummer 315 wird eine kleine Web-Seite aufgebaut und mit Informationen aus der Datenbank ergänzt. Der HTML-Code hat etwa folgendes Aussehen, wobei CO-HTML-Befehle fett gedruckt sind:

```
<HTML>
<HEAD>
<TITLE>Projekt 315</TITLE>
</HEAD>
<BODY>
<!-- CO SET Projekt P -->
 <!-- CO ITER P -->
  <!-- CO IF P.Id – «315» -->
   <H1><!-- CO $P.Titel --></H1>
   <H1><CENTER><!-- CO $P.Titel --></CENTER></H1>
   <P><B>Manager: </B>
   <!-- CO SET Mitarbeiter M : MitId = $P.LeiterId -->
    <!-- CO FREF = <mitdetail.cohtml> SINGLE M -->
     <!-- CO $P.LeiterName --> <!-- CO $P.LeiterVorname -->
    <!-- CO /FREF -->
   <!-- CO /SET -->
```

Bild 5. Web-Seite, ergänzt mit Informationen aus der Datenbank.

```
    <HR>
    <!-- CO SET P.Mitarbeiter PM -->
     <!-- CO IF #PM > «0» -->
      <B>Team: </B><BR>
      <!-- CO ITER PM -->
       <!-- CO FREF = «mitdetail.cohtml» SINGLE PM -->
        <!-- CO $PM.Vorname --> <!-- CO $PM.Name -->
       <! CO /FREF --><BR>
      <!-- CO /ITER -->
      <HR>
     <!-- CO /IF -->
    <!-- CO /SET -->
   <!-- CO /IF -->
  <!-- CO /ITER -->
 <!-- CO /SET -->
 </BOBY>
 </HTML>
```

Dieses Beispiel ist natürlich nur ein Auszug aus einem real existierenden Gesamtsystem, das Mitarbeiter, Projekte und anderes beinhaltet (Bild 5).

Ergänzen des Informationssystems mit Java Applets

Jave ist in erster Linie eine von SUN Microsystems entwickelte, an C++ angelehnte, objektorientierte Programmiersprache, die einige Probleme der modernen Programmierpraxis lösen soll. Auch soll sich die Sprache als Internet Programming Language etablieren. Hier die obligate Hello-World-Applikation:

```
class HelloWorldApp
{
 public static void main (String args[ ])
 {
  System.out.println(«Hello World!»);
 }
}
```

Dieser Code kann kompiliert und als Stand-Alone-Applikation ausgeführt werden. Mit einigen Modifikationen kann diese Applikation aber auch als sogenanntes Applet in eine Webseite miteinbezogen werden. Dabei sind einige Einschränkungen zu berücksichtigen:

- Web-Client muss Applets unterstützen (z. B. HotJava, Netscape 2.0)
- Das Applet wird vom Web-Client interpretiert (Performanceeinbusse)

Um ein Applet zu implementieren, kann auf eine umfangreiche Klassenbibliothek, die von SUN mitgeliefert wird, zurückgegriffen werden. Die geschätzte Einarbeitungszeit beträgt ungefähr sechs Monate. Die Klassenbibliothek ermöglicht innerhalb eines Applets mehr oder weniger alles, was auch mit einer X11- oder Windows-Applikation gemacht werden kann.

Datenbankabfragen mit Java

Um auf Datenbanken zuzugreifen, gelten für Java in etwa die gleichen Spielregeln, wie für jede andere Programmiersprache auch; man benötigt Bibliotheken, die die Schnittstelle zur Datenbank bilden. Die Abfragesprache ist und bleibt SQL.
Eine Java-Applikation unterscheidet sich durch nichts von einer mit herkömmlichen Sprachen programmierten Applikation, ausser dass sie, wenn sie ein Applet ist, vom Web-Server zum Web-Client übertragen und dort interpretiert wird.
Es ist durchaus möglich, dass auch Java Applets den CO-Manager als Schnittstelle zur Datenbank benutzen, wie dies etwa im Modell d beschrieben ist. Applets können auch von der Abstraktion der Datenbank profitieren, und der sogenannte «impedance mismatch», der bei der Interaktion von Programmiersprachen mit Datenbanken auftritt, kann vermieden werden, weil Java eine objektorientierte Sprache ist und der CO-Manager bereits eine objektorientierte Sicht der Datenbank bietet.
Es ist auch denkbar, das CO-HTML und Java-Applets gemischt werden, denn die beiden Technologien sind komplementär und nicht konkurrenzierend. Ein Java Applet zu implementieren ist wesentlich aufwendiger als ein CO-HTML-Dokument zu erstellen. Wenn jedoch beispielsweise Grafiken dynamisch aus Daten der Datenbank erzeugt werden sollen (beispielsweise Organigramm), so ist Java ein geeignetes Mittel.

Ausblick

Im ersten Teil des Berichts haben wir den Web als künftige Drehscheibe für globale Märkte vorgestellt, den Weg dorthin allerdings von einer verbesserten Informationsaufbereitung abhängig gemacht. Wir haben festgestellt, dass aus verschiedenen Gründen die Anbindung von Datenbanken zur Ergänzung des Informationsangebots eines Web-Servers sinnvoll ist.
Die Diskussion der vorgestellten Lösungsmodelle führte zu der Entwicklung eines Gateways, welches die Zielsetzung «Anbindung von Datenbanken» erfüllt, gleichzeitig aber durch Entkopplung mit einer objektorientierten Zwischenschicht die Datenbank abstrahiert und die Gestaltungsfreiheit beim Entwurf von Web-Seiten nicht beeinträchtigt.

Betriebliche Überlegungen

Beim Einsatz des Gateways gilt es, nebst rein technischen Aspekten, in erster Linie die Zusammenarbeit des Web-Masters und des Datenbankadministrators zu beachten. Durch geschicktes Design und Strukturierung erreichen sie eine optimale Verteilung der Informationen auf den Web-Server und die Datenbank.

Technologietransfer

Wie bereits erwähnt, entstand das Gateway im Rahmen eines Forschungsprojekts. Zurzeit ist es an verschiedenen Stellen zu Test- und Demozwecken in Betrieb. Im Sinne des Technologietransfers ist es vorgesehen, das Projekt auszudehnen und so konkrete Einsätze des Gateways in der Praxis zu ermöglichen.
Die Autoren wie auch die Projektleitung sind an den Reaktionen auf den vorliegenden Bericht interessiert und würden Kontakte mit Firmen und Institutionen, die als mögliche Anwender in Frage kommen, begrüssen.

Dokumentation und praktische Einsätze des Gateways

Technische Dokumentation zur Implementierung des vorgestellten Gateways:

http://www.isbe.ch/~wwwinfo/fe315prs.html

Testbetrieb an der Ingenieurschule Bern (ISBE), Anbindung der Sybase-pubs2-Datenbank:

http://www.isbe.ch/~wwwinfo/fe315prs.html

Live-Betrieb bei der Telecom PTT, Forschung & Entwicklung, Anbindung der internen Datenbank:

http://www.vptt.ch/FE-page.html 7

Rudolf Johner studierte Elektro- und Nachrichtentechnik an der Ingenieurschule in Bern und schloss 1982 als Ingenieur HTL ab. Anschliessend besuchte er die Softwareschule Schweiz. Nach einem Aufenthalt in England trat er bei der Telecom PTT im Bereich Paketvermittlung (X.25) ein, und seit 1987 arbeitet er bei der Forschung und Entwicklung und leitet die Fachgruppe Informationstechnologie. Der «objektorientierte Weg» ist für ihn eine grosse Herausforderung. Die Resultate verschiedener Aktivitäten und Projekte, wie das vorgesehene Gateway für den Web, haben einen praktischen Nutzen und geben Antworten auf künftige Problemstellungen.

Beat Liechti studierte berufsbegleitend an der Ingenieurschule Bern (ISBE) und schloss 1996 als Ingenieur Informatik HTL ab. Während des Studiums war er während mehrerer Jahre für Design und Entwicklung von Kommunikationstreibern und Teilen von Prozessvisualisierungssoftware verantwortlich. Seit 1995 ist er als Projektmitarbeiter für die ISBE tätig, wo er im Rahmen eines Projekts zur Anwendung neuer Datentechnologien mit der Telecom PTT Datenbankanbindungen an den World Wide Web entwickelt und implementiert.

SUMMARY

Data bases in the Web

Many people today use the World Wide Web for a large number of applications. Unlike earlier on-line services, the Web has rapidly exceeded the critical levels of available information and, not least for this reason, found acceptance amongst users. The Web is regarded as the basis of future electronic markets and thus the hub of global markets. As a prerequisite for this, however, the quality of information offered must be substantially increased. An extremely high degree of security and topicality as well as a complete range must be guaranteed. Basically we are aiming to improve the processing of information by adding data bases to a Web server's range of information.

As part of a research project, a gateway has been developed for tying data bases to the Web. When implementing a gateway, the abstraction of the data base is of central importance. By choosing this design, we are able to realize an object-oriented intermediate layer and thus the decoupling of the data base at network and SQL level.

For the purposes of technology transfer it is planned to extend the project and thus enable specific operations of the gateway in practice.

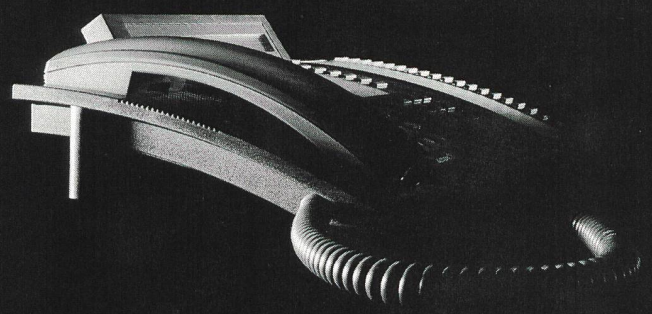
Teilnehmervermittlungsanlage Ascotel bcs 16.

Selbstverständlich können Sie uns die Daten auch via PC schicken...

Amerika? Und ich hätte gewettet, Sie stünden nebenan...

An Ascotel bcs 16 lassen sich bis 24 Endgeräte anschliessen. Vom analogen und digitalen Telefon bis hin zum Fax und PC.

Dank ISDN-Technologie wird mit Ascotel bcs 16 die Übertragungsqualität im Telefonverkehr verbessert.



Ich bin jetzt im Archiv. Wo sagen Sie, liegt das Dossier Zebra genau?

Mit Ascotel bcs 16 lässt sich auch das drahtlose integrierte Telefonsystem bcs cordless betreiben.

Grüezi Frau Muster. Ich verbinde Sie sofort mit Herrn Felix...

Die vielseitigen digitalen Funktionen eröffnen Ihnen völlig neue Möglichkeiten in der Telefonie. Die Anrufer-identifikation ist eine davon.

JABERG + PARTNER

TVA

Für grossen Betrieb im Kleinbetrieb.

Telekommunikation verbindet. Personen und Welten. Zum Beispiel jene Ihrer Kunden und Lieferanten mit jener Ihres Unternehmens. Die Teilnehmervermittlungsanlage Ascotel bcs 16 schafft die Voraussetzungen für perfektes Kommunizieren. Mit der digitalen Technologie (ISDN/SwissNet) kommt eine Fülle von zusätzlichen Komfortfunktionen und multimedialen Möglichkeiten zum Einsatz. Rufen Sie an. Gratis. Wir geben Ihnen gerne weitere Auskünfte über Ascotel bcs 16.

TEL. 155 0 113

TELECOM PTT
Ihre beste Verbindung