

PASTA-3

Eine Schnittstelle zur "Direkten Manipulation" von Wissensbanken

Michel Kuntz, Rainer Melchert, München

Zusammenfassung

Für eine gegebene Wissensbank (Knowledge Base Management System) wird eine Benutzungsschnittstelle entwickelt, die auf dem Prinzip der "Direkten Manipulation" beruht: Wissensbankinhalte, deren Struktur sowie Befehlsmenüs werden am Bildschirm grafisch angezeigt, und Operationen auf ihnen lassen sich durch Selektionen mit der Maus ausführen. Die Formulierung von Suchfragen basiert auf einer zweidimensionalen grafischen Syntax. Dieser Ansatz soll die Orientierung in der Wissensbank erleichtern, die kognitive Belastung der Benutzer verringern (Erinnerung) sowie triviale Eingabefehler weitgehend unterbinden. Ein weiteres wichtiges Merkmal ist die Unterstützung verschiedener Aktivitäten zur selben Zeit ("Task Switching").

1. Ausgangspunkt und Entwicklungsziel

Daten- und Wissensbanken, manipulierbar mit Hilfe formaler Sprachen, entwickeln sich zu immer komplexeren Werkzeugen. Sie finden vermehrt Verwendung in Anwendungsbereichen, in denen es nicht mehr allein um die Speicherung großer Datenmengen geht, sondern zusätzlich um deren Strukturierung (einschließlich Integritätsbedingungen und Ableitungsregeln). Gleichzeitig wächst auch die Komplexität der Interaktion erheblich.

Aus diesem Grund bedeutete es eine Herausforderung, Fortschritte aus der Entwicklung grafikgestützter Dialogsysteme zu nutzen, um eine Benutzungsschnittstelle für Wissensbanken zu entwerfen. Ihre wesentlichen Merkmale sollten sein:

- Abdeckung aller Funktionen der zugrundeliegenden Wissensbank
- Visualisierung des konzeptuellen Modells
- Anpaßbarkeit an gewohnte Arbeitsabläufe
- Dialog basierend auf der Manipulation angezeigter Strukturen

Wissensbanken

Von P.P.Chen wurde das Entity-Relationship-Modell für Datenbanken entwickelt [Chen 76], das Semantischen Netzen ähnelt. Objekte sind als Datentyp "Entität" dargestellt und zwischen ihnen existierende Beziehungen als Datentyp "Relation". Zur Erweiterung des ursprünglichen Modells wurden später Hierarchie-Beziehungen vorgeschlagen ("is_a"). - Ein Anwendungsbereich wird auf der Basis

des vorgegebenen Datenmodells strukturiert. Das Resultat ist ein konzeptuelles Modell oder "Schema".

In den vergangenen Jahren wurden konventionelle Datenbanksysteme zu sogenannten "**Wissensbanken**" (deduktiven Datenbanksystemen/ Knowledge Base Management Systems) erweitert. Sie umfassen zusätzlich Integritätsbedingungen (Constraints) und Regeln zur Ableitung weiteren Wissens aus den gespeicherten Fakten (Deduktionsregeln). Es hängt vom betreffenden System ab, ob Regeln und Bedingungen den Benutzern zugänglich sind oder ihnen verborgen bleiben.

Auf der Basis des erweiterten Entity-Relationship-Modells wurde am ECRC eine Wissensbank, "KB2", entwickelt [Wallace 86], die über eine formale Sprache abfragbar und manipulierbar ist.

Interaktionsarten

Ausgehend von den am weitesten verbreiteten **textuellen formalen Anfragesprachen** gibt es verschiedene Ansätze, den Zugang zu Datenbanken und Wissensbanken zu erleichtern: Benutzungsschnittstellen, die Anfragen in (quasi-) **natürlicher Sprache** zulassen und solche, die eine **grafische Visualisierung** von Strukturen und Daten zur Grundlage der Interaktion machen.

Grafische Schnittstellen sind Gegenstand dieses Beitrags. Sie ermöglichen die Visualisierung des Wissensbank-Schemas und damit dessen Erkundung ("Browsing") durch die Benutzer - eine wesentliche Voraussetzung zur späteren Formulierung sinnvoller Anfragen. Darüber hinaus ist die grafikbasierte Interaktion in den letzten Jahren folgerichtig um den Aspekt der "**Direkten Manipulation**" visualisierter Strukturen und Daten erweitert worden [Shneiderman 83]. Diese Interaktionsart erfordert neben der Tastatur eine Maus als Eingabemedium und einen hochauflösenden Bildschirm als Ausgabemedium.

Benutzerkreis

Wissensbanken sind äußerst komplexe Werkzeuge, deren Bedienung angemessene Kenntnisse von der Anwendung, ihrer internen Repräsentation (Datenmodell) sowie einer Kommandosprache erfordern. Das Vorschalten ergonomischer Benutzungsschnittstellen kann die Interaktion bestenfalls erleichtern. Unsere Erfahrungen mit eigenen Prototypen legen den Schluß nahe, daß es unrealistisch ist anzunehmen, man könne derart komplexe Systeme durch eine Benutzungsschnittstelle auch Nicht-Experten zugänglich machen. Die Zielgruppe von Pasta-3 sind daher **professionelle Nutzer**.

Entwicklungsziel

Auf der Basis einer **zweidimensionalen grafischen Syntax** sollte eine "Direct Manipulation"-Benutzungsschnittstelle entwickelt werden, die den vollen Funktionsumfang der weiter oben erwähnten Wissensbank unterstützt:

- erzeugen und aktualisieren des Schemas/ der Daten
- erkunden des Schemas/ der Daten
- formulieren und auswerten von Anfragen

Großer Wert wurde darauf gelegt, den Benutzern die Freiheit zur selbständigen **Organisation** ihrer **Arbeitsumgebung** (Arbeitskontext) zu geben: In beliebig erzeugbaren Fenstern sollten (Teil-)Aufgaben begonnen, ruhen gelassen und zu einem späteren Zeitpunkt wieder aufgenommen werden können

("Task Switching"). Diese Teilaufgaben sollten voneinander unabhängig ausführbar sein.

Dieses Vorgehen versprach eine geringere Beanspruchung des **Erinnerungsvermögens**, da Anzeigen gleich welcher Komplexität nach Belieben wiederholt oder zum Vergleich nebeneinandergestellt werden könnten, wann immer die Situation es erfordert. Ebenso sollten einmal formulierte Anfragen in ihrer ursprünglichen Form oder als Teile umfassenderer Anfragen wiederverwendbar sein. - Auf eine Änderbarkeit elementarer Schnittstellen-Eigenschaften (wie zum Beispiel der Menü-Anzeigen) wurde verzichtet.

Ausgangsbasis

Pasta-3 ist der dritte, vollständig überarbeitete Prototyp einer Direct-Manipulation-Schnittstelle zu einer Wissensbank am ECRC. Bislang konnten wertvolle Erfahrungen insbesondere mit dem Vorgänger, Pasta-2, gewonnen werden [Melchert 87, VonDerGruen 87]. Er wurde von Ergonomen der Mutterfirmen des ECRC (Bull/ICL/Siemens) informell ausgewertet; auf systematische Versuchsreihen mußte verzichtet werden, da die Antwortzeiten von Pasta-2 realistisches Arbeiten nicht zuließen.

Die Pasta-Systeme wurden auf Arbeitsplatzrechnern mit hochauflösendem Grafikbildschirm und Maus entwickelt.

Ausgangspunkte für die Entwicklung des PASTA-Systems waren bestehende, frühe Ansätze zu graphischen oder Formular-basierten Anfragesprachen für Datenbanken. Während der letzten Jahre wurde in der Literatur über ungefähr 20 solcher Systeme berichtet. Aus Platzmangel kann hierauf nicht näher eingegangen werden; eine Auswahl findet sich im Literaturverzeichnis. Keines der dort genannten Systeme vermag jedoch alle Funktionen einer Datenbank oder einer Wissensbank abzudecken.

2. Grundelemente der Interaktion mit Pasta-3

Das für die Interaktion gewählte Grundprinzip - Direkte Manipulation - setzt die Sichtbarkeit der zu bearbeitenden Strukturen und Daten voraus. Die zur Manipulation der Wissensbank nötige formale Sprache beruht hier auf einer **zweidimensionalen Syntax**. Deren grafische Ausdrücke sind ungerichtet im Gegensatz zur linearen Formulierung in textuellen formalen Sprachen. Es ist daher gleichgültig, in welcher Reihenfolge die grafischen Elemente einer Anfrage aufgebaut werden.

Menüs

Um den Bildschirm nicht durch große, statische Menüs zu überladen, wurden die Befehle gruppenweise auf verschiedene Menüs verteilt. Zentraler Einstiegspunkt zum Dialog ist ein Hauptmenü, das lediglich die Oberbegriffe der verschiedenen Aktionsfelder enthält. In darauffolgenden Menüstufen können die Aktionen dann feiner spezifiziert werden: **hierarchische pop-up-Menüs**. Zwei weitere Menüarten schließen sich an: **statische Menüs**, die lokal in einem Fenster wirken und **pop-up-Menüs**, die an bestimmte Maus-sensitive Strukturen oder Daten innerhalb eines Fensters gebunden sind und bei deren Selektion auftauchen. So wird auf eine permanente Sichtbarkeit aller Befehle verzichtet zugunsten einer dem jeweiligen Dialogzustand angemessenen Auswahl von Befehlen.

Eine weitere Anwendung von Menüs erwies sich als besonders hilfreich bei der Spezifikation von

Daten: **Werte-Menüs**. Soll ein Datenfeld mit einem Wert aus einer bekannten Menge von Alternativen besetzt werden, so können diese automatisch in Form eines Menüs angeboten werden (Abbildung 1).

AC	mixP	DEL	KB2Tr	Answer	2xER	subQ
QUERYING WORKSPACE 4590						
<div>machines</div> <div>hostname</div> <div>machinetype</div>			<div>Possible Values</div> <div>bullsps7</div> <div>bullsps9</div> <div>microvax2</div> <div>mk500</div> <div>perq2</div> <div>sun3-75</div> <div>sun3-140</div> <div>sun3-160</div> <div>sun3-280</div> <div>symbolics3600</div> <div>vax785</div> <div>kerok</div>			

Abbildung 1: Auswahlmenü für eine begrenzte Anzahl von Attribut-Werten

Zusätzlich sind fast alle in Fenstern enthaltenen Namens- und Datenfelder Maus-sensitiv. Ihre Inhalte können ebenfalls zur Spezifikation noch leerer Felder kopiert werden. Auf diese Weise lassen sich Irrtümer und schlichte Tippfehler weitgehend vermeiden.

Bildschirmfenster

Jede über die Menüs eingeleitete Interaktion (Aktualisierung, Erkundung, Anfrageformulierung) spielt sich in einem eigenen Bildschirmfenster ab. Sie kann - unabhängig von anderen - nach belieben ausgeführt, unterbrochen und wieder aufgenommen werden.

Fenster können grafische oder rein textuelle Informationen (Listen) enthalten. Im ersten Fall ist der Fensterinhalt durch zwei Rollbalken, im Falle einer Liste durch einen vertikalen Rollbalken zu bewegen. Jedes Fenster besitzt außerdem eine eigene Menüzeile mit lokal (innerhalb des Fensters) wirksamen Befehlen. Zur Manipulation von Fenstern wurde der Interlisp-D-Window-Manager unverändert beibehalten. Das heißt, Position und Größe der Fenster können variiert, die Fenster selbst zu Piktogrammen geschlossen oder gelöscht werden.

Piktogramme

Die Elemente des Schemas - **Entitäten** und **Relationen** - werden in grafischen Übersichten durch ihre Namen repräsentiert. Entitäten in normalem Font und Relationen in fetter Schrift (vergleiche Abbildung 2). In detaillierteren Anzeigen, zum Beispiel zur Anfrageformulierung, erscheinen die Namen eingerahmt in Rechtecke (vergleiche Abbildung 1). Ihre **Attribute** können nach Wahl in einer Zeile darunter angezeigt werden.

Sämtliche während einer Sitzung verwendeten **Fenster** sind als Piktogramme in einem gesonderten Übersichtsfenster dargestellt. Daraus können sie wieder aufgerufen werden, wenn ihre Bearbeitung vorübergehend unterbrochen war. Durch diesen Mechanismus sind die Benutzer frei, ihren Arbeitskontext selbst zu gestalten. Außerdem lassen sich "Sackgassen"-Situationen (deadlocks)

vermeiden, in denen eine wichtige Information zur Fortsetzung des Dialogs fehlt, die im Falle strenger Linearität aber nicht zwischendurch beschafft werden könnte. Dieser wichtige Dialogschritt wird auch als **"Task Switching"** bezeichnet.

3. Funktion von Pasta-3

Die Interaktion soll für die drei eingangs unterschiedenen Bereiche Erzeugung/Aktualisierung, Browsing und Anfrageformulierung getrennt beschrieben werden, wenngleich viele Dialogelemente in mehr als einem der Bereiche vorkommen.

3.1 Erzeugung und Aktualisierung

Beim Aufbau einer Wissensbank ist die Erzeugung des **Schemas** der erste Schritt. Für das von uns gewählte Datenmodell bedeutet dies :

- Entitäten und Relationen installieren und benennen
- Verbindungen zwischen ihnen ziehen
- Attribute für Entitäten und Relationen vereinbaren
- Regeln und Integritätsbedingungen definieren

Für jede der Aktivitäten gibt es ein spezielles pop-up-Menü. Abbildung 2 zeigt, wie eine Verbindung zwischen der Relation **"use"** (fett gedruckt) und der Entität **"machines"** (in normaler Schrift) gezogen wird. Hierfür steht der Befehl **"Link"** im pop-up-Menü.

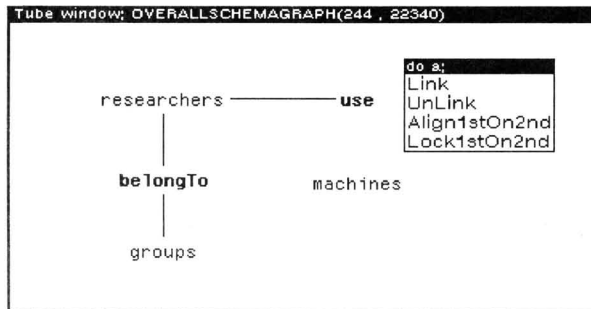


Abbildung 2: Erzeugung eines Wissensbankschemas

Mit demselben Menü lassen sich von Hand gezogene Verbindungen horizontal und vertikal sauber ausrichten ("Align1stOn2nd"). Teilgraphen können zu Gruppen zusammengefaßt ("Lock1stOn2nd") und dann gemeinsam verschoben werden; zum Beispiel **"researchers--belongTo--groups"**. Alle übrigen Verbindungslinien bleiben flexibel und passen sich bei Verschiebungen automatisch an. Der Graph bleibt editierbar, so daß jederzeit Veränderungen möglich sind. - Ein umfangreicheres Beispiel findet

sich in Abbildung 3 im linken oberen Fenster.

Die Definition hierarchischer Beziehungen läuft ähnlich ab und soll daher nicht gesondert beschrieben werden. Abbildung 3 enthält hierzu ebenfalls ein Beispiel (unteres Fenster).

Auch die Attribute der Entitäten und Relationen werden mit Hilfe von pop-up-Menüs definiert. Regeln und Konsistenzbedingungen müssen auf Grund ihrer Komplexität jedoch nach wie vor über die Tastatur eingegeben werden.

3.2 Erkundung (Browsing)

Es ist zweckmäßig, bei der Interaktion zwischen Schema- und Datenebene zu unterscheiden. Während im ersten Fall vorwiegend mit grafischen Repräsentationen gearbeitet wird, bieten sich im zweiten Fall Listendarstellungen an.

Schema

Der Vorgänger von Pasta-3 vereinigte alle Schema-Bestandteile in einem einzigen Graphen: Entitäten, deren Attribute, Relationen sowie die hierarchischen Beziehungen zwischen den Entitäten. Es zeigte sich wiederholt, daß Benutzer des Systems von der Komplexität dieser Anzeige überfordert waren. Der naheliegende Ausweg war eine **Teilung nach Aspekten** (Abbildung 3):

- Entitäten und Relationen
- Hierarchie-Beziehungen
- Attribute von Entitäten

Die Attribute werden bei dieser Lösung nicht permanent angezeigt, sondern erscheinen als pop-up-Menüs bei Selektion einer Entität oder einer Relation.

Alternativ bietet Pasta-3 auch Namenszugriff auf Entitäten und Relationen über alphabetisch geordnete Listen. Die Selektion eines Namens in einer Liste invertiert seine Abbilder auch im Hierarchie- und im ER-Graphen, wie Abbildung 3 für "researchpersonnel" zeigt. Dadurch wird seine Position im Schema deutlich, was besonders bei großen Schemagraphen wichtig ist.

Zur Erkundung großer Schema-Graphen sind zwei weitere Einrichtungen vorgesehen: Es können beliebig viele Fenster für das Browsing gleichzeitig geöffnet sein. So lassen sich verschiedene Ausschnitte aus dem Schema miteinander vergleichen. Außerdem kann bei Vorgabe zweier Endpunkte im Schemagraphen die dazugehörige Verbindung über Relationen und Hierarchie-Beziehungen automatisch gefunden werden. Dazu dienen die Befehle "E-R Links" und "Is-a Links" in den statischen Menüs über den Titelzeilen der Browsing-Fenster.

In einer Wissensbank umfaßt das Schema auch **Integritätsbedingungen** und **Ableitungsregeln**. Sollen sie den Benutzern zur Orientierung zugänglich sein, so muß für sie im Rahmen des Browsing ebenfalls eine Darstellung gefunden werden. Regeln und Bedingungen sollten zusammen mit den Entitäten und Relationen präsentiert werden, auf die sie sich beziehen. Hierzu werden eines oder

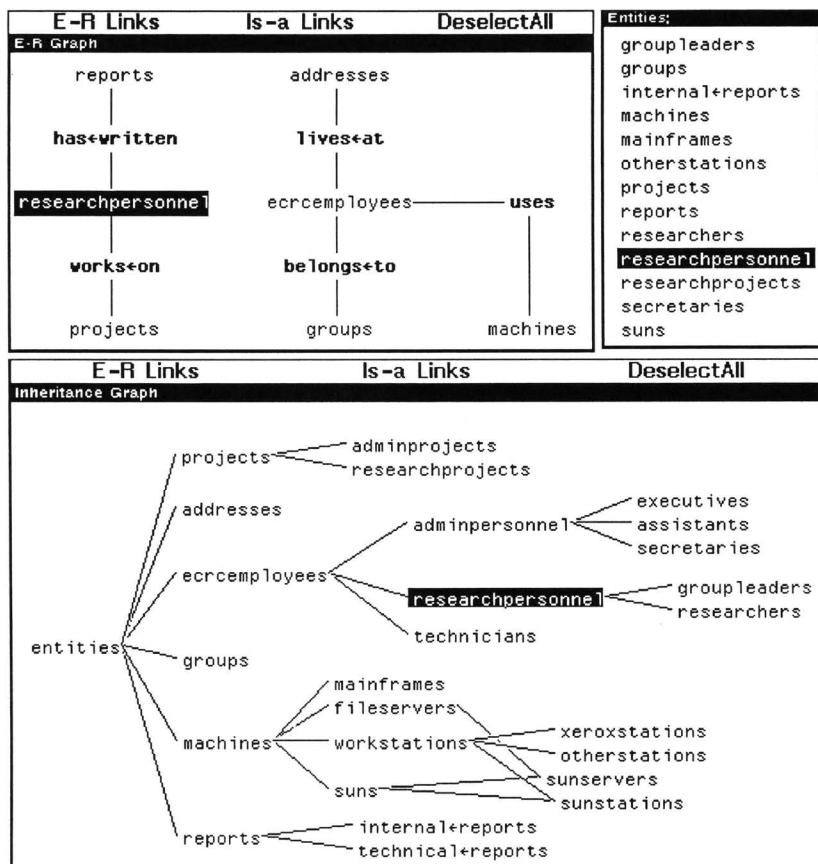


Abbildung 3: Anzeigen zur Erkundung des Schemas

mehrere Schema-Elemente als Auswahlkriterien selektiert. Die relevanten Regeln und Bedingungen werden angezeigt:

- in interner Darstellung - als textuelle Definition
- oder
- in grafischer Darstellung - als Beziehungsgraph

Im Idealfall lassen sie sich als Beziehungsgraph anzeigen, wie unter anderem von [Poltrock 86] vorgeschlagen wurde. Für Pasta-3 ist zur Zeit allerdings nur eine textuelle Anzeige in Form von Listen geplant.

Daten

Das Browsing im Datenbestand wird in Pasta-3 durch wenige Selektionen eingeleitet: Aus einem pop-

up-Menü werden die Attribute einer Entität selektiert, für die Daten ausgegeben werden sollen. Es erscheint danach neben der betreffenden Entität eine Liste, deren Titelzeile die Namen der selektierten Attribute trägt. Die Spalten darunter enthalten die entsprechenden Daten. Da die Liste in ihrer Länge begrenzt ist, können sämtliche Daten in der Liste durch Blättern ("Scrolling") erreicht werden.

3.3 Anfragen (Queries)

Anfragen stellen den Kernbereich der Interaktion mit Pasta-3 dar. Entsprechendes Gewicht wurde auf den Entwurf einer **zweidimensionalen grafischen Syntax** gelegt, die die volle Ausdrucksfähigkeit der ursprünglichen formalen Anfragesprache besitzt.

Anfrageformulierung

Aus einem Browsing-Fenster (Abbildung 3) werden die gewünschten Entitäten und Relationen in ein Anfrage-Fenster kopiert (Abbildung 4). Zu jeder von ihnen können aus pop-up-Menüs die interessierenden Attribute gewählt werden. Solche, die später Teil des Ergebnisses sein sollen, werden durch Fragezeichen markiert. Andere dienen zur Formulierung von Bedingungen.

Im einfachsten Fall soll ein Attribut einen bestimmten Wert haben. Dieser wird in das leere Feld unterhalb des Attributnamens eingetragen und legt es damit fest. Hier sind Werte-Menüs hilfreich: Sie bieten die in Frage kommenden Alternativen an und verhindern so sinnlose oder fehlerhafte Eingaben (siehe Abbildung 1).

Komplexere Bedingungen werden getrennt formuliert. Sie sind aus drei Teilen zusammengesetzt: Operand1--Operator--Operand2 (Abbildung 4, rechts oben). Hier lassen sich Attribute über verschiedene Vergleichsoperatoren an Werte oder sogar an andere Attribute binden. Auch **logische Operatoren** sind eingeführt (And/Or/Not), um Bedingungen miteinander zu verknüpfen. Jeder logische Operator bildet zusammen mit seinen Operanden (rechts anhängende Operatoren und Bedingungen) einen Teilausdruck. Dieser kann, ebenso wie einzelne Bedingungen, herausgetrennt und an anderer Stelle wieder eingefügt werden. Zur Verschiebung solcher Teilstrukturen dienen die kleinen schwarzen Quadrate, die sich mit der Maus "greifen" lassen.

Auch andere Elemente, wie Quantoren, sind bei der grafischen Syntax berücksichtigt, können hier aber aus Platzmangel nicht behandelt werden.

Wichtiges Merkmal der Formulierung grafischer Anfragen ist ihre **permanente Editierbarkeit**. Damit sind die Nachteile der sequentiellen Entstehung textuell formulierter Anfragen überwunden. Jedes Element einer grafischen Anfrage läßt sich jederzeit auswechseln oder verändern. So kann die gewünschte und korrekte Form einer Anfrage gegebenenfalls auch experimentell gefunden werden.

Ergebnisanzeige

Um einmal formulierte Anfragen mehrfach verwenden und modifizieren zu können, wurden die Ergebnisanzeigen von den Anfrage-Graphen getrennt (siehe Abbildung 4). So kann eine Anfrage mehrfach modifiziert und die nacheinander entstandenen Ergebnisse miteinander verglichen werden. Außerdem läßt sich zu jedem Zeitpunkt der Anfrageformulierung der entsprechende formalsprachliche

AC	mixP	DEL	KB2Tr	Answer	2xER	subQ																							
QUERYING WORKSPACE 4718																													
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin-bottom: 10px;">machines</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 25%;">location</th> <th style="width: 25%;">hostname</th> <th style="width: 25%;">machinetype</th> </tr> </thead> <tbody> <tr> <td>??</td> <td>??</td> <td>??</td> </tr> </tbody> </table>				location	hostname	machinetype	??	??	??	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td rowspan="4" style="width: 5%; text-align: center; vertical-align: middle;">A N D</td> <td style="width: 5%; text-align: center;">O</td> <td style="width: 15%;">location</td> <td style="width: 5%; text-align: center;">=</td> <td style="width: 20%;">room#219</td> </tr> <tr> <td style="text-align: center;">R</td> <td>location</td> <td>=</td> <td>room#220</td> </tr> <tr> <td style="text-align: center;">O</td> <td>machinetype</td> <td>=</td> <td>sun3#140</td> </tr> <tr> <td style="text-align: center;">R</td> <td>machinetype</td> <td>=</td> <td>xerox</td> </tr> </table>			A N D	O	location	=	room#219	R	location	=	room#220	O	machinetype	=	sun3#140	R	machinetype	=	xerox
location	hostname	machinetype																											
??	??	??																											
A N D	O	location	=	room#219																									
	R	location	=	room#220																									
	O	machinetype	=	sun3#140																									
	R	machinetype	=	xerox																									
<div style="background-color: black; color: white; padding: 2px; font-weight: bold;">Answer:</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 25%;">hostname</th> <th style="width: 25%;">location</th> <th style="width: 25%;">machinetype</th> </tr> </thead> <tbody> <tr> <td>janus1</td> <td>room#220</td> <td>sun3#140</td> </tr> <tr> <td>janus2</td> <td>room#220</td> <td>sun3#140</td> </tr> </tbody> </table>				hostname	location	machinetype	janus1	room#220	sun3#140	janus2	room#220	sun3#140	<div style="background-color: black; color: white; padding: 2px; font-weight: bold;">KB2 Translation:</div> <pre>(hostname(+491 ,+492) ,location(+491 ,+498) ,machinetype(+491 ,+501)) ,((+501=xerox ;+501=sun3#140) ,(+498=room#220 ;+498=room#219)) ,machines(+491)</pre>																
hostname	location	machinetype																											
janus1	room#220	sun3#140																											
janus2	room#220	sun3#140																											

Abbildung 4: Anfrage : "Welche Maschinen des Typs 'xerox' oder 'sun3_140' stehen in den Räumen 219 oder 220 ?"

Ausdruck in einem separaten Fenster abrufen. - Alle so entstandenen Fenster können, wie schon beschrieben, zur späteren Wiederverwendung als Piktogramme zwischengespeichert werden. Fenster, die Anfragen enthalten, lassen sich als Bausteine in komplexere Anfragen integrieren.

4. Zusammenfassung

Nach umfangreichen Erfahrungen mit Vorversionen wurde die Implementierung von Pasta-3 im Sommer 1988 begonnen. Zielgruppe waren von Beginn an professionelle Nutzer von Wissensbanken.

Pasta-3 ist eine grafische Benutzungsschnittstelle, deren Ausdrucksfähigkeit sich nicht auf die bloße Anzeige von Daten und Strukturen beschränkt. Folglich können alle grafischen Elemente (und nicht nur Menüs) mit Hilfe der Maus direkt manipuliert werden. Wesentliche Grundlage für die Anfrageformulierung ist eine zweidimensionale grafische Syntax. Werte-Menüs tragen im gleichen Zusammenhang erheblich zur Vermeidung von Eingabefehlern bei. Im Rahmen des "task switching" können Anfragen jederzeit editiert, wiederholt oder ausgewertet werden, ohne dabei andere Aktivitäten zu behindern.

Im Januar 1989 waren die wesentlichen Funktionen implementiert. Weitere, wie zum Beispiel die Wiederverwendung von Anfragen als Bestandteile komplexerer Anfragen, befinden sich in der Entwicklung.

Literatur

- [Bryce 86] Bryce, D.; Hull, R.
SNAP: A Graphics-based Schema Manager.
In *Proc. of the IEEE 2nd Int. Conf. in Data Engineering, Los Angeles*, pages 151-164.
1986.
- [Catarci 88] Catarci, T.; Santucci, G.
Query By Diagram: A Graphic Query System.
In *Proc. of the 7th Int. Conf. of Entity Relationship Approach, Rome, Italy*. The ER
Institute, 1988.
- [Chen 76] Chen, P. P.
The Entity-Relationship Model: Towards a Unified View of Data.
ACM Transactions on Database Systems 1(1):9-36, March, 1976.
- [Goldman 85] Goldman, K. J.; Goldman, S. A.; Kanellakis, P. C.; Zdonik, S. B.
ISIS: Interface for a Semantic Information System.
In *Proceedings of ACM-SIGMOD 1985 Int. Conf. of Management of Data, Austin,
Texas*, pages 328-342. May, 1985.
- [McDonald 75] McDonald, N.; Stonebraker, M.
CUPID - The Friendly Query Language.
In *Proc. of the 1975 ACM PACIFIC, San Francisco*, pages 127-131. ACM, April,
1975.
- [Melchert 87] Melchert, R.
Graphische Unterstuetzung beim Umgang mit Wissensbanken.
In *Datenbanksysteme in Buero, Technik und Wissenschaft (BTW '87); GI-
Fachtagung, Darmstadt*, pages 100-115. Gesellschaft fuer Informatik, April,
1987.
- [Poltrack 86] Poltrack, S. E.; Steiner, D. D.; Tarlton, P. N.
Graphic Interfaces for Knowledge-Based System Development.
In *Proceedings of the CHI'86*, pages 9-14. ACM SIGCHI, April, 1986.
- [Shneiderman 83] Shneiderman, B.
Direct Manipulation: A Step Beyond Programming Languages.
IEEE Computers (16):57-69, 1983.
- [VonDerGruen 87] von der Gruen, W. O.; Melchert, R.
Evaluative Comments on PASTA-II and Suggested Experiments for PASTA-III.
Technical Report IR-MMI-08, ECRC GmbH, Munich, W. Germany, 1987.
- [Wallace 86] Wallace, M.
KB2: A Knowledge Base System Embedded in Prolog.
Technical Report KB-12, ECRC GmbH, Munich, W. Germany, 1986.
- [Williams 84] Williams, M. D.
What makes RABBIT run ?
Int. Journal of Man-Machine Studies 21:333-352, 1984.
- [Wong 82] Wong, H. K. T.; Kuo, I.
GUIDE: Graphic User Interface for Database Exploration.
In *Proc. of the 8th Very Large Databases Conf., Mexico*, pages 22-32. 1982.

Adresse der Autoren:

Michel Kuntz, Rainer Melchert

ECRC

European Computer-Industry Research Centre GmbH

Arabellastraße 17, 8000 München 81

e-mail (UUCP): unido!ecrcvax!kuntz