

Graphisch-interaktive Manipulation komplexer Wissensstrukturen mit dem Framenetz-Editor TOPOGRAPHIC I¹

R. Hammwöhner und U. Thiel

Universität Konstanz Informationswissenschaft

Projekt TOPOGRAPHIC

Mit diesem Beitrag soll die graphisch-interaktive Benutzerschnittstelle TOPOGRAPHIC des wissensbasierten Textkondensierungssystems TOPIC vorgestellt werden. Ein Prototyp des Interface erlaubt die Visualisierung komplexer Datenstrukturen und stellt Editierfunktionen auf frame-basierten Wissensnetzen bereit.

1. TOPIC/ TOPOGRAPHIC : ein experimentelles Informationssystem mit graphisch interaktiver Benutzerschnittstelle

Das Ziel des Gesamtprojektes TOPIC / TOPOGRAPHIC ist die Entwicklung eines experimentellen volltextorientierten Informationssystems (vgl. /2/). Fachtexte aus einem eingeschränkten Diskursbereich (hier: Anwendungen von Mikrocomputern in Büroumgebungen) sollen einer automatischen, wissensbasierten Analyse unterzogen werden, aus der sogenannte Textkondensate (Repräsentationen der thematischen Struktur und der Fakteninformation des Textes) hervorgehen. Die so gewonnene Textinformation wird als Ergebnis der Analyse in einer Wissensbasis abgelegt, auf die mit unterschiedlichen Retrievaloptionen (Überblick, Faktenabfrage etc) zugegriffen werden kann.

Das durch diese Aufgabenbeschreibung skizzierte Gesamtsystem läßt sich in zwei Hauptblöcke aufteilen:

- Erschließung des Inhalts von Texten durch textspezifische (/3/) Parsing- und Repräsentationskomponenten und flexibles Kondensieren der Textinformation (TOPIC)
- Auf Texte bezogene Verwaltung (Aufbau und Pflege) von Wissen und graphisch-interaktiver flexibler Zugriff auf die Wissensbestände

Thema dieses Beitrags sind die Editierfunktionen von TOPOGRAPHIC. Dazu werden zunächst die strukturellen Eigenschaften der Wissensbasis erörtert und die Systemphilosophie von TOPOGRAPHIC erläutert. Anhand von Beispielen wird ein realisierter Prototyp vorgestellt, dessen Design unter einem ergonomischen Aspekt diskutiert wird.

1.1 Die Wissensbasis

Das TOPIC-System basiert auf einer Weltwissensbasis, die zur Erschließung der Textinhalte während des Parsing dient, und auf einer die Resultate der Analyse speichernden Textwissensbasis. Zur Repräsentation sowohl des Welt- wie auch des Textwissens wird ein

¹ Dieser Text ist erschienen in: Bullinger, H.-J. (Hrsg.): Software-Ergonomie ,85. Mensch-Computer-Interaktion. Tagung III des German Chapter of the ACM am 24. und 25.9.1985 in Stuttgart, S. 119-128.



This text is published under the following Creative Commons License: Attribution-NonCommercial-NoDerivs 2.0 Germany (<http://creativecommons.org/licenses/by-nc-nd/2.0/de/>).

Wissensrepräsentationsmodell verwendet, dem der von MINSKY /6/ vorgeschlagene Frame-Konstrukt zugrunde liegt. In diesem Ansatz wird eine Wissensseinheit (Begriff) nicht isoliert betrachtet, sondern unter Berücksichtigung ihres semantischen Kontextes.

Ein Frame besteht aus einem Namen und einer Menge von Slots. Ein Slot wiederum besitzt einen Namen und eine (möglicherweise leere) Menge von Slotenträgen. Welche Einträge potentiell erlaubt sind, spezifiziert eine dem Slot zugeordnete Konsistenzregel. Ein Beispiel:

LILITH			(Framename)
cpu	Hauptspeichergröße	Peripherie	(Slotnamen)
	128 Kwords	Tastatur Maus Graphikbildschirm	(Sloteinträge)

Neben unstrukturierten Strings können auch Frames als Slotenträge auftreten, die ihrerseits eine Slotmenge besitzen. Dies erlaubt eine strukturierte Modellierung der durch Slots erfaßten Aspekte (Bestandteile, Eigenschaften etc.) des Begriffs.

In TOPIC/TOPOGRAPHIC werden die Repräsentation von Konzepten als Framestrukturen und die in semantischen Netzen verwendete Relationierung der Begriffe kombiniert (vgl. /8/). Zwischen den Frames können Relationen bestehen, die zusätzliches Weltwissen repräsentieren, z. B. is-a (Ober-/Unterbegriff), inst (Instanz-von). Diese Relationen werden vom System bei der Eingabe eines neuen Frame in das System automatisch erweitert, d.h. wenn spezifische formale Bedingungen erfüllt sind, werden die Relationskanten generiert, die den Frame in das bestehende Netz integrieren. So wird z.B. ein neu hinzugefügter Frame durch is-a-Kanten mit allen Frames verknüpft, deren Slotmenge eine Teilmenge der Slots des neuen Frames darstellt. Ein Frame ohne aktuelle Slotenträge wird als Prototyp bezeichnet, legt man eine Kopie von ihm an, die zusätzlich noch aktuelle Slotenträge enthält, so wird sie mit ihm durch die Instanzrelation (inst) verbunden.

Es ist aber auch möglich, nicht aus der formalen Struktur herleitbare Beziehungen (z.B. Rollenkonzepte: Ein PC kann als (intelligentes) Terminal benutzt werden) oder aus dem Text abgeleitete Relationen zu modellieren, z. B. is-compared-with (Text enthält den Vergleich zweier Objekte). Die Relationen zwischen Frames wirken sich auf die Zulässigkeit potentieller Slotenträge aus, ist nämlich ein Frame als Eintrag möglich, so können auch seine Unterbegriffe (Spezialisierungen) eingetragen werden. Die Komplexität der Wissensstrukturen erfordert eine Unterstützung der Manipulation dieser Daten durch Softwareinstrumente, die im Projekt TOPOGRAPHIC entwickelt werden und auf interaktiver Graphik basieren.

1.2 Graphische Interaktionsformen für Wissensakquisition und Information Retrieval

Die Benutzerschnittstelle des Informationssystems, die im Teilprojekt TOPOGRAPHIC entwickelt wird, soll folgende Aufgaben erfüllen:

- Graphisch-interaktive Komponente zur Navigation in Wissensnetzen
- Graphisch-interaktive Editier- und Retrievalfunktionen
- Präsentation von Textwissen auf unterschiedlichen Abstraktionsniveaus.

Damit werden bisherige Ansätze zur graphischen Unterstützung von Editier- und Retrievalfunktionen (vgl. /5/,/11/,/12/) in die Gesamtkonzeption eines graphisch-interaktiven Informationssystems integriert. Als Alternative zur Auffassung, die Ausgabe eines

Textkondensierungssystem sollte in natürlichsprachlicher Form (Abstracts) angeboten werden, wird eine Präsentation der extrahierten Textinformation mit Mitteln der graphischen Interaktion (Menüanwahl mit der Mouse, Windowtechnik) realisiert. Dies hat folgenden Hintergrund:

- a) Zur Darstellung komplexer, stark vernetzter Strukturen, wie sie für TOPICs Welt- und Textwissen charakteristisch sind, eignen sich Graphiken besonders gut (vgl. /1/) und sind eventuell sogar textuellen Darstellungsformen überlegen, weil sie Textstrukturen transparent machen.
- b) Spezielle Techniken erlauben eine Anpassung des Informationsniveaus an die Bedürfnisse des Benutzers und erleichtern ihm die Orientierung:
 - Das Navigieren im Wissensnetz erlaubt ein zielgerichtetes Aufsuchen der relevanten Information.
 - Zooming-Techniken gestatten die Ausgabe der Daten mit unterschiedlichen Abstraktionsgraden. Optionen dieser Art sind in Texten nicht realisierbar und stellen eine eigene Variante der Informationsvermittlung dar.

Damit leistet TOPOGRAPHIC einen Beitrag zur Entwicklung der von (/4/) vorgeschlagenen graphisch-interaktiven Kommunikationsform.

2. Der Framenetz-Editor: eine prototypisch realisierte Systemkomponente

Da sich der Wert von Präsentationsformen nur im praktischen Test ermitteln läßt, werden die in der Konzeptionsphase definierten Prinzipien der Dialoggestaltung in einem Prototyp implementiert, der bei eingeschränktem Funktionsumfang gestattet, Verfahren zur Netzpräsentation und -navigation zu erproben. Er realisiert Editierfunktionen, die in der derzeitigen Projektphase zum Aufbau des Weltwissens benötigt werden, und dient zur Evaluierung von Darstellungsmethoden.

Der Editor, realisiert auf einer LILITH-Workstation, basiert auf der an der ETH Zürich entwickelten Benutzerschnittstelle XS-2 (/9/), die Werkzeuge zur prototypischen Gestaltung von Kommandostrukturen und zur Präsentation hierarchisch organisierter Daten zur Verfügung stellt. Über eine V24-Schnittstelle wird die LILITH-Workstation an einen Unix-Rechner gekoppelt, auf dem das Weltwissen verwaltet wird. Der Editor umfaßt folgende Funktionen:

- a) Graphische Präsentation der Wissensstrukturen
- b) Orientierungsmöglichkeiten durch:
 - Navigation in Relationen
 - Zooming zum Wechseln des Abstraktionsniveaus frame - slot – entry
- c) Einfügen und Löschen von Elementen der Wissensbasis.

Relationale Vernetzungen der Frames sind i.a. gerichtete Graphen, sie werden jedoch in hierarchische Teilnetze partitioniert und als Baumstruktur präsentiert, wobei Frames auch mehrfach dargestellt werden, falls dies notwendig ist (z.B. in polyhierarchischen Strukturen, in denen ein Knoten mehrere Väter haben kann). Damit ist eine kohärente Darstellungsform auf allen Ebenen des Framedatenmodells (/8/) möglich. (Die Slots eines Frames erscheinen als dessen Söhne, sind aber ihrerseits Väter ihrer Sloteträge).

2.1 Die Systemoberfläche des Editors

Der Editor ist als Applikation in das XS2-System integriert, das eine Benutzerführung im Sinne des "Sites, Modes and Trails"-Modells (s. /7/) ermöglicht. Die unterschiedlichen Informationen, die dem Benutzer über den aktuellen Zustand des Systems angeboten werden, sind speziellen Fenstern zugeordnet (s. Abb.1). Während sich die Dialoghistorie bei Bedarf aus den im oben liegenden Trail-Fenster rekonstruieren läßt (s./9/), zeigt das Site-Fenster den jeweils aktuellen Ausschnitt aus dem Weltwissen. Im Kommando-Fenster erscheint ein Fragment des Kommandoabums, das den Kontext des aktuellen Kommandos enthält. Die Abb. 1 illustriert die Startsituation eines Editierdialogs: Die Verbindung zum Hintergrundrechner ist aufgebaut und der Editor erwartet Angaben zur Expansion eines Weltwissensfragmentes.

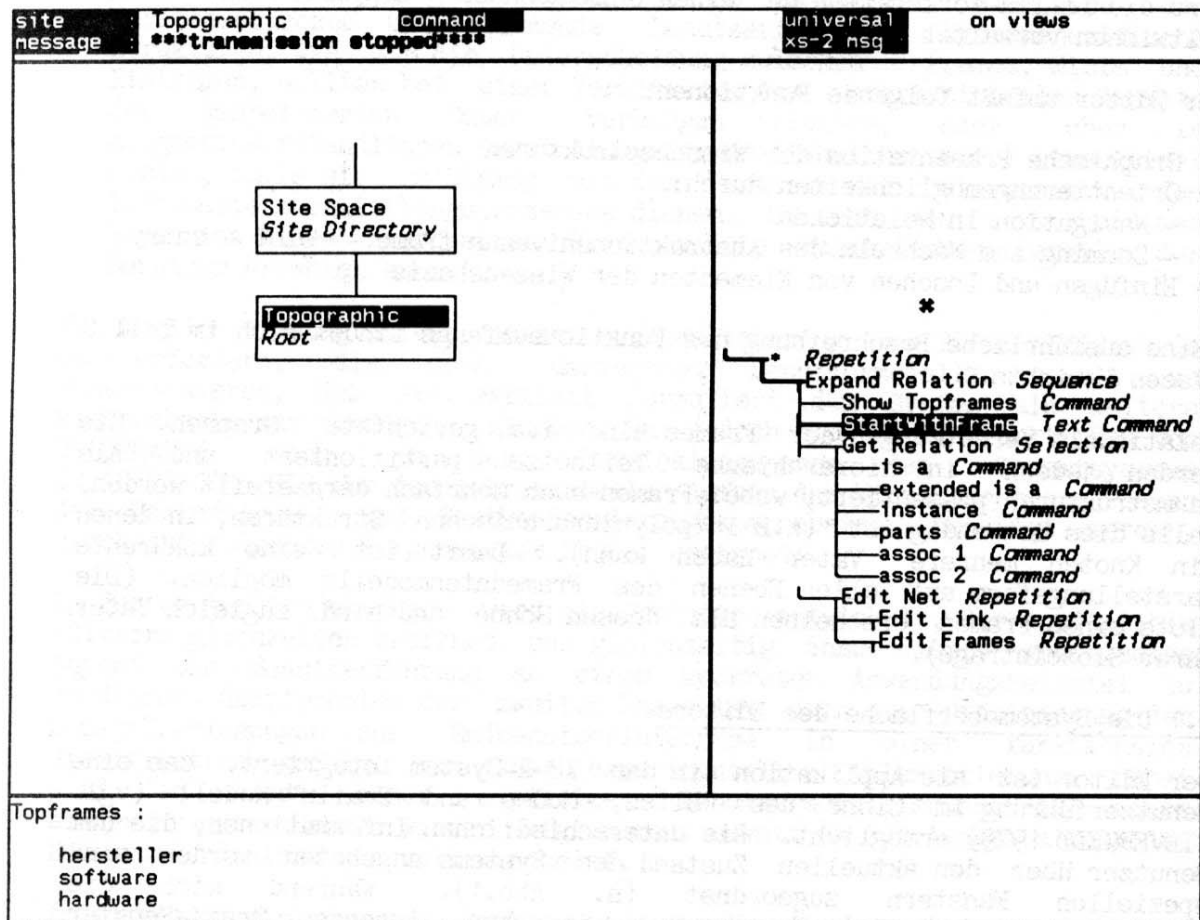


Abbildung 1

Nach Eingabe eines Frames als 'Einstiegspunkt' muß eine der angegebenen Relationen gewählt werden. Zur Unterstützung des Benutzers werden die 'Top-frames', d.h. Frames ohne (im System modellierten) Oberbegriff, in einem weiteren Fenster aufgelistet, dessen Funktion die Darstellung der wichtigsten Informationen über das gezeigte Wissensfragment ist.

Abb.2 zeigt den Dialogzustand nach Angabe des Frames 'Software' als Einstieg in die Relation 'extended-is-a', die als transitive Hülle der Vereinigung der Relationen 'is-a' und 'inst' definiert ist. Sämtliche Unterbegriffe und Instanzen finden sich im Baum mit der Wurzel 'Software', dargestellt im Site-Fenster. Als aktueller Frame wird 'Systemsoftware' hervorgehoben, die Möglichkeiten, ihn zu manipulieren, entnimmt der Benutzer dem Kommandoabum:

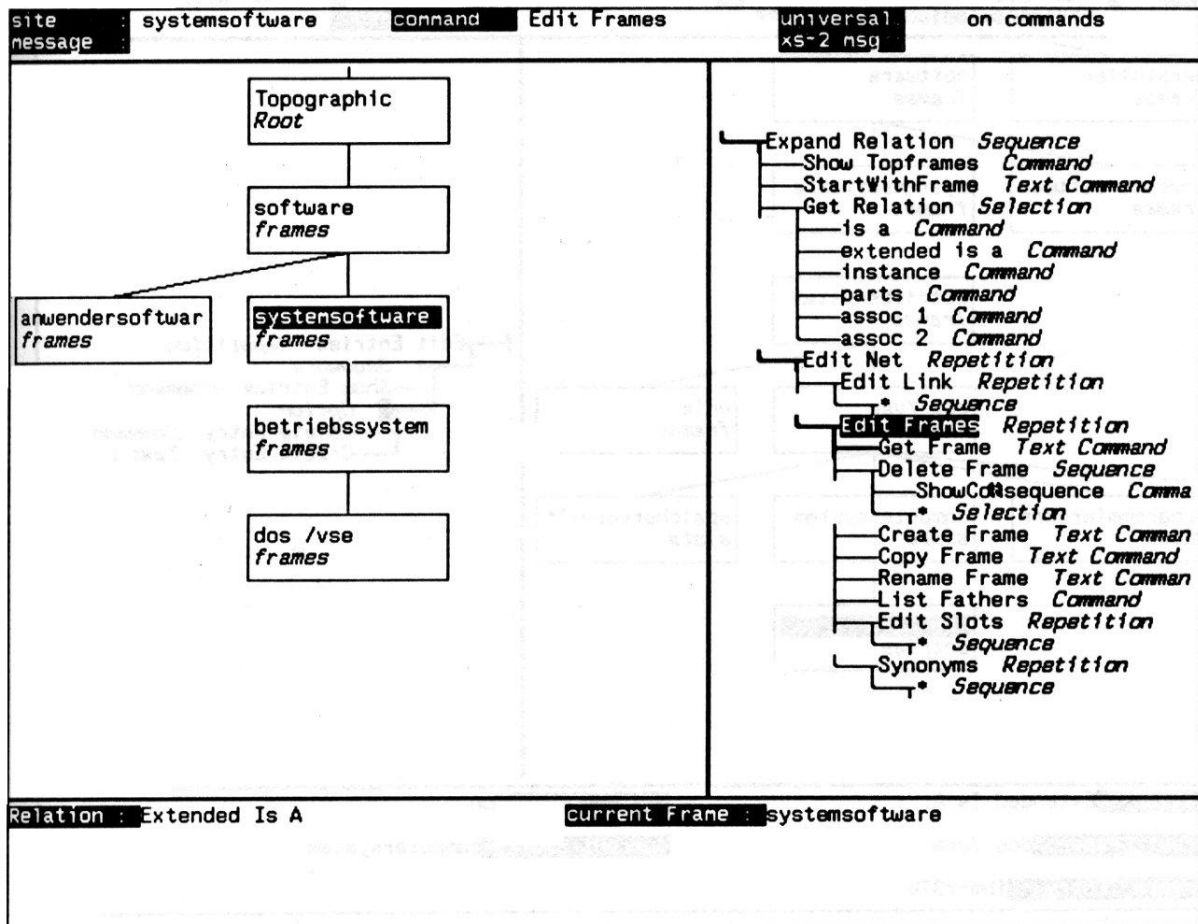


Abbildung 2

Der Einstieg in das Untermenü 'Edit Frames' ist bereits erfolgt. Neben dem direkten Zugriff auf einen im Site-Fenster nicht sichtbaren Frame ('Get Frame') kann der jeweils aktuelle Frame (durch Navigation im Site-Fenster bestimmt) gelöscht, kopiert oder umbenannt werden, eine Liste seiner Vaterknoten (von denen nur einer graphisch dargestellt ist) kann angefordert oder schließlich seine Slots expandiert werden. Auch das Editieren von Synonymen unterstützt das System. Das 'Create'-Kommando erlaubt das Anlegen eines völlig neuen Frames, der jedoch zunächst als Top-frame betrachtet wird, bis durch das Hinzufügen von Slots eine automatische Verrelationierung stattfindet.

Auf der Slotebene finden sich analoge Kommandos zum Löschen und Einfügen von Slots. Das Editieren von Einträgen und Konsistenzregeln erfordert einen weiteren Wechsel der Navigationsebene, hierfür sind wiederum Untermenüs implementiert. Abb. 3 zeigt den expandierten Eintrag des Slots 'computersystem', der eine spezifische Eigenschaft des Konzeptes 'dos/vse' charakterisiert ('dos/vse' ist also ein Betriebssystem für Rechner des Typs 'ibm-370'). Im Statusfenster wird signalisiert, daß dieser Eintrag nicht terminal, d.h. seinerseits ein Frame ist. Wechselt der Benutzer ins Framemenü, so kann er diesen Frame wiederum manipulieren.

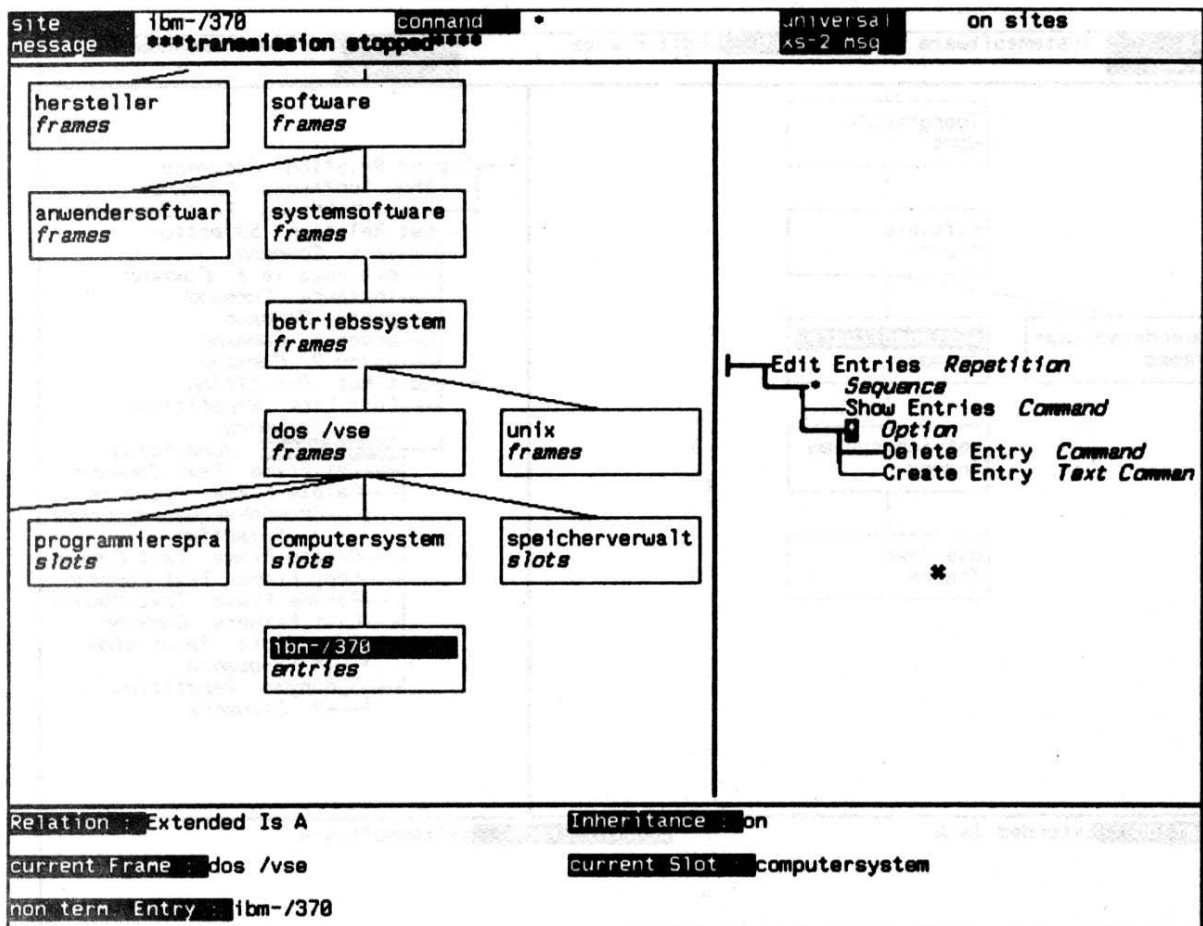


Abbildung 3

2.2 Ergonomische Design-Kriterien

Die Ausgestaltung der Funktionsbereiche Präsentation, Navigation und Manipulation basiert auf den folgenden Prämissen:

a) Präsentation von Wissensfragmenten

- Semantische Netzwerke werden im allgemeinen als gerichtete Graphen dargestellt. Im Kontext eines Editiervorgangs jedoch erscheint es sinnvoll, die Konzentration des Benutzers auf die 'unmittelbare' Umgebung eines Konzepts zu lenken, das zumeist entweder als Verallgemeinerung (Oberbegriff) oder als Spezialisierung eines bereits bestehenden Begriffs editiert wird. Somit bietet die baumartige Präsentation der Konzepthierarchie - Unterbegriffe werden stets unterhalb ihrer Oberbegriffe angeordnet - eine vereinfachte Darstellung des Kontextes, die jedoch wesentliche Informationen zur Einordnung des neuen Begriffs beinhaltet. Die endgültige Version des Interface, die dann primär den Zugang zu Textwissensstrukturen ermöglichen soll, wird nicht-hierarchische Relationen und aus unterschiedlichen Relationen zusammengesetzte Framenetze darstellen können.
- Auf allen Ebenen des Weltwissens (d.h. Netz-, Frame-, Slot-, Entry-, und Bedingungebene) wird eine einheitliche Darstellungsform verwendet, denn der Benutzer kann aus dem jeweils aktiven Kommando sofort erschließen auf welchem Detaillierungsniveau er sich gerade befindet und sollte deshalb nicht durch unterschiedliche Präsentationsstile irritiert werden.

- Die Bedeutung der Kanten ergibt sich aus den Typen der verknüpften Knoten und den Hinweisen im Statusfenster, so ist z.B. eine Kante, die zwei Frames verbindet, Teil der jeweils dargestellten Relation, während eine Kante zwischen einem Frameknoten und einem Slotknoten anzeigt, daß der Slot zu diesem Frame gehört. Die bedeutungstragenden Teile der Graphik sind also die Knoten des Baumes, eine Konsequenz aus der Systemphilosophie, die zur Modellierung von relationalen Beziehungen zwischen Konzepten erfordert, daß die betroffenen Konzepte spezifische Strukturen aufweisen. Das Einfügen einer Relationskante geschieht i.a. implizit durch die Manipulation der Slots eines der betroffenen Frames, die darauffolgende graphische Darstellung der Kante dient einer bestätigenden Rückmeldung an den Benutzer.

b) Navigation in der Wissensbasis

- Innerhalb eines der ausgewählten Wissensfragmente, die nicht immer vollständig auf dem Bildschirm gezeigt werden können, navigiert der Benutzer mit den Mitteln, die ihm das XS-2 bietet. So kann ein Objekt mit der Maus angewählt, der Baum erforscht und verschoben werden (vgl. /9/).
- Zusätzlich zu dieser 'horizontalen' Navigation erlauben die Expansionskommandos eine Bewegung in 'vertikaler' Richtung: Der Wechsel vom Frame- zum Slotmenü läßt sich als 'Zoom'-Operation interpretieren, d.h. als Navigation orthogonal zur Darstellungsebene.

c) Manipulation der Konzepte

- Gemäß dem Prinzip 'Ähnliche Operationen sollten auf ähnliche Weise auszuführen sein' sind die Kommandos auf allen Abstraktionsebenen analog:
 - Das 'Edit'-Kommando dient zum Einstieg in das Menü.
 - 'Get'- bzw. 'Show'-Kommandos dienen der Orientierung.
 - 'Create'- und 'Delete'-Befehle ermöglichen eine Änderung der Wissensbasis auf dem jeweils gewählten Niveau.
- Spezifische, den Situationen angepaßte Rückkopplungskommandos ('Show Consequences') bieten dem Benutzer die Möglichkeit, die globalen Auswirkungen seiner lokalen Änderungen zu verfolgen, indem vom System automatisch durchgeführte Folgeoperationen tabellarisch aufgeführt werden. Wird z.B. ein Oberbegriff manipuliert, so kann dies Auswirkungen auf alle Unterbegriffe haben.

3. Konsequenzen für das Gesamtsystem

Der Framenetz-Editor stellt ein Software Instrument dar, das dem Systementwickler die Möglichkeit gibt, die dem Textanalyseprozeß zugrunde liegenden Wissensstrukturen den jeweiligen Erfordernissen anzupassen (Modellierung spezifischer Diskursbereiche etc.).

Auch für den Endbenutzer, der die Resultate von Textanalysen betrachten will, eröffnet die Präsentation relevanter Informationen in kaskadierten Abstraktionsstufen (relationale Beziehungen zwischen Frames, Slotmenge eines Frames, Menge der Einträge eines Slots) neue Zugriffspfade zur Textinformation, die nicht an die lineare Textform gebunden sind (vgl. /10/):

- Bewegung entlang den Stufen einer Abstraktionshierarchie, die Textkondensate unterschiedlicher Ausführlichkeit enthält.

- Navigation innerhalb der Wissensstrukturen eines gewählten Abstraktionsniveaus.

Ein Interface mit diesen Möglichkeiten wird zur Zeit entwickelt.

Literaturangaben

- / 1/ Cahn, David P.: Computer-Aided Visualization of Database Structural Relationships. In: Proc. of the Annual Meeting of the ASIS, 17, 1980, pp. 358-360.
- / 2/ Hahn, Udo; Hammwöhner, Rainer; Kuhlen, Rainer; Reimer, Ulrich; Thiel, Ulrich: TOPIC II / TOPOGRAPHIC II: Automatische Textkondensierung und text-orientiertes Informationsmanagement, Universität Konstanz, Informationswissenschaft, Dezember 1984, (Bericht TOPIC-12 & TOPOGRAPHIC-3 / 1984).
- / 3/ Hahn, Udo; Reimer, Ulrich: Wortexperten-Parsing: Text-Parsing mit einer verteilten lexikalischen Grammatik im Rahmen des automatischen Textkondensierungssystems 'TOPIC'. In: Linguistische Berichte, 1983, No.88 (Dec), pp. 56-78.
- / 4/ Hayes, Phil: Expanding the Horizons of Natural Language Interfaces. In: 18th Annual Meeting of the ACL, 1980, pp. 71-74.
- / 5/ Frei, H.P.; Jauslin, J.-P.: Graphical Presentation of Information and Services: A User Oriented interface. In: Information Technology: Research and Development, Vol. 2, 1983, pp. 23-42.
- / 6/ Minsky, Marvin: A Framework for Representing Knowledge. In: P.H. Winston (ed.): The Psychology of Computer Vision, New York: McGraw-Hill, 1975, pp. 211-277
- / 7/ Nievergelt, Jürg; Weydert, Jean: Sites, Modes and Trails: telling the User of an Interactive System where he is, what he can do, and how to get to Places. In: Guedj et al. (eds.): Methodology of Interaction, IPIP, North-Holland, 1980, pp. 327-338.
- / 8/ Reimer, Ulrich; Hahn, Udo: On Formal Semantic Properties of a Frame Data Model. In: Computers and Artificial Intelligence, Vol. 4, No. 5, 1985.
- / 9/ Stelovsky, Jan: XS-2: The user interface of an interactive operating System, Dissertation ETH, 1983.
- /10/ Weyer, Stephen A.:The Design of a Dynamic Book for Information Search. In: Int. J. Man - Machine Studies, 17, 1982, pp. 87 – 107.
- /11/ Zdybel, Prank; Greenfeld, Norton R.; Yonke, Martin D.; Gibbons, Jeff: An Information Presentation System, in: Proc. of the 7th Int. Joint Conf. on AI, Vancouver 1981, pp. 978 – 984.
- /12/ Zhang, Quian-Zing; Mendelzon, Alberto O.: A graphical query Language for Entity-Relationship. In: C.G. Davis et al.(eds.): Entity-Relationship Approach to Software Engineering, Amsterdam: North Holland, 1983, pp. 441-448