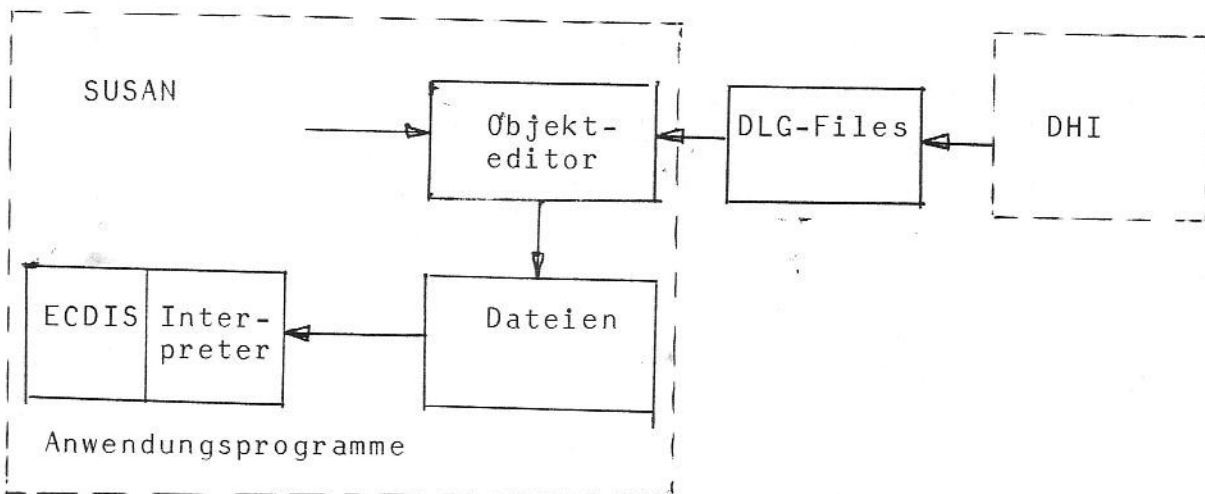


Rodenbach, den 28. 1. 90

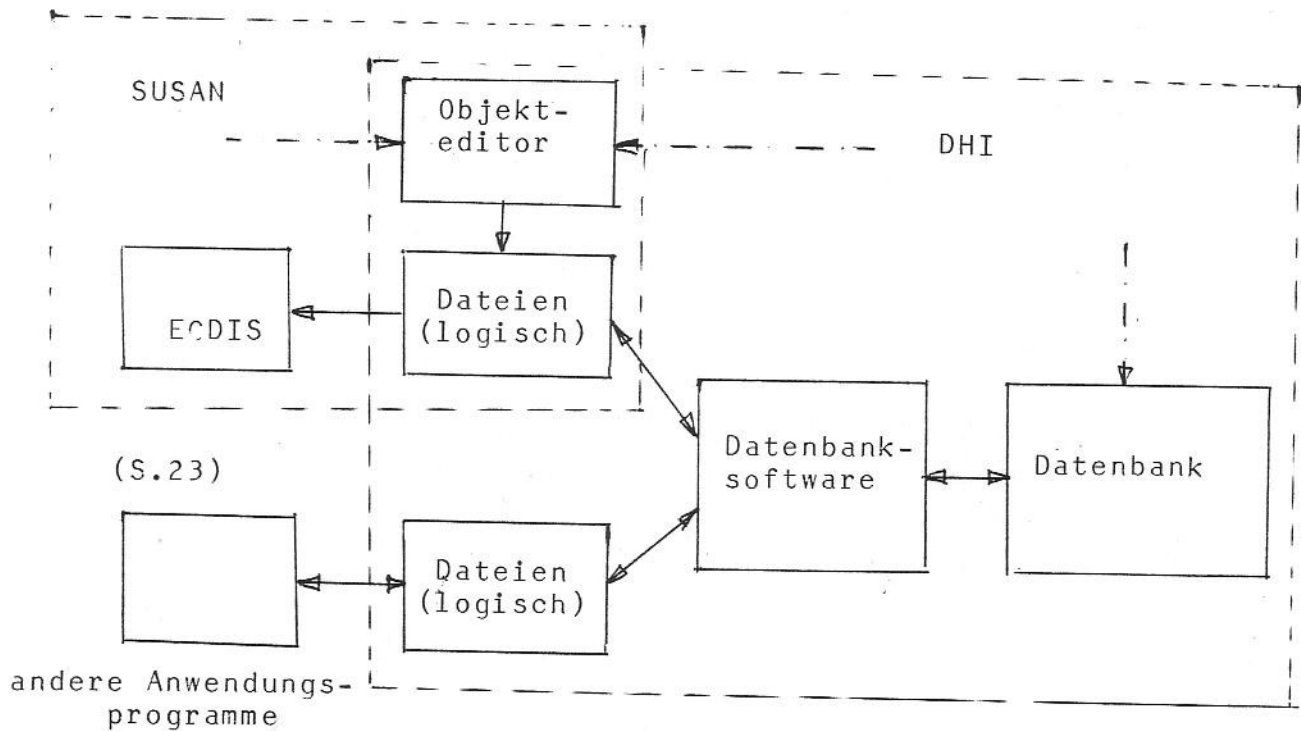
Hallo Hamburger, Norddeutsche, Rheinländer und Exil-Schwäbinnen!
Ich will Euch hier meine Überlegungen im stillen Kämmerlein mit-
teilen, die sich am Buch von Schlageter, Stucky: "Datenbanksysteme:
Konzepte und Modelle" (Teubner, Stuttgart 1983) entlanghangeln
(Seitenzahlen beziehen sich auf dieses Buch) und vielleicht zur
Klärung des Verhältnisses SUSAN- DHI beitragen, was die Erstellung
der Datenbank angeht, sowie das weitere Vorgehen bzgl. Objekt-
editor und das Gespräch mit Mellis am 7. 2. betreffen.

Eigentlich ist das alles für das laufende Mufdi-Projekt nur sehr
peripher, (wir wollen ja nur was zum Sehen haben) aber für spätere
Projekte SUSAN-DHI in Sachen elektronische Seekarte ziemlich
grundlegend. An SUSAN bauen wir derzeit einen Objekteditor, der
es uns erlaubt, auf der Grundlage offizieller DHI-DLG-Files Ob-
jekte (samt Attributen und Relationen) gemäß Objektkatalog zu
basteln und das alles für eine spätere Darstellung in ECDIS:



Der Objekteditor soll dabei Objektbeschreibungen in Dateien
liefern, die wir mittels eines Interpreters als Seekarte darstellen
können. Wir werden diese Dateien natürlich so bauen, daß sie für
unsere Anwendung möglichst effizient sind (vgl. S.19). Da jeder
Anwender ~~Anwender~~ seine eigenen, auf ihn zugeschnittenen Dateien
mit derselben Information wünschen wird, sieht das Konzept eines
Datenbanksystems vor, diesen Zustand beizubehalten, jedem Anwender
also die Daten in der ihm gewohnten Form zu liefern (logische
Dateien), sie in Wirklichkeit aber im Hintergrund in einer Daten-

bank zentral zu verwalten (S.23). Damit werden die Probleme Redundanz, Inkonsistenz, Daten-Programm-Abhängigkeit, Inflexibilität vermieden:



Ein gutes Datenbanksystem soll nicht nur dem Anwender aus der Datenbank Information in der von ihm gewohnten Form liefern, sondern es ihm auch erlauben, in seinen Dateien Änderungen vorzunehmen, die von dem Datenbanksystem in die Datenbank transformiert werden.

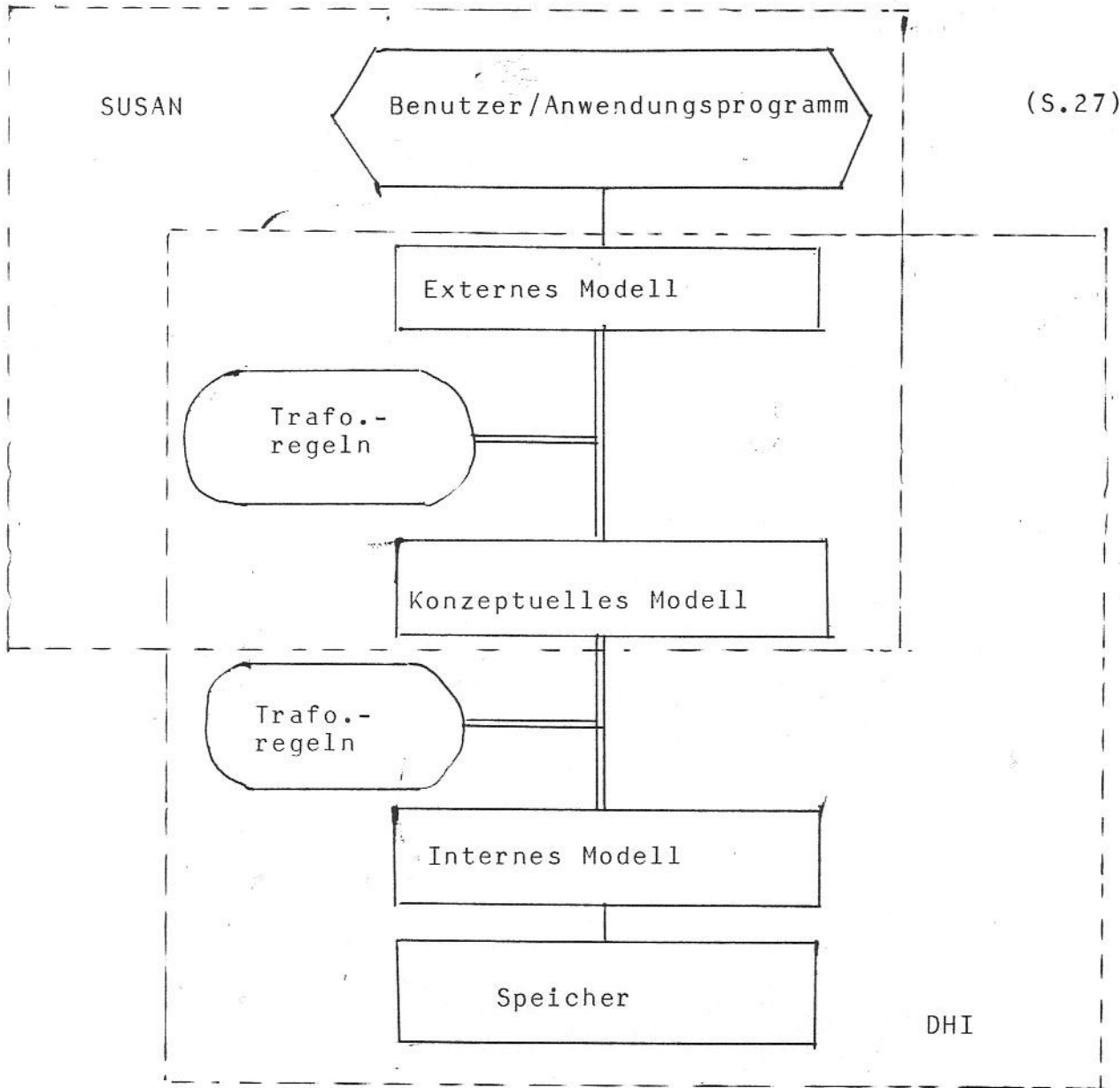
Noch ein Wort zur Redundanz: "Möglicherweise wird aus Effizienzgründen in gewissem Umfang Redundanz auch in der Datenbank beibehalten; da aber jetzt alle Daten zentral verwaltet werden, kann für eine quasi-simultane Veränderung von Duplikaten derselben Information gesorgt werden, d.h. für jedes beliebige Anwendungsprogramm enthält die Datenbank zu keinem Zeitpunkt widersprüchliche Angaben über dasselbe Objekt." (S.25) Aus obigem Schema ist zu sehen,

wo sich die Interessen von SUSAN und DHI direkt überschneiden und gemeinsames Vorgehen für beide unmittelbar von Nutzen ist.

Prinzipiell sieht die Architektur eines Datenbanksystems die Darstellung der Daten als Beschreibung der realen Welt aus der Sicht dreier verschiedener Perspektiven vor:

- logische Gesamtsicht (konzeptuelle Ebene)
- physische Datenorganisation (interne Ebene)
- die Sicht einzelner Benutzer (externe Ebene) (S.26)

Jeder Ebene entspricht ein eigenes Datenmodell:



"Das konzeptuelle Modell beschreibt die Daten...auf logischer Ebene, völlig unabhängig von Gesichtspunkten der Datenverarbeitung. Es gibt das Modell der Daten wieder, das durch die Analyse der realen Welt über Abstraktion und Klassenbildung erhalten wird. Es wird deshalb nicht von EDV-Fachleuten erstellt."(S.28) "Die Instanz, die das konzeptuelle Modell entwickelt, wird Unternehmensadministrator genannt."(S.29) "Die Beschreibung des externen Modells wird vom Anwendungsadministrator erstellt. Es kann mehrere Anwendungsadministratoren geben, etwa für jede Anwendung einen. Aufgrund des geschilderten Datenbankaufbaus wird sich ihre Tätigkeit gegenseitig nicht beeinflussen, vorausgesetzt, daß alle von ihnen gestellten Anforderungen berücksichtigt wurden." (S.32f). "Wie die jetzt logisch beschriebenen Daten im Speicher abgelegt werden und welche Zugriffsmöglichkeiten auf diese Daten bestehen sollen" ist Sache des Datenbankadministrators. (S.33)

SUSAN kann also die Rolle eines Anwendungsadministrators übernehmen, während das DHI die Rollen von Unternehmens- und Datenbankadministrator spielt. Kann SUSAN ein schlüssiges externes Modell der Daten liefern, so ist dies sicher von sehr großem Nutzen für den Aufbau der Datenbank am DHI!

Stillschweigend scheint man sich für das relationale Datenmodell gegenüber anderen möglichen Modellen (Netzwerkmodell, hierarchisches Modell) entschieden zu haben, da es das am besten theoretisch fundierte Modell ist und viele Vorzüge aufweist.

Mathematisch ist eine Relation einfach eine Teilmenge des kartesischen Produkts einer Anzahl beliebiger Mengen:

$$R \subseteq M_1 \times M_2 \times \dots \times M_n$$

Das kartesische Produkt der Mengen M_1, M_2, \dots, M_n ist die Menge aller n -Tupel (m_1, m_2, \dots, m_n) mit $m_i \in M_i$ ($i = 1, 2, \dots, n$).

R ist dann eine n -stellige Relation.

Beispiel: die 3-stellige Relation LIEGT_ZWISCHEN läßt sich als Teilmenge des kartesischen Produkts STÄDTE \times STÄDTE \times STÄDTE der Menge aller Städte bestimmen. So ist das Tripel (= 3-Tupel)

(Kaiserslautern, Stuttgart, Paris) oder (Hanover, Stuttgart, Hamburg) sicher Element dieser Teilmenge und damit der Relation, nicht aber

(New York, Hamburg, Stade). Andererseits: eine Objektklasse ist eindeutig beschrieben durch eine Attributklassenkombination, ein Objekt dieser Klasse durch eine entsprechende Attributwertkombination. "Dies bedeutet gerade, daß man in natürlicher Weise die Objekte einer bestimmten Objektklasse als Relation über den Wertebereich der entsprechenden Attributklassen auffassen kann, sofern man diese in irgendeiner Weise anordnet. Einer Objektklasse entspricht eine Relationenklasse." (S.81) Man hat es also lediglich

mit einem Konzept zu tun, was das Hantieren mit den Daten sehr erleichtert. Wichtig ist, daß die Elemente der Mengen des kartesischen Produkts auch wirklich elementar sind. Daraus folgt, daß auch die Attribute von Objekten elementar sein müssen, eine Forderung, die der Objektkatalog noch nicht durchgängig erfüllt.

"Eine Relation, bei der jedes Attribut elementar ist, ist in der ersten Normalform (oder 1NF-Relation)" (S.83) Diese und weitere

Normalformen (2. Normalform (2NF), 3. Normalform (3NF), Boyce-Codd-Normalform (BCNF)) sind zur Vermeidung sogenannter Anomalien

wichtig, die bei der inadäquaten Konstruktion von Relationen auftreten können (Herr Mellis wird uns darüber und über Verfahren zur Normalisierung von Relationen sicher einiges sagen können):

Angenommen man will eine relationale Datenbank über Mitarbeiter an einem Projekt anlegen und ordnet Projekt-Nr., Mitarbeiter-Nr. und Telefon-Nr. zu einer Relation zwischen Mitarbeiter und Projekt, etwa, weil man seine Leute im Zusammenhang mit dem Projekt anrufen will. Aus Gründen der Redundanzfreiheit soll die Telefon-Nr. nirgends sonst gespeichert sein. (S.163)

"Soll ein neuer Mitarbeiter in die Datenbank aufgenommen werden, der sich zunächst einarbeiten muß und daher noch an keinem Projekt mitarbeitet, so kann die Telefonnummer des Mitarbeiters nicht in der Datenbank abgespeichert werden - es liegt eine sog. insertion anomaly vor. Wird andererseits ein bestimmtes Projekt abgeschlossen, so werden auch die betreffenden Elemente der Relation entfernt. Für Mitarbeiter, die nur an diesem einen Projekt arbeiteten verschwindet somit auch die Information über ihre Telefonnummer - es liegt eine sog. deletion anomaly vor. Ändert sich die Telefonnummer eines Mitarbeiters, so ist die gesamte Relation zu durchsuchen und alle Einträge in verschiedenen Projekten sind abzuändern; obwohl nur ein einziger Tatbestand verändert wird, müssen in der Datenbank mehrere Änderungen vorgenommen werden; in diesem Fall spricht man von einer update anomaly."(S.164) Die Normalformen helfen diese Anomalien zu vermeiden.

"Eine 1NF-Relation ist in zweiter Normalform (oder 2NF-Relation), wenn jedes Nichtschlüsselattribut von jedem Schlüssel voll funktional abhängig ist."(S.185)

Eine Attributwertekombination ist identifizierend für ein Objekt. Dies kann aber auch bereits für eine Teilmenge der Attributwerte gelten. Ist eine solche Teilmenge minimal in dem Sinne, daß man kein Attribut daraus weglassen kann, ohne die Eigenschaft "identifizierend" zu verlieren, so ist sie ein Schlüssel für das Objekt. (S.46)

"Beschreibt eine Relation lediglich ein Konzept, d.h. sind keine transitiven Abhängigkeiten von einem Schlüssel vorhanden, so ist die Relation in dritter Normalform"(oder 3NF-Relation)(S.186)

Für derart normalisierte Relationen steht ein ganzes Instrumentarium an Operationen zur Verfügung um neue Relationen zu erzeugen. Da man in Mengen denken muß kommen auch Mengenoperationen zum Einsatz (S.87ff). "Das Konzept der Normalformen...ist von großer praktischer Bedeutung sowohl für den Unternehmensadministrator wie für den Datenbankadministrator und auch den Benutzer... Dem Unternehmensadministrator sind die Normalformen (insbesondere die 3. Normalformen 3NF und BCNF) eine große Hilfe, da sie ihn zur klaren Trennung der einzelnen Konzepte in seinem Modell zwingen... Das Problem des Data Design, d.h. der Aufstellung von Relationen, ist ein größeres Entwurfsproblem. Die vorhandenen funktionalen Abhängigkeiten müssen vom Unternehmensadministrator voll erkannt und erfaßt werden, eine automatische Unterstützung ist höchstens in Form von Plausibilitätsprüfungen sowie Entdecken von Widersprüchen möglich. Der Unternehmensadministrator muß daher in der Lage sein, die Semantik der Daten zu verstehen. - Hinzu kommt, daß die Normalformen nicht eindeutig sind, sondern daß aus mehreren Alternativen eine möglichst optimale auszuwählen ist. Kriterien für Optimalität sollten allerdings nicht einfach 'Anzahlen der Relationen' sein, sondern die Optimalität muß sich auf die Effizienz des Gesamtsystems unter Berücksichtigung des Benutzerverhaltens beziehen."(S.197f)

Des langen Referates kurzer Sinn: zum einen wollte ich Euch damit die wichtigsten Grundbegriffe aufzählen, die im Zusammenhang mit Relationen auftauchen, zum anderen glaube ich, daß das, was wir SUSAN in der Rolle des "Benutzers" der (zukünftigen) DHI-Datenbank

tun können nicht nur nicht überflüssig, sondern sogar unbedingt sinnvoll und notwendig für eine gute Strukturierung derselben ist. Wenn es richtig ist, wie ich oben die Unterschiede und Gemeinsamkeiten in den Aufgaben von DHI und SUSAN skizziert habe, so kann dieses vielleicht als Argumentationshilfe für die sachliche Begründung geförderter gemeinsamer Projekte dienen, bzw. als Begründung des DHI für einen Projektauftrag an SUSAN.

Ich meine, daß wir weiter an unserem Objekteditor und Objektinterpreter arbeiten können und je mehr gute Ideen wir dabei haben, desto besser! Sollte mir in der Zwischenzeit was Brauchbares einfallen, lasse ich es Euch wissen. Ich werde kommende Woche mal anrufen um mich über den Stand der Dinge zu informieren. Ansonsten grüße ich die gesamte Crew und wünsche insbesondere den Mufdis samt kartografischer Abteilung gute Erfolge und Fortschritte.

Viele Grüße, bis demnächst

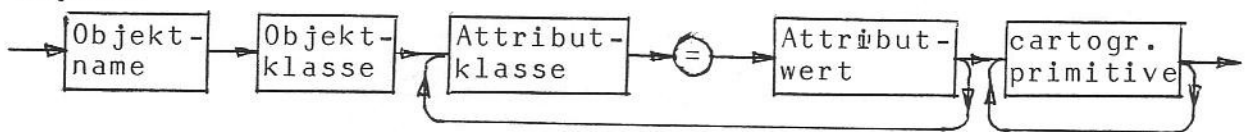
A handwritten signature, possibly 'S.G.', written in dark ink.

Rodenbach, den 30.1.90

Hallo Objektiviererinnen und Objektivierer!

Mir ist was zu den Funktionen von Objekteditor und Objektinterpreter eingefallen, mit dem ich Euch nicht verschonen möchte: Wenn wir in der Lage sind, aus den mit den DLG-Files gelieferten geometric primitives neue Flächen und Linien im Sinne von cartographic primitives zusammzusetzen (Heino hat Skizzen zu den entsprechenden Algorithmen), so könnten wir den Kern des Objekteditors schreiben, der eine Objektbeschreibung in folgender Form (Syntax) liefert:

Objekt:



Der Objektinterpreter könnte eine Objektbeschreibung nach dieser Syntax lesen und die Attributklassenwerte der konkreten Objektklasse z.B. als Farbe, Textur der Flächen oder als Signatur der Linien bei der Darstellung der konkreten cartographic primitives interpretieren und damit das konkrete Objekt darstellen. Damit wäre der vollständige Inhalt der Papierseekarte auf dem Bildschirm übertragbar, aber noch nicht allzuviel mehr. Gewiß würde der Objektansatz z.B. eine dem Bildausschnitt angepaßte Beschriftung erlauben, etwas, was bereits deutlich über die Möglichkeiten einer gescannten Papierseekarte hinausginge. Z.B. könnte die Schrift unabhängig vom Ausschnitt ("Maßstab") lediglich nach Lesbarkeit in ihrer Größe und Form ausgewählt sein.

Eine Bemerkung zu den Attributklassen: Manche Werte sind als variable Eingaben vorgesehen. Das ist dann nicht weiter schlimm, wenn diese Eingaben als Einheiten, Atome betrachtet werden, die nicht weiter interpretiert werden müssen, sondern mit denen als ganzes Operationen vorgenommen werden können, etwa Einträge, die Herkunft, Quelle und Urheber des Objekts kennzeichnen und ungeteilt für urheberrechtliche Zwecke weiterverarbeitet werden, z.B. ausgegeben werden können.

Richtig interessant wird der Objektansatz aber erst dann, wenn abhängig vom Maßstab, oder besser vom Ausschnitt (nach geographischer Längen- und Breitenausdehnung (=sm!), da sonst bei gleichem Ausschnitt, Zeilenraster und Bildinhalt die Maßstäbe auf einem 12"-Monitor andere wären als auf einem 15"-Monitor) Objekte ein- oder ausgeblendet werden sollen, bzw. wenn z.B. das Thema der

Karte gewechselt werden soll.

Ihr erinnert Euch an den Streit, ob für diesen Fall das Layer-Modell oder das topologische Modell vorzuziehen sei. In beiden Fällen kommt man nicht ohne Relationen im Sinne von Beziehungen zwischen Objekten aus und ich behaupte nun, daß es in beiden Fällen dieselben Relationen sind, die nur je nach Modell unterschiedlich für die Darstellung interpretiert werden müssen. Das will ich im folgenden exemplarisch untermauern. Damit wird nämlich die Objektbeschreibung wieder darstellungsunabhängig.

Zunächst eine Auflistung der mathematischen Eigenschaften, die Relationen haben können:

- | | |
|----------------------------------|--|
| 1) R reflexiv | $Ax: (xRx)$ |
| 2) R symmetrisch | $AxAy: (xRy \Rightarrow yRx)$ |
| 3) R asymmetrisch | $AxAy: (xRy \Rightarrow \neg(yRx))$ |
| 4) R identitiv (antisymmetrisch) | $AxAy: (xRy \ \& \ yRx \Rightarrow x=y)$ |
| 5) R konnex (linear) | $AxAy: (xRy \vee yRx)$ |
| 6) R transitiv | $AxAyAz: (xRy \ \& \ yRz \Rightarrow xRz)$ |

(dtv-Atlas Mathe S.31, Bdl)

Je nachdem, welche Eigenschaften eine konkrete Relation hat, kann man diese bestimmten Relationsarten zuordnen:

Eine Teilmenge \subseteq von $M \times M$ heißt Ordnungsrelation, wenn sie reflexiv, identitiv und transitiv ist. $(M; \subseteq)$ heißt geordnete Menge.

Eine Teilmenge $=$ von $M \times M$ heißt strenge Ordnungsrelation, wenn sie asymmetrisch und transitiv ist. $(M; =)$ heißt streng geordnete Menge. (dtv-Atlas Mathe S.43)

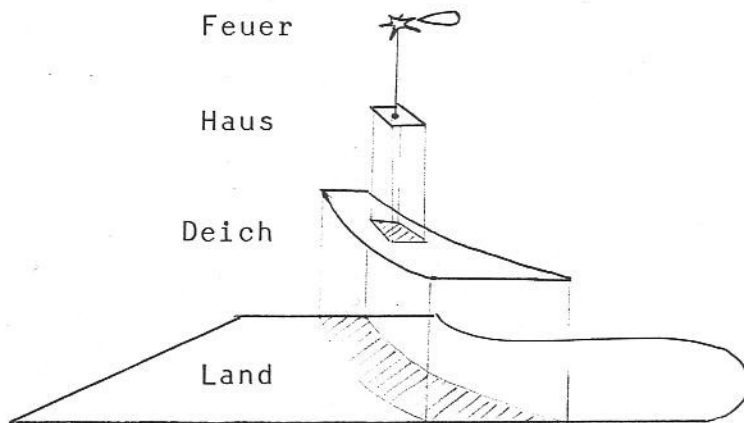
Zu jeder Relation $R \subseteq M_1 \times M_2$ existiert eine Umkehrrelation (inverse Relation) $R^{-1} \subseteq M_2 \times M_1$ definiert durch $R^{-1} := \{(x_2, x_1) \mid (x_1, x_2) \in R\}$, d.h. $x_2 R^{-1} x_1 \Leftrightarrow x_1 R x_2$.

Eine Äquivalenzrelation ist eine Relation, die reflexiv, symmetrisch und transitiv ist. Eine Menge M wird durch eine Äquivalenzrelation in paarweise disjunkte nichtleere Teilmengen zerlegt (Klassenbildung). (dtv-Atlas Mathe S.31)

Nun aber zu einem modellhaften Ausschnitt aus der Realität und zu meinem Lösungsvorschlag, wie die Forderung

"Bei kleinem Maßstab (=großem Ausschnitt) stelle nur die navigatorisch wichtigen Objekte dar!"

erfüllt werden kann:



Dies seien konkrete unverwechselbare Objekte. Ich definiere nun die strenge Ordnungsrelation LIEGT_ÜBER, welche die Menge aller Objekte {Feuer, Haus, Deich, Land} zu einer streng geordneten macht:

Feuer LIEGT_ÜBER Haus,

Haus LIEGT_ÜBER Deich,

Deich LIEGT_ÜBER Land sind Elemente dieser Relation. Statt

Haus LIEGT_ÜBER Deich schreibe kurz LIEGT_ÜBER(Haus, Deich) oder

noch kürzer (Haus, Deich) als geordnetes Paar dieser Relation, die das kartesische Produkt der Menge aller Objekte mit sich selbst

ist, genauer, eine echte Teilmenge davon. Als Elemente gehören zu dieser Ordnungsrelation also

(Feuer, Haus),

(Haus, Deich),

(Deich, Land)

und weiter wegen der Transitivität der Ordnungsrelation (vgl. Eigenschaft 6)) deduzierbar:

(Haus, Deich) & (Deich, Land) \rightarrow (Haus, Land), ebenso

(Feuer, Deich), (Feuer, Land)

Mit diesen sechs Paaren ist die Relation LIEGT_ÜBER für unsere Modellobjektmenge vollständig bestimmt. Natürlich gibt es zu dieser Relation auch eine Umkehrrelation, die logischerweise LIEGT_UNTER heißt: wenn gilt 'Haus LIEGT_ÜBER Deich', dann gilt gemäß der formalen Definition 'Deich LIEGT_UNTER Haus' und das ist sogar richtig. LIEGT_UNTER ist selbst wieder eine strenge Ordnungsrelation und der Beweis oder die Widerlegung des Satzes, daß jede Relation vom selben Typ ist wie ihre Umkehrrelation wäre reine, abstrakte Mathematik.

Ich definiere weiter die einstellige Relation WICHTIG entsprechend der Bedeutung der Objekte für die Navigation (den speziellen Gebrauch/Zweck):

WICHTIG(Feuer)

WICHTIG(Land)

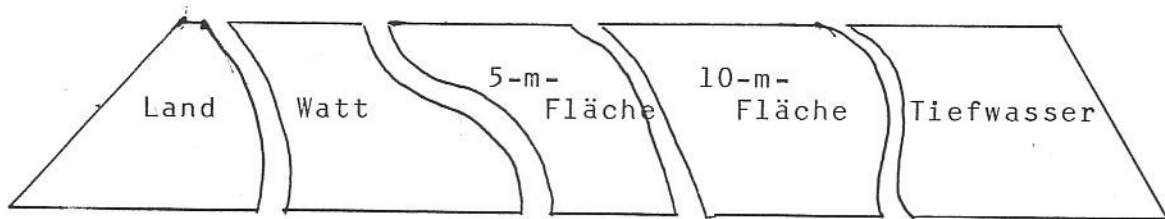
Entsprechend gibt es eine Relation UNWICHTIG, die zusammen mit der Relation WICHTIG die Menge aller Objekte in zwei disjunkte Teilmengen gliedert:

UNWICHTIG(Haus)

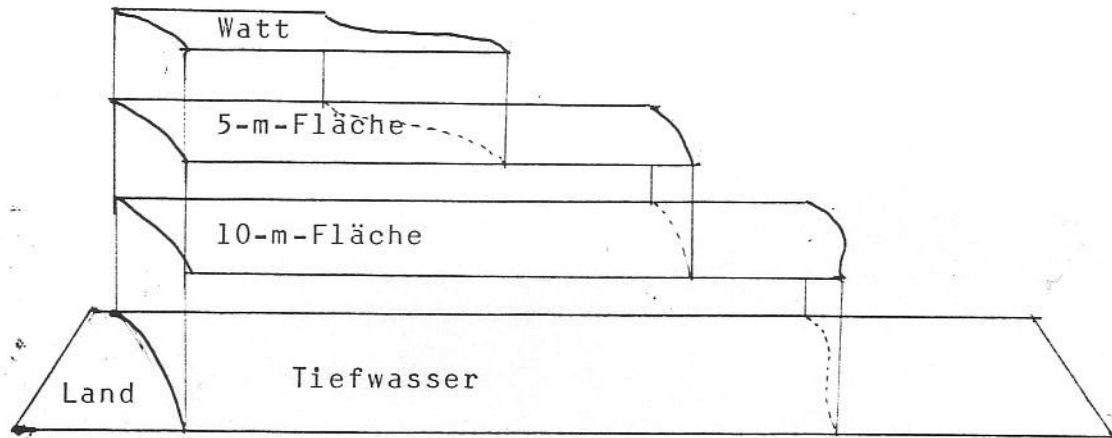
UNWICHTIG(Deich)

Formal ist das noch nicht ganz sauber, da WICHTIG und UNWICHTIG wohl eher zwei Klassen einer genauer zu bestimmenden zweistelligen Äquivalenzrelation sind, als selber Relationen. Das ist im Moment aber nicht weiter wichtig. Die Ausblendungsforderung bedeutet nun also: Stelle bei kleinem Maßstab nur alle WICHTIGen Objekte dar! Diese Forderung bedeutet im Layer-Modell, daß nur die Objekte der Relation WICHTIG dargestellt werden und zwar in der Reihenfolge, die durch die Ordnungsrelation LIEGT_ÜBER (bzw. LIEGT_UNTER) festgelegt ist. Im Topologie-Modell ist die Reihenfolge der Darstellung der Objekte egal, da sie sich nicht überlappen. Beim Weglassen der UNWICHTIGen Objekte entstehen allerdings Löcher in der Karte. Diese müssen gefüllt werden. Dafür muß wieder die LIEGT_ÜBER-Relation erhalten (in Verbindung mit der WICHTIG-Relation): Soll im topologischen Modell ein flächiges Objekt ausgeblendet werden, so dürfen als erstes die zur Fläche gehörenden Linien nicht mehr gezeichnet werden. Als nächstes muß die Fläche so eingefärbt werden, daß man sie nicht mehr sieht: Die Farbe und Struktur der auszublendenden Fläche bestimmt sich nach Farbe und Struktur des ersten WICHTIGen Objekts, auf das man beim transitiven Abstieg durch die LIEGT ÜBER-Relation stößt! Beispiel: Welche Farbe muß das Haus haben, wenn es als UNWICHTIGes Objekt ausgeblendet werden soll? In der LIEGT_ÜBER-Relation finden wir, sofern diese vollständig aufgestellt ist, die beiden Elemente: (Haus, Deich) und (Haus, Land). Deich ist wie Haus selbst ein UNWICHTIGes Objekt, Land dagegen ein WICHTIGes, also wird die Haus-Fläche wie die Landfläche eingefärbt (und schon ist das Haus weg!). Beim Tektronik-Terminal muß der Interpreter beim Aufbau der Objekte diese Prüfungen vorweg durchführen und für die betreffenden UNWICHTIGen

Objekte sogleich die Darstellung (Segmente) mit aufbauen, in der sie erscheinen sollen, wenn sie nicht erscheinen sollen!
Noch ein Beispiel zum Test: Im kleinen Maßstab sollen Wattflächen und Tiefenflächen zu einer einzigen Flachwasserfläche verschmolzen werden (könnte ich mir jedenfalls so vorstellen!):
Topologisches Modell:



Layer-Modell:



Während im topologischen Ansatz das Umfärben mittels einer Relation schwierig wird (wie soll man diese definieren? FLACHER_ALS_RECHTS? Das kann anderswo genau umgekehrt sein!), macht es in der Layer-Anschauung keine Probleme:

WICHTIG: Land, 10-m-Fläche, Tiefwasser

UNWICHTIG: Watt, 5-m-Fläche

LIEGT_ÜBER: (Watt, 5-m-Fläche)

(5-m-Fläche, 10-m-Fläche) → (Watt, 10-m-Fläche)

(10-m-Fläche, Tiefwasser) → (Watt, Tiefwasser)

→ (5-m-Fläche, Tiefwasser)

Trotzdem bedeutet das nicht, daß man nun das Layer-Modell realisieren muß (wir waren uns ja darin einig, daß dies schwerwiegende technische Nachteile mit sich bringen würde). Man kann beim Interpretieren ja entsprechend dem geschilderten Verfahren die Flächen in den für die jeweilige Darstellung nötigen Farben erzeugen und als ~~als~~ Segmente für den evtl. Gebrauch vorhalten. Das Layer-Modell liefert hier ein eindeutiges Verfahren für die Einfärbung von

Flächen, auch wenn sich diese garnicht überlappen. Dieses ergäbe z.B. für das Watt die Fläche einmal in der Wattfarbe (wenn es dargestellt wird) und einmal in der 10-m-Flächenfarbe, wenn es nicht dargestellt werden soll.

Der Objekteditor muß also neben der Menge der Objektbeschreibungen (einer Zelle) auch die Menge der Relationen über diese Objekte liefern, die ihrerseits Teilmengen von kartesischen Produkten der Menge der Objekte mit sich selbst sind. Es muß auch klar werden, was für eine Art von Relation es jeweils ist (zu welcher Relationsklasse sie gehört, ob es eine Ordnungs- Äquivalenzrelation oder sonstwas ist), da dies unterschiedliche Anwendungen im Objektinterpretier bedeutet.

Soweit meine Überlegungen dazu. Alles was mir sonst noch einfällt werde ich nächste Woche wieder höchstpersönlich vortragen. Ich hoffe, daß ich einige Anregungen liefern konnte.

Bis dann, Gruß



Zu meinem Wunsch, auf Forschungsschiffen zu fahren.

Der Wissenschaft und Technik galten von jeher meine Interessen. Allerdings erkannte ich während meines Studiums der Mathematik und Physik, daß meine Fähigkeiten weniger auf akademisch - theoretischem Gebiet liegen, als vielmehr praktischer Natur sind. In der Seefahrt habe ich meinen idealen Beruf gefunden. Jedoch habe ich das Interesse an der Wissenschaft nicht verloren und mich in die Programmierung eingearbeitet, so daß ich während des Studiums am Fachbereich Seefahrt der Fachhochschule Hamburg am Schiffsführungssimulator tätig sein konnte und an Simulationserweiterungen beteiligt war. Wenn auch das Programmieren nicht zu den Aufgaben eines Nautikers zählt, kann doch in dieser Beziehung große Aufgeschlossenheit von mir erwartet werden.

Als Matrose konnte ich während meiner Fahrzeit 1981 auf dem Bergungsschlepper "Baltic" Erfahrungen beim Vermessen von Ölfeldern sammeln. Im Auftrag der "Burmah Oil Exploration Limited" wurde der Meeresboden des zu erschließenden Ölfeldes "Dentale" vor der Küste Gabuns mittels geschleppter Sonden vermessen. Aufgrund meines Interesses wurde ich vom Kapitän des Schleppers, Herrn Rohleder, als Steuerer eingesetzt. Meine Aufgabe bestand darin, das Schiff mit Hilfe eines eigens installierten mobilen Hyperbelnavigationssystems metergenau auf geraden Bahnen über das Ölfeld zu steuern, am Ende der Bahn in einer Art Williamson-Turn auf Gegenkurs in 100m Parallelabstand zu gehen, ohne daß im Drehkreis der Fahrtverlust so groß wurde, daß die Schleppsonde auf den Meeresboden absank.

Als Nautiker möchte ich neben Routinetätigkeiten auch mit interessanten und schwierigen Aufgaben gefordert werden. Da ich glaube, solche Aufgaben in der Forschungsschiffahrt zu finden und aus meinem grundsätzlichen Interesse an Wissenschaft, Forschung und Technik möchte ich als Steuerer auf Forschungsschiffen fahren.

Hamburg, den 20.10.1986

A handwritten signature in black ink, consisting of a cursive name that appears to be 'Wolfgang' followed by a surname, possibly 'Schmidt', written in a fluid, connected style.