

Automatische Generierung von Oberflächen zur interaktiven Lösung von Zuordnungsproblemen

Diplomarbeit im Fach Informatik
vorgelegt von

cand. inform.

Frank Forster

Betreuer

Prof. Dr. Frank Puppe

Dipl.-Inform. Ciske Busch

Lehrstuhl für Künstliche Intelligenz
und Angewandte Informatik
Universität Würzburg

Ich erkläre hiermit, die vorliegende Diplomarbeit selbständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben.

Würzburg, den 31.10.1999

(Frank Forster)

Inhalt

1	Einleitung	5
1.1	Motivation	5
1.2	Ziel und Aufbau der Arbeit	6
2	Grundbegriffe	9
2.1	Definition von Zuordnungsproblemen.....	9
2.2	Klassen, Objekte und Attribute der Zuordnung	11
2.3	Randbedingungen.....	12
2.3.1	Definition.....	12
2.3.2	Klassifizierung von Randbedingungen	13
2.3.2.1	Einfache Randbedingungen.....	13
2.3.2.2	Komplexe Randbedingungen.....	15
2.3.2.3	Zusammenfassung	17
2.3.3	Harte und weiche Randbedingungen	18
2.4	Fazit	19
3	Anforderungen an eine Oberfläche zur interaktiven Lösung von Zuordnungsproblemen.....	21
3.1	Allgemeine Anforderungen	21
3.1.1	Allgemeine Anforderungen an Software.....	21
3.1.2	Allgemeine Anforderungen an ein Benutzerinterface.....	22
3.1.3	Fazit.....	24
3.2	Anforderungen an die Darstellung des Zuordnungsproblems	25
3.2.1	Darstellung der Zuordnung.....	25
3.2.1.1	Grundtypen der Zuordnungsdarstellung.....	26
3.2.1.2	Kodierungsformen des Markierungsobjekts	31
3.2.1.3	Fazit	36
3.2.2	Visualisierung verletzter Randbedingungen	38
3.2.2.1	Relevante Informationen	38
3.2.2.2	Mögliche Orte der Visualisierung	39
3.2.2.3	Mögliche Kodierungsformen.....	41
3.2.2.4	Spezielle Aspekte einzelner Regeltypen	44
3.2.2.5	Besondere Situationen.....	47
3.2.2.6	Standarddarstellung	49
3.2.2.7	Benutzer- und problemspezifische Adaption.....	53
3.2.2.8	Fazit	55
3.3	Anforderungen an die Interaktivität.....	56
3.3.1	Funktionelle Grundanforderungen.....	56
3.3.2	Präsentation der Interaktionsmöglichkeiten.....	58
3.3.2.1	Navigation innerhalb der Zuordnungsdarstellung	58
3.3.2.2	Selektion innerhalb der Zuordnungsdarstellung	58
3.3.2.3	Selektionsorientierte Interaktionsmöglichkeiten.....	59
3.3.2.4	Direkte Manipulation.....	60
3.3.3	Weitere Maßnahmen zur Unterstützung der interaktiven Planung.....	60
3.3.3.1	Darstellung der Existenz von Randbedingungen	60
3.3.3.2	Darstellung von Veränderungen.....	62

3.3.4	Fazit.....	63
4	Beschreibung einzelner Zuordnungsprobleme	64
4.1	Dienstplanung für das Pflegepersonal des Juliusspitals Würzburg	64
4.1.1	Beschreibung der Dienstplanung	64
4.1.1.1	Aufgabe der Dienstplanung	64
4.1.1.2	Pflegepersonal.....	64
4.1.1.3	Schichten	64
4.1.1.4	Urlaubsplanung	65
4.1.1.5	Randbedingungen.....	65
4.1.1.6	Dienstplanung als Zuordnungsproblem	67
4.1.2	Traditionelle Darstellung	67
4.1.3	Einordnung in die Grundtypen.....	70
4.1.3.1	Zuordnungsdarstellung.....	70
4.1.3.2	Randbedingungen.....	70
4.1.4	Beispiele für die resultierende Darstellung.....	71
4.1.5	Fazit.....	72
4.2	Dienstplanung für das Pflegepersonal der anästhesiologischen Klinik der Universität Würzburg.....	73
4.2.1	Unterschiede zur Dienstplanung des Juliusspitals	73
4.2.1.1	Aufgabe der Dienstplanung	73
4.2.1.2	Pflegepersonal.....	73
4.2.1.3	Schichten	73
4.2.1.4	Randbedingungen.....	73
4.2.2	Traditionelle Darstellung	74
4.2.3	Fazit.....	74
4.3	Schulstundenplanung:.....	75
4.3.1	Beschreibung der Schulstundenplanung.....	75
4.3.1.1	Aufgabe der Schulstundenplanung	75
4.3.1.2	Wochenstunden.....	75
4.3.1.3	Räume	76
4.3.1.4	Lehrer.....	76
4.3.1.5	Klassen.....	76
4.3.1.6	Fächer.....	77
4.3.1.7	Unterrichtsveranstaltungen.....	77
4.3.1.8	Unterrichtsstunde.....	77
4.3.1.9	Randbedingungen der Stundenplanung.....	77
4.3.1.10	Schulstundenplanung als Zuordnungsproblem	79
4.3.2	Traditionelle Darstellung	79
4.3.3	Einordnung in die Grundtypen.....	80
4.3.3.1	Zuordnungsdarstellung.....	80
4.3.3.2	Randbedingungen.....	81
4.3.4	Beispiel für die resultierende Darstellung	82
4.3.5	Fazit.....	83
4.4	Dienstplanung für das ärztliche Personal der Klinik für Anästhesiologie der Universität Würzburg.....	84
4.4.1	Beschreibung der Dienstplanung	84
4.4.1.1	Der Dienstplan	85
4.4.1.2	Der Verteilungsplan.....	86
4.4.1.3	Dienst- und Verteilungsplanung als Zuordnungsproblem.....	87
4.4.2	Traditionelle Darstellung	87

4.4.2.1	Traditionelle Darstellung des Dienstplans.....	87
4.4.2.2	Traditionelle Darstellung des Verteilungsplans	88
4.4.3	Einordnung in die Grundtypen.....	88
4.4.3.1	Zuordnungsdarstellung.....	88
4.4.3.2	Randbedingungen.....	89
4.4.4	Fazit.....	90
5	Entwurf einer Lösung	92
5.1	Verwandte Arbeiten.....	92
5.1.1	Toolkits.....	92
5.1.2	User Interface Management.....	93
5.1.3	Modell-basierte Ansätze.....	93
5.2	Generierung von Oberflächen zur interaktiven Lösung von Zuordnungsproblemen.....	95
5.2.1	Einführung.....	95
5.2.2	Wissensbasis	98
5.2.2.1	Grundstruktur	98
5.2.2.2	Erweiterung der Wissenshierarchie	100
5.2.2.3	Minimalkonfiguration eines Zuordnungsproblems	100
5.2.2.4	Ausblick	101
5.2.3	Bibliothek	102
5.2.3.1	Die abstrakte Basisklasse „Visualizable Object“	102
5.2.3.2	Die Visualisierungsmethode	102
5.2.3.3	Die Grundstruktur der Bibliothek.....	103
5.2.3.4	Hüllen-Klassen.....	104
5.2.3.5	Die Visualisierungs-klasse Zuordnungstabelle	104
5.2.3.6	Konfiguration einer Zuordnungsdarstellung.....	106
5.2.3.7	Visualisierung von Zuteilungen.....	107
5.2.3.8	Visualisierung von Restriktionsverletzungen	108
5.2.4	Schnittstelle	110
5.2.5	Präsentationstool.....	111
5.2.6	Fazit.....	112
6	Vorstellung und Evaluation einer erzeugten Oberfläche	114
6.1	Beschreibung der Oberfläche.....	114
6.1.1	Nachrichtenfenster.....	115
6.1.2	Hauptmenü.....	115
6.1.2.1	Verwaltung von Dienstplänen.....	116
6.1.2.2	Festlegung der Eckdaten	116
6.1.2.3	Erstellung von Dienstplänen.....	120
6.1.3	Unterstützung der interaktiven Planung	126
6.2	Evaluation.....	127
7	Zusammenfassung und Ausblick	130
7.1	Ergebnisse.....	130
7.2	Bewertung.....	131
7.3	Ausblick	131
Anhang A: Abbildungsverzeichnis		132
Anhang B: Quellen.....		134
Anhang C: Source-Code		136
Anhang D: Danksagung.....		138

1 Einleitung

1.1 Motivation

„An interface which is difficult to use will, at best, result in a high level of user errors. At worst, it will cause the software system to be discarded, irrespective of its functionality.“ Dieses eindeutige Urteil fällt [Sommerville 96] über die grundsätzliche Bedeutung einer Benutzeroberfläche. Eine solche Oberfläche ist also kein Nebenaspekt eines Programmes, sondern ein Hauptbestandteil, der ganz grundsätzlich über die Akzeptanz der Software beim Benutzer entscheidet – insbesondere bei Programmen, bei denen der Interaktivität, also den Eingriffsmöglichkeiten des Anwenders über die Oberfläche, eine zentrale Rolle zukommt.

So auch bei Programmen zur interaktiven Lösung von Zuordnungsproblemen; unter einem Zuordnungsproblem versteht man die Aufgabe, Nachfrageobjekten unter möglichst weitgehender Einhaltung von gewichteten Randbedingungen Angebotsobjekte zuzuordnen, wie zum Beispiel bei der Dienstplanung¹ für das Pflegepersonal eines Krankenhauses: Schichten müssen unter der Einhaltung von Dienstvorschriften mit Personal besetzt werden. Auch die Schulstundenplanung und die Ressourcenbelegungsplanung eines Betriebs sind Beispiele für alltägliche Zuordnungsprobleme.

Da Zuordnungsprobleme weit verbreitet, meistens sehr komplex und nur unter großem Aufwand von Hand zu lösen sind, liegt der Einsatz von Computern zu ihrer Lösung nahe. Eine vollautomatische Lösung eines realen² Zuordnungsproblems erweist sich jedoch oft als problematisch; zum Beispiel erfordert eine solche Lösung eine komplette Spezifikation des Problems, was oft sehr aufwendig und in manchen Fällen nicht möglich ist [Busch 97]. Zur Vermeidung unter anderen dieser Problematik empfiehlt [Kurbel 93] am Beispiel der Ressourcenbelegungsplanung eine interaktive Lösung von Zuordnungsproblemen, also einen benutzergesteuerten Lösungsprozeß, bei dem der Anwender mit der Software in einem Dialog steht; so kann er den Lösungsprozeß nach seinen Vorstellungen steuern und sein Expertenwissen einbringen, indem er zum Beispiel die Erfüllung von Randbedingungen sicherstellt, die dem System nicht oder nur teilweise bekannt sind. Eine vollständige Spezifikation des Zuordnungsproblems ist dann nicht mehr notwendig.

Wie bereits erläutert, kommt bei einer solch zentralen Rolle der Interaktivität der Oberfläche noch mehr Gewicht zu. Trotz dieser Erkenntnis verfügen vorhandene Zuordnungswerkzeuge³ oft über effektive Lösungsmethoden, jedoch über keine moderne, ausgereifte Oberfläche [vgl. Busch 97]. Ein Grund für dieses Defizit ist sicherlich, daß

¹ Zuordnungsprobleme werden oft informell als Planungen bezeichnet, obwohl sie eigentlich eine eigene Problemklasse bilden und nicht zu der, allerdings verwandten, Problemklasse Planung gehören (vgl. [Puppe 90]).

² Es gibt eine Reihe von Zuordnungsproblemen, die nur von theoretischem Interesse sind und sich durch eine sehr einfache Spezifikation auszeichnen, etwa die Aufgabe, n Damen auf einem $n \times n$ Schachbrett zu platzieren, ohne daß sie sich gegenseitig bedrohen.

³ Als Werkzeug zur Lösung eines Zuordnungsproblems (kurz Zuordnungswerkzeug) wird hier ein System verstanden, das für dieses Zuordnungsproblem sowohl die Mengen der Angebotsobjekte, der Nachfrageobjekte und der Randbedingungen verwaltet als auch die eigentliche Zuordnung durchführen kann. Das Verfahren, mittels dessen ein solches Werkzeug den Nachfrageobjekten Angebotsobjekte zuordnet, wird als Lösungsmethode oder (Lösungs-) Algorithmus bezeichnet. Ein einzelnes Werkzeug kann mehrere Algorithmen implementieren.

Algorithmen meistens allgemein oder nur mit geringen Modifikationen zur Lösung von beliebigen Zuordnungsproblemen verwendet werden können, Benutzeroberflächen aber im Gegensatz dazu für jedes konkrete Zuordnungsproblem mit oft großem Aufwand speziell entwickelt werden müssen.

So existiert mit COKE [Poeck 95] ein Baukasten, der weitgehend universell zur Lösung von Zuordnungsproblemen eingesetzt werden kann. Von den verschiedenen in COKE realisierten Lösungsmethoden wurde mit „Vorschlagen- und- Vertauschen“ [Poeck 95] auch ein zur interaktiven Lösung geeigneter Algorithmus implementiert, der sich bisher bei so verschiedenen Zuordnungsproblemen wie der Schulstundenplanung (REST [Poeck 95]) oder der Ressourcenbelegungsplanung (WIZARD [Hestermann 97]) bewährt hat. Während bei diesen Beispielen derselbe Algorithmus eingesetzt werden konnte, mußten die problemspezifischen Oberflächen per Hand entworfen, implementiert und an COKE angekoppelt werden, was in beiden Fällen mit großem Aufwand verbunden war [Busch 97]).

Problematisch ist auch, daß die Erstellung qualitativ hochwertiger Oberflächen Fachwissen auf dem Gebiet des Interface Design erfordert, das oft nicht im ausreichenden Maß zur Verfügung steht.

1.2 Ziel und Aufbau der Arbeit

Ziel der Arbeit ist es, ein Konzept zu entwickeln, mit dem die Erstellung von benutzerfreundlichen graphischen Oberflächen zur interaktiven Lösung beliebiger Zuordnungsprobleme (oder zumindest für eine möglichst große Gruppe von Zuordnungsproblemen) weitgehend automatisiert werden kann. Dabei soll der Schwerpunkt der Arbeit auf zwei zentralen Aspekten solcher Oberflächen liegen – zum einen der Darstellung des eigentlichen Zuordnungsproblems, zum anderen der Unterstützung der interaktiven Lösung des Zuordnungsproblems, wobei letztere noch in eine funktionelle (Welche Manipulationsmöglichkeiten sollen bereitgestellt werden?) und eine graphische (Wie sollen diese Möglichkeiten graphisch repräsentiert werden?) Komponente unterteilt werden kann. Andere Gesichtspunkte solcher Oberflächen, wie die Verwaltung und graphische Repräsentation von Angebots- oder Nachfragemenge, sollen im Rahmen dieser Arbeit nur am Rande behandelt werden.

Die grundsätzliche Eignung des Lösungsansatzes für die Problemstellung soll anhand eines Beispiels, einer Benutzeroberfläche für die in Coke modellierte Dienstplanung für das Pflegepersonal eines Krankenhauses, demonstriert werden.

Im zweiten Kapitel sollen der Begriff des Zuordnungsproblems und weitere in diesem Zusammenhang wesentliche Fachausdrücke näher erläutert und formal definiert werden.

Mittels allgemeiner Erkenntnisse aus den Bereichen Software Engineering und Interface Design sollen dann im dritten Kapitel Grundanforderungen an eine Benutzeroberfläche erarbeitet werden, die diese unabhängig von der konkreten Aufgabe erfüllen sollte. Dieses Anforderungsprofil soll anschließend - ausgehend von der Definition von Zuordnungsproblemen - um die speziellen Ansprüche an Oberflächen von interaktiven Werkzeugen zur Lösung von Zuordnungsproblemen erweitert werden, wobei primär die zentralen Aufgaben der Darstellung des Zuordnungsproblems und der interaktiven Manipulationsmöglichkeiten behandelt werden sollen.

Im vierten Kapitel werden Beispiele von praktischen Zuordnungsproblemen betrachtet und ihre besonderen Ansprüche an eine graphische Oberfläche analysiert, um das in Kapitel 3 erstellte Anforderungsprofil auf seine Plausibilität zu überprüfen. Explizit sollen hier die Dienstplanung für das Pflegepersonal eines Krankenhauses am Beispiel des Juliusspitals und der anästhesiologischen Klinik der Universität Würzburg, die Schulstundenplanung und die Dienstplanung für Ärzte wiederum am Beispiel der anästhesiologischen Klinik der Universität Würzburg betrachtet werden. Dabei soll der Schwerpunkt entsprechend dem Thema der Arbeit auf den Ansprüchen dieser realen Probleme an eine Benutzeroberfläche liegen und nur so weit zum Verständnis nötig in einer allgemeinen Beschreibung dieser Zuordnungsprobleme.

Nach dem Abschluß der Anforderungsanalyse sollen im fünften Kapitel bestehende Ansätze zur Konstruktion von Benutzeroberflächen vorgestellt werden. Anschließend soll ein eigenes Konzept vorgestellt werden, das die in den vorhergehenden Abschnitten spezifizierten Ansprüche möglichst weitgehend erfüllt.

Im sechsten Kapitel wird eine Oberfläche für die Dienstplanung des Pflegepersonals eines Krankenhauses vorgestellt, die mittels eines Prototyps der vorgeschlagenen Lösung generiert wurde. Auch wird der Versuch beschrieben, mit dieser Oberfläche einen Plan für die Anästhesiologie der Universität zu erstellen.

Das abschließende siebte Kapitel soll die Ergebnisse zusammenfassen, beurteilen und einen Ausblick über die Punkte geben, an denen eventuelle weiterführende Arbeiten anknüpfen könnten.

2 Grundbegriffe

2.1 Definition von Zuordnungsproblemen

Formal soll im Rahmen dieser Arbeit⁴ ein (finites)⁵ Zuordnungsproblem $ZP(N, A, K)$ wie folgt definiert werden:

Seien $N = \{n_1, \dots, n_k\}$ und $A = \{a_1, \dots, a_l\}$ endliche Mengen.
 Sei $Z := \{x_{11}, \dots, x_{kl} \mid x_{ij} \in \{0, 1\} \forall i \in \{1, \dots, k\}, j \in \{1, \dots, l\}\}$.
 Sei $K : Z \rightarrow \mathbb{R}$ eine Abbildung von Z auf $\mathbb{R}$.
 Die Aufgabe, ein $z \in Z$ zu finden, so daß $z \in Z$ gilt $K(z) = K^*(z)$
 (alternativ $K(z) \leq r$ für ein vorgegebenes $r \in \mathbb{R}$),
 heißt Zuordnungsproblem $ZP(N, A, K)$.

Definition 1: Zuordnungsproblem

Die Menge N heißt auch Nachfragemenge, die Elemente von N Nachfrageobjekte, die Menge A Angebotsmenge, die Elemente von A entsprechend Angebotsobjekte. Die Funktion K wird als Kostenfunktion bezeichnet. Im Fall $x_{ij}=1$ sagt man, daß Angebotsobjekt a_j sei dem Nachfrageobjekt n_i zugeordnet. Ein Element der Menge z wird auch als Zuordnung bezeichnet und kann als Abbildung der Menge der Nachfrageobjekte auf die Menge der Angebotsobjekte interpretiert werden. In Abgrenzung zum Begriff der Zuordnung soll der Begriff der Zuteilung einen einzelnen Funktionswert der Zuordnung, also ein Nachfrageobjekt und das ihm zugeteilte Angebotsobjekt, bezeichnen. Eine Zuteilung ist somit ein Tupel (Angebotsobjekt, Nachfrageobjekt), die Zuordnung die Menge aller dieser Tupel.

Gilt zusätzlich die Bedingung, daß nicht zwei Nachfrageobjekten dasselbe Angebotsobjekt zugeordnet werden kann, spricht man von einer 1-zu-1 Zuordnung, sonst allgemein von einer 1-zu-n Zuordnung. Sind bei der Zuordnung eigentlich unendlich große Mengen beteiligt (wenn etwa wie bei der Ressourcenbelegungsplanung Arbeitsschritten Intervalle auf einem Zeitstrahl zugeordnet werden sollen), spricht man von einem kontinuierlichen Zuordnungsproblem, sonst von einem diskreten Zuordnungsproblem. Bei einer Mehrfachzuordnung sollen den Nachfrageobjekten nicht

⁴ In der Literatur finden sich unterschiedliche Definitionen, je nach fachlicher Provenienz - siehe etwa die Definition im Rahmen einer mathematischen Formelsammlung [Bronstein 95] oder [Poeck 95]

⁵ Während der Fall abzählbar unendlicher oder gar überabzählbarer Mengen theoretisch denkbar ist, wird in der Realität beim Einsatz von Computern bei der Lösung von Zuordnungsproblemen immer mit eventuell sehr großen, aber dennoch endlichen Vergrößerungen der ursprünglichen Mengen gearbeitet werden. Deshalb ist es sinnvoll, sich auf die einfacheren endlichen Mengen zu beschränken und so die Problematik unendlicher Mengen ganz zu vermeiden.

nur je ein Element aus einer Angebotsmenge A zugeordnet werden, sondern jeweils ein Element aus mehreren Angebotsmengen $A_1, \dots, A_p$. Ein Beispiel dafür ist die Schulstundenplanung⁶, bei der einer Unterrichtsstunde (bestehend aus Lehrer, Klasse(n) und Fach) sowohl ein Element der Angebotsmenge Wochenstunden als auch ein Element der Angebotsmenge Räume zugewiesen werden muß.

Interpretiert man eine Zuordnung z als Abbildung der Nachfragemenge auf die Angebotsmenge(n), dann ist eine Zuordnung genau dann eine 1-1 Zuordnung, wenn sie injektiv ist. Eine Mehrfachzuordnung entspricht bei dieser Sichtweise einer Abbildung der Nachfragemenge auf das Kreuzprodukt der Angebotsmengen oder einer Komposition von Abbildungen: erst eine Abbildung eines Nachfrageobjektes auf ein Objekt der Angebotsmenge A_1 , dann Abbildung des neuen Nachfrageobjektes, das durch diese erste Abbildung entstanden ist, auf ein Element von A_2 , bis hin zu einer Abbildung eines Nachfrageobjektes bestehend aus je einem Element aller Mengen bis auf A_p auf ein Element der Angebotsmenge A_p .

Auf den ersten Blick muß man bei Zuordnungsproblemen oft einem Nachfrageobjekt nicht nur 1, sondern n Angebotsobjekte zuordnen, beispielsweise wenn man bei der Schichtplanerstellung die Schichten als Nachfrager und die Mitglieder des Personals als Angebotsobjekte auffaßt und manchen Schichten mehr als eine Arbeitskraft ~~z~~geteilt werden muß. In einem solchen Fall kann man jedes Nachfrageobjekt aufgrund der Endlichkeit der Angebotsmenge logisch in n separate Nachfrageobjekte aufteilen, denen dann jeweils nur ein einzelnes Angebotsobjekt zuzuordnen ist, also etwa das Nachfrageobjekt Dienstag, Frühschicht mit einem Personalbedarf von m Personen in die m Einzelobjekte $(\text{Dienstag, Frühschicht})_1, \dots, (\text{Dienstag, Frühschicht})_m$ mit einem Personalbedarf von jeweils einer Arbeitskraft aufteilen. Der Fall einer m -zu- n Zuordnung läßt sich somit immer zu einer 1-zu- n Zuordnung reduzieren, und somit werden auch solche Probleme von obiger Definition erfaßt. Analog kann jede 1-zu- n Zuordnung formal weiter zu einer 1-zu-1 Zuordnung vereinfacht werden.

⁶ Die Schulstundenplanung wird im Kapitel 4.3 kurz beschrieben. Eine ausführliche Beschreibung findet sich in [Busch 97].

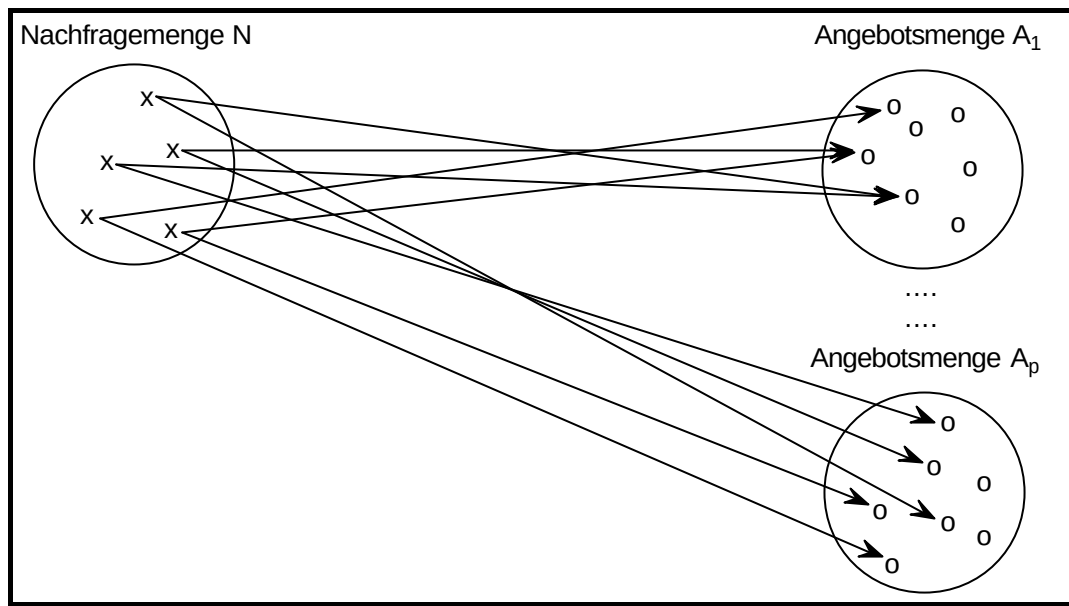


Abbildung 1: Darstellung einer Mehrfachzuordnung, wobei die Abbildung von N auf A_1 eine 1-n Zuordnung und die Abbildung von N auf A_p eine 1-1 Zuordnung ist.

2.2 Klassen, Objekte und Attribute der Zuordnung

Algorithmen, die Zuordnungsprobleme lösen, differenzieren nur zwischen Angebots- und Nachfrageobjekten. Für die Diskussion von Zuordnungsproblemen und auch für die graphische Darstellung ist es jedoch sinnvoll, diese Objekte – falls möglich – weiter zu strukturieren. Dazu bietet sich die Verwendung des in der Informatik populären Konzepts der Objektorientierung an. Schließlich setzen sich aus der Sicht des Anwenders bei vielen Zuordnungsproblemen die Elemente der Angebots- oder Nachfragemenge aus mehreren einzelnen Objekten zusammen, wobei man unter einem Objekt üblicherweise eine Abstraktion einer Entität der realen Welt versteht ([Sommerville 96], [Jacobson 92]). Definiert man Klassen als Schemata oder Schablonen für Objekte, dann kann man davon sprechen, daß zum Beispiel ein Nachfrageobjekt der Schulstundenplanung, eine Unterrichtsstunde, aus einem Objekt (oder Instanz) der Klasse Lehrer, einem Objekt der Klasse Schulklasse und einem Objekt der Klasse Schulfach besteht. Eine Pflegekraft, ein Angebotsobjekt der Dienstplanung für das Pflegepersonal eines Krankenhauses oder auch eine Instanz der Klasse Pflegekraft, läßt sich im Kontext der Dienstplanung jedoch nicht sinnvoll in weitere Objekte aufgliedern; solche Objekte werden manchmal als Grundobjekte der Zuordnung bezeichnet, die zugehörigen Klassen als Grundklassen ([Busch 97]).

Ein Attribut ist eine Abbildung eines Objekts auf gegebene Wertemengen. Der Definitionsbereich eines Attributs wird immer mittels Klassen formuliert, das heißt, entweder ist ein Attribut für alle Objekte einer Klasse definiert oder für keines. Beispiele für Attribute der Klasse Pflegekraft der Dienstplanung wären Wochenstunden, eine Abbildung von Pflegekräften in die positiven reellen Zahlen, oder Qualifikation, eine Abbildung von Pflegekräften in die Wertemenge {Praktikant(in), Schüler(in), Examinierte(r), Schichtleiter(in)}.

2.3 Randbedingungen

2.3.1 Definition

Die Kostenfunktion dient dazu, Zuordnungen durch positive reelle Zahlen zu bewerten, so daß der Computer verschiedene Lösungsvorschläge vergleichen und den relativ besten auswählen kann. Diese Bewertung sollte der eines menschlichen Experten möglichst ähnlich sein. Deswegen versucht man, die Beurteilungskriterien eines solchen Experten auf exakte Randbedingungen, auch Regeln, Restriktionen oder englisch Constraints genannt, abzubilden, die vom Computer überprüft werden können. Diese sind mathematisch gesehen wiederum Abbildungen von Zuordnungen auf positive reelle Zahlen, konzeptionell Regeln, deren Nichtbeachtung von der Schwere des Verstoßes abhängige Kosten (auch Strafpunkte genannt) verursacht. Die Gesamtheit der Randbedingungen bildet dann die Kostenfunktion, meistens über eine einfache Summe der Funktionswerte der einzelnen Randbedingungen. Es können jedoch auch kompliziertere Funktionen verwendet werden; wesentlich ist nur, daß der Weg zur Minimierung der Kostenfunktion über eine Minimierung der Funktionswerte der Randbedingungen führt. Wenn also eine Zuordnung z so zu einer Zuordnung z' verändert wird, daß mindestens eine der bisher verletzten Regeln r_i in geringerem Maße übertreten wird (also $r_i(z') < r_i(z)$) und keine der Randbedingungen schwerer verletzt wird (für keine der Regeln r gilt: $r(z') > r(z)$), dann muß die Kostenfunktion die Zuordnung z' insgesamt besser bewerten als die Zuordnung z (genauer $K(z') < K(z)$). Definition 2 versucht, diesen Sachverhalt formal darzustellen:

Sei $Z(N, A, K)$ ein Zuordnungsproblem.

Sei $f : \mathbb{R}^m \rightarrow \mathbb{R}$, wobei jede Partialfunktion von f streng monoton steigend ist.

Die endliche Menge $R = \{r_1, \dots, r_m\}$ mit r_i Abbildung von Z in $\mathbb{R}$ und

$K(z) = f(r_1(z), \dots, r_m(z))$,

heißt Menge der Randbedingungen des Zuordnungsproblems $Z(N, A, K)$ unter f .

Definition 2: Randbedingung

Diese Definition reflektiert die menschliche Sichtweise, daß eine Kostenfunktion aus einzelnen Randbedingungen zusammengesetzt ist, und integriert sie in die Definition eines Zuordnungsproblems, rein mathematisch ist sie weniger sinnvoll, da nichts ausgesagt wird, eine Randbedingung ist ihr zufolge eine Kostenfunktion und umgekehrt. Prinzipiell kann eine Randbedingung beliebig kompliziert sein; hier sollen nun einige Grundtypen definiert werden, auf die sich die meisten der in der Praxis auftretenden Regeln zurückführen lassen (siehe Kapitel 4).

2.3.2 Klassifizierung von Randbedingungen

2.3.2.1 Einfache Randbedingungen⁷

Die einfachste vorstellbare Randbedingung bewertet eine Zuteilung eines einzelnen Angebotsobjekt an ein einzelnes Nachfrageobjekt unabhängig von anderen Zuteilungen. Ein Beispiel aus der Schichtdienstplanung für das Pflegepersonal eines Krankenhauses ist eine Pflegekraft, die eine bestimmte Schicht nicht arbeiten will. Wird sie ihr dennoch zugeteilt, sollte diese Zuteilung mit Strafpunkten bewertet werden. Auch das Gegenteil ist vorstellbar, nämlich eine Pflegekraft, die eine bestimmte Schicht gerne arbeiten möchte. Die Berücksichtigung dieses Wunsches könnte über eine Regel, die Kosten verursacht, falls sie ihr nicht zugeteilt wird, umgesetzt werden. Generell soll also die Vermeidung, bzw. Einhaltung einzelner Zuteilungen gewährleistet werden. Formal kann man solche Randbedingungen respektive wie folgt darstellen, wobei π_{ij} als die Abbildung einer Zuordnung auf ihr Element x_{ij} definiert sei:

$$r(z) = \begin{cases} k_{ij} & \text{falls } p_{ij}(z) = 1 \\ 0 & \text{sonst} \end{cases} \quad (k_{ij} \geq 0)$$

Beispiel 1: Vermeidung der einzelnen Zuteilung (n_i, a_j)

oder

$$r(z) = \begin{cases} 0 & \text{falls } p_{ij}(z) = 1 \\ k_{ij} & \text{sonst} \end{cases} \quad (k_{ij} \geq 0)$$

Beispiel 2: Durchsetzung der einzelnen Zuteilung (n_i, a_j)

Natürlich könnte man formal Regeln des einen Typs vermeiden, indem man sie durch äquivalente Regeln des anderen Typs ersetzt: Die Präferenz eines Pflegers für gewisse Schichten kann man zum Beispiel auch durch eine Regel ausdrücken, welche die Zuteilung des Pflegers zu allen übrigen Schichten mit Kosten belegt. Allerdings ist eine solche Sichtweise wenig intuitiv und führt sehr häufig dazu, daß unnötig viele oder unnötig komplizierte Regeln benötigt werden.

Im allgemeinen ist es zu aufwendig, Regeln zu formulieren, die nur zwei konkrete Objekte betreffen. Im Rahmen der Schulstundenplanung könnte etwa die Regel gelten, daß Hauptfächer nicht nachmittags unterrichtet werden sollten. Für jedes einzelne

⁷ Die hier getroffene Klassifizierung stellt einen Versuch dar, Randbedingungen mit Blick auf ihre Formulierung und ihre graphische Darstellung in intuitiv verständliche Klassen einzuteilen und orientiert sich somit nicht an in der Mathematik gebräuchlichen, sich auf die anwendbaren Lösungsverfahren beziehenden Attributen wie „linear“ oder „quadratisch“.

Hauptfach und für jede Nachmittagsstunde eine eigene Regel zu spezifizieren, die diese Kombination mit Kosten belegt, ist offensichtlich nicht praktikabel.

Mathematisch ausgedrückt bietet sich daher folgende Vorgehensweise an: Man faßt Teilmengen der Nachfrage- oder Angebotsmengen zu Mengen zusammen, zum Beispiel die Fächer Mathematik, Deutsch und Englisch zur Menge der Hauptfächer oder alle Pflegekräfte, die über die entsprechende Qualifikation verfügen, zur Menge der potentiellen Schichtleiter. Dann kann man obige Randbedingung ohne direkte Bezugnahme auf einzelne Objekte, sondern unter Verwendung von Mengen verallgemeinert formulieren:

$$\begin{aligned}
 & \text{Sei } M \text{ ? die Menge der Nachmittagsstunden, } H \text{ ? } N \text{ die Menge der} \\
 & \text{Unterrichtsstunden, deren Fächer Elementeder Menge der Hauptfächer sind.} \\
 & \text{Sei } h(z, i, j) \text{ ? } \begin{cases} k & \text{falls } p_{ij}(z) \text{ ? } 1 \text{ ? } a_i \text{ ? } H \text{ ? } n_j \text{ ? } M \text{ ? } (k \text{ ? } ? \text{ ?}) \\ 0 & \text{sonst} \end{cases} \\
 & r(z) \text{ ? } \sum_{i,j} h(z, i, j)
 \end{aligned}$$

Beispiel 3: Keine Hauptfächer am Nachmittag (Vermeidung einzelner Zuteilungen unter Verwendung von Mengen)

Dabei dient die Summenfunktion dazu, eine Regel auf die gesamte Zuordnung, nicht nur auf einzelne Zuteilungen, anzuwenden. Allerdings dient das nur der Vereinfachung der Formulierung – die zugrundeliegende Randbedingung betrifft immer noch ausschließlich einzelne Zuteilungen, unabhängig von anderen Zuteilungen. Will man den abstrakten Begriff einer Menge bei der Formulierung vermeiden, so bietet sich als Alternative eine Formulierung durch Attribute an; Wieder gilt, daß dies nur dazu dient, sich der menschlichen Sichtweise anzunähern, und mathematisch äquivalent zur Formulierung mittels Mengen ist. Schließlich werden nur Mengen implizit durch Attribute gebildet, und somit Mengen geformt, die sich dynamisch durch die Modifizierung von Attributwerten ändern können.

Ein Beispiel aus der Schulstundenplanung macht deutlich, daß die Verwendung von Attributen die Formulierung von Randbedingungen intuitiver machen kann: Jeder Schulraum kann nur eine gewisse Anzahl von Schülern aufnehmen. Eine Zuteilung eines Raumes an eine Klasse, die mehr Schüler umfaßt als der Raum aufnehmen kann, sollte unbedingt vermieden werden. Formal könnte jede Klasse das Attribut „Schülerzahl“ und jeder Raum das Attribut „Kapazität“ haben, beides Abbildung in die natürlichen Zahlen. Eine Regel, die obiges Problem vermeiden soll, würde dann wie folgt lauten:

$$\text{Sei } h(z, i, j) = \begin{cases} k & \text{falls } p_{ij}(z) = 1 \text{ \& } \text{Schülerzahl}(n_i) \leq \text{Kapazität}(a_j) \\ 0 & \text{sonst} \end{cases}$$

$$r(z) = \sum_{i,j} h(z, i, j)$$

Beispiel 4: Berücksichtigung der Raumkapazität (Vermeidung einzelner Zuteilungen unter Verwendung von Attributen)

Genauso ließe sich die in Beispiel 3 aufgeführte Regel intuitiver durch das Attribut Fachtyp, das eine Unterrichtsstunde auf ein Element der Wertmenge {Hauptfach, Nebenfach} abbildet, formulieren.

2.3.2.2 Komplexe Randbedingungen

Die logische nächste Stufe sind Regeln, die nicht nur eine einzelne Zuteilung betreffen, sondern Kombinationen oder Konstellationen von Zuteilungen, also im einfachsten Fall

$$r(z) = \begin{cases} k_{ijkl} & \text{falls } p_{ij}(z) = 1 \text{ \& } p_{kl}(z) = 1 \text{ \& } (k_{ijkl} \leq ?) \\ 0 & \text{sonst} \end{cases}$$

Beispiel 5: Vermeidung der Zuteilungskombination (n_i, a_j) und (n_k, a_l)

zur Vermeidung von Kombinationen von Zuteilungen oder

$$r(z) = \begin{cases} 0 & \text{falls } p_{ij}(z) = 1 \text{ \& } p_{kl}(z) = 1 \\ k_{ijkl} & \text{sonst } (k_{ijkl} \leq ?) \end{cases}$$

Beispiel 6: Durchsetzung der Zuteilungskombination (n_i, a_j) und (n_k, a_l)

zur Durchsetzung von Kombinationen von Zuteilungen. Prinzipiell kann hier also nicht mehr eine einzelne Zuteilung als Verursacher eines Regelverstoßes bestimmt werden, sondern alle an der Kombination beteiligten Zuteilungen tragen zum Regelverstoß bei. Wie im Falle einfacher Randbedingungen können hier zwecks Formulierung wieder Mengen, bzw. Attribute verwendet werden.

In der Schulstundenplanung darf es normalerweise keine zwei Unterrichtsstunden mit demselben Lehrer oder derselben Klasse geben, die auf dieselbe Wochenstunde abgebildet werden. Auch kann ein Raum in der Regel pro Wochenstunde nur eine Unterrichtsstunde aufnehmen, alles Beispiele von Randbedingungen, die bestimmte Zuteilungskombinationen vermeiden sollen. Sollen andererseits bestimmte Unterrichtsstunden derselben Wochenstunde oder mehrere Pflegekräfte derselben Schicht

zugeteilt werden (aneinander gekoppelt werden), so wäre das ein Beispiel für eine Regel, die Zuteilungskombinationen durchsetzen soll.

Randbedingungen über mehrere Zuteilungen können prinzipiell beliebig kompliziert sein und beliebige Zuteilungen betreffen; eine (denkbare) Randbedingung der Dienstplanung für das Pflegepersonal eines Krankenhauses, die durchsetzen soll, daß die Schichten möglichst gleichmäßig zwischen den einzelnen Pflegekräften verteilt werden, beträfe zum Beispiel sämtliche Zuteilungen der Zuordnung gleichzeitig. Dennoch soll nun der Begriff der komplexen Randbedingung weiter strukturiert werden, indem mehrere spezielle Typen von komplexen Randbedingungen definiert werden.

1. Komplexe Randbedingungen über Zuteilungen eines einzelnen Objekts

Solche Randbedingungen betreffen nur Zuteilungen, die in (mindestens) einem Objekt übereinstimmen, also zum Beispiel in der Dienstplanung für das Pflegepersonal eines Krankenhauses nur Zuteilungen, an denen eine einzelne Schicht oder die Pflegekraft A beteiligt ist, oder in der Schulstundenplanung nur Unterrichtsstunden, an denen das Fach Latein und die Schulklasse 5a beteiligt ist.

2. Vermeidung

Dieser Typ besteht nur aus Vermeidungen einzelner Zuteilungen, die durch ein logisches „und“ verknüpft sind.

3. Durchsetzung

Dieser Typ besteht nur aus Durchsetzungen einzelner Zuteilungen, die durch ein logisches „und“ verknüpft sind.

4. Ziel- oder Grenzwert einer arithmetischen Funktion

Hier werden Zuteilungen auf einen (reellen) Wert abgebildet, der möglichst nahe an einem arithmetischen Zielwert, bzw. unter oder über einem Grenzwert liegen soll. Die Schwere der Verletzung hängt davon ab, wie stark der Zielwert verfehlt, bzw. der Grenzwert überschritten wurde. Regeln dieses Typs können Zuteilungen durchsetzen, vermeiden oder beides gleichzeitig. Werden nur Zuteilungen eines Objektes betrachtet, so kann man diesen Zahlenwert auch als arithmetisches Attribut eines Objektes interpretieren. Zum Beispiel soll ein Pfleger im Rahmen der Dienstplanung für das Pflegepersonal eines Krankenhauses die vorgeschriebene Zahl von Stunden (Soll-Stunden) absolvieren, weder mehr noch weniger. Die entsprechende Randbedingung würde dann die Schichtdauer aller Zuteilungen einer Pflegekraft summieren (Ist-Stunden) und eine Differenz zu den Soll-Stunden mit Strafpunkten belegen.

5. Reihenfolge/ Muster

Eine Reihenfolge bezüglich einer totalgeordneten Menge, zum Beispiel der Zeit, muß eingehalten werden: z.B. muß in der Ärzteausbildung der Ausbildungsabschnitt Chirurgie vor dem Ausbildungsabschnitt Intensivstation absolviert werden. Sehr häufig kommen solche Randbedingungen in der Ressourcenbelegungsplanung einer Fabrik vor – die

(zeitlich) richtige Reihenfolge der Arbeitsschritte ist hier wesentlich. Bei der Dienstplanung für das Pflegepersonal eines Krankenhauses kann es bestimmte Schichtenfolgen oder –muster geben, denen der Dienstplan einer Pflegekraft möglichst weitgehend entsprechen sollte. Auch Regeln dieses Typs können Zuteilungen durchsetzen, vermeiden oder beides gleichzeitig.

2.3.2.3 Zusammenfassung

Abbildung 2 versucht, einen Überblick über die verschiedenen Typen von Randbedingungen zu geben und diese anhand von Beispielen aus der Pflegeplanung im Krankenhaus zu veranschaulichen. Dabei ist die Einordnung einer Randbedingung nicht immer eindeutig möglich, da sich die einzelnen Regeltypen überschneiden können; eine Regel der Dienstplanung für das Pflegepersonal eines Krankenhauses wie „Nicht mehr als 5 Nachtschichten in Folge“ kann man als komplexe Vermeidung (Vermeidung von 6 Nachtschichten in Folge) und auch als arithmetischen Zielwert über die Zuteilungen einer Pflegekraft interpretieren. (Maximale Kette von Nachtschichten < 6). Mit großem Aufwand könnte man diese Regel auch noch als komplexe Durchsetzung vorstellen – alle denkbaren Kombinationen über 6 Tage bis auf die mit 6 Nachtschichten in Folge sollen durchgesetzt werden. Allerdings stellt sich dieses Problem in der Praxis kaum – die Einteilung fast aller Regeln realer Zuordnungsprobleme (siehe Kapitel 4) in das vorgestellte System ist eindeutig möglich.

Abbildung 2 stellt einfache Randbedingungen, also Regeln über einzelne Zuteilungen, und komplexe Regeln über Zuteilungen eines einzelnen Objekts in den Mittelpunkt, da sie, so lassen zumindest empirische Untersuchung (siehe Kapitel 4) vermuten, bei weitem die Mehrheit der Restriktionen realer Zuordnungsprobleme bilden. Natürlich können Regeln etwa bezüglich der Reihenfolge prinzipiell auch über beliebige Zuteilungen auftreten, da sie in der Praxis jedoch kaum vorzukommen scheinen, sind sie in Abbildung 2 nicht eingezeichnet.

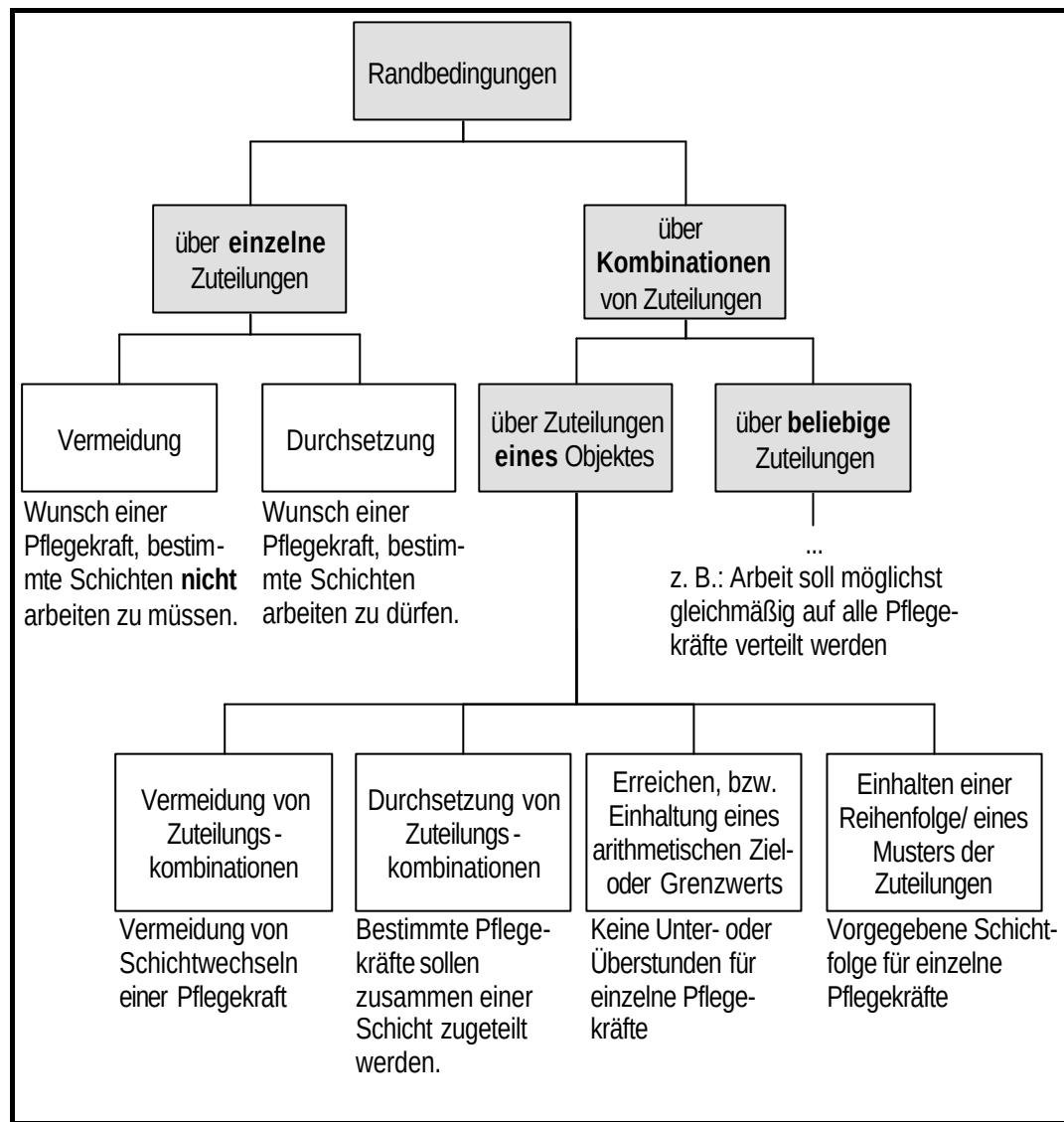


Abbildung 2: Überblick über die Typen von Randbedingungen

2.3.3 Harte und weiche Randbedingungen

Anders als in den Definitionen 1 und 2⁸ wird in der Literatur bei Optimierungsproblemen, zu denen Zuordnungsprobleme mathematisch gesehen gehören [Bronstein 95], häufig zwischen harten Randbedingungen, gegen die nicht verstoßen werden darf, und weichen Randbedingungen, die wünschenswerte Optimierungsbedingungen darstellen, unterschieden. Während eine solche Einteilung insbesondere bei den exakten Lösungsverfahren der Mathematik sinnvoll und auch notwendig ist, erscheint es gerade bei der interaktiven Lösung von Zuordnungsproblemen günstiger, diese scharfe Trennungslinie aufzuheben und die Bedeutung von Randbedingungen ausschließlich über

⁸ Formal ist die in Definition 1 aufgeführte Bedingung, daß jedem Nachfrageobjekt genau ein Angebotsobjekt zugeordnet werden soll, eine harte Randbedingung.

ihre Gewichtung auszudrücken. Schließlich kann ein menschlicher Planer oft einen Verstoß gegen eine an sich unverletzbare Randbedingung durch eine nachträgliche Korrektur der Ausgangssituation oder eine informelle Absprache aufheben.

Eine Lehrkraft an einem Tag zum Unterricht einzuteilen, an dem sie nicht an der Schule ist, etwa weil sie an einer anderen Schule unterrichtet, ist gemäß obiger Definition sicher ein Verstoß gegen eine harte Randbedingung. Ein menschlicher Planer kann jedoch eventuell durch eine kurze Nachfrage erreichen, daß die Lehrkraft den Einsatz in der anderen Schule auf einen anderen Wochentag verschiebt und somit an diesem Tag doch zur Verfügung steht.

In der Anästhesiologie der Universitätsklinik Würzburg liegen zwischen Spätschicht des einen und Frühschicht des nächsten Tages nur 8 Stunden Pause. Da der Gesetzgeber ein Minimum von 10 Stunden Ruhe zwischen zwei Schichten vorschreibt (harte Regel), kann eine Pflegekraft nicht beide Schichten hintereinander arbeiten; dennoch wird in seltenen Ausnahmefällen eine Pflegekraft zu diesen beiden Schichten eingeteilt, die dann zum Beispiel die Frühschicht zwei Stunden später als üblich beginnt. In einem solchen Fall muß sich der menschliche Planer allerdings unter anderem vergewissern, daß die Pflegekraft einverstanden ist und daß in der Frühschicht aller Voraussicht nach kein Hochbetrieb herrschen wird, so daß die ersten zwei Stunden auch mit einer Pflegekraft weniger als normal zu bewältigen sind.

Will man dennoch Randbedingungen einführen, die kategorisch eingehalten werden sollen, so kann man dieses Ziel erreichen, indem man sie höher bewertet als die Kombination aller möglichen Verstöße gegen alle andere Randbedingungen oder schlicht mit sehr hohen Strafpunkten. Die Realisierung eigentlich harter Randbedingungen mittels der Gewichtung +? erscheint problematisch, da sie mathematisch gesehen eine Lösung, die nur eine solche Randbedingung verletzt, einer anderen Lösung, die im Konflikt mit vielen solcher Regeln steht, gleichgestellt.

Insgesamt soll also aus den aufgeführten Gründen für den Rest dieser Arbeit nicht weiter zwischen harten und weichen Randbedingungen unterschieden werden.

2.4 Fazit

Die in diesem Kapitel definierten Begriffe dienen dazu, in den nächsten Kapiteln eine strukturierte Diskussion von Zuordnungsproblemen und den Oberflächen zu ihrer interaktiven Lösung zu ermöglichen. Insbesondere die Typisierung der Randbedingungen sollte den Zugang zu diesem sehr komplexen Begriff erleichtern und auch in Hinblick auf die Darstellung hilfreich sein; schließlich liegt die Vermutung nahe, daß unterschiedliche Typen von Regeln auch unterschiedlich visualisiert werden sollten.

3 Anforderungen an eine Oberfläche zur interaktiven Lösung von Zuordnungsproblemen

Die Anforderungen an ein Benutzerinterface lassen sich nur dann vollständig formulieren, wenn der konkrete Zweck, zu dem es eingesetzt werden soll, feststeht. Dennoch kann man ausgehend von den Grundprinzipien des Software Engineering, des Interface Designs und der Definition eines Zuordnungsproblems einen allgemeinen Anforderungskatalog erstellen.

3.1 Allgemeine Anforderungen

3.1.1 Allgemeine Anforderungen an Software

In der Literatur ([Shneiderman 98], [Sommerville 96] und [Stary 96]) sind folgende allgemeine Anforderungen an Software etabliert, denen natürlich auch ein Benutzerinterface, bzw. ein Programm zur Generierung von Oberflächen genügen sollte:

Anforderung	Erläuterung	Zum Beispiel erreichbar durch
Adaptierbarkeit	Programm muß an evolvierende Anforderungen anzupassen sein	OOP
Adäquate Funktionalität	Programm muß die nötige Funktionalität haben, die gestellte Aufgabe zu bewältigen	Rapid Prototyping
Benutzbarkeit	Benutzerinterface soll angemessen sein	Siehe 3.1.2
Durchschaubarkeit	Programm sollte von einem Fachmann nach kurzer Einarbeitungszeit verstanden werden können	Klare, logische Programmstruktur; kein monolithischer Aufbau
Leistung	Programm muß seine Aufgaben schnell und effektiv durchführen	Effiziente Algorithmen
On-Budget	Programm muß zu den vorgesehenen Kosten fertiggestellt werden	Verwendung von CASE-tools, erfahrene Programmierer
On-Time	Programm muß zum festgelegten Termin fertiggestellt werden.	Verwendung von CASE-tools, erfahrene Programmierer
Portabilität	Programm soll unabhängig von der jeweiligen Plattform sein	Programmierung in Java
Robustheit	Programm soll Fehler des Benutzers abfangen und behandeln	Prüfung des Formats und der Plausibilität von Eingabewerten

Schutz vor Informationsverlust	Kein unabsichtliches Löschen von Daten, reversible Kommandos	Bestätigungsaufforderung bei destruktiven Aktionen, Undo
Sicherheit, Datenschutz	Kein unautorisierter Zugriff auf Daten	Benutzung von Paßwörtern, Verschlüsselung von Daten
Verlässlichkeit	Programm muß seine Aufgaben korrekt durchführen	Keine Programmfehler
Wiederverwendbarkeit	Teile des Programmes sollten in anderer Software wiederverwendet werden können	Erstellen von Bibliotheken, Abstraktion, Modularität, OOP

Tabelle 1: Designziele bei der Softwareerstellung

3.1.2 Allgemeine Anforderungen an ein Benutzerinterface

Während die Ziele aus Tabelle 1 generell für das Design von Software formuliert sind, hat unter anderen [Shneiderman 98] weitere Ansprüche speziell für das Design von Benutzeroberflächen erarbeitet. Allerdings empfiehlt es sich hierbei, gewisse Vorsicht walten zu lassen; so schreibt etwa [Maybury & Wahlster 98] über „the nature of human-computer interaction“: „it is only partially understood and requires the multidisciplinary design talents of experts in human factors, computer science, device technology and media design“. Eine Betrachtung allein aus dem Blickwinkel der Informatik würde der Problematik also nicht gerecht werden. Es müssen zum Beispiel psychologische Faktoren berücksichtigt werden, die sich eines formalen Beweises entziehen und die in Einzelfällen von verschiedenen Experten auch durchaus unterschiedlich bewertet werden können. So erwähnt [Stary 96], daß sich Dean für die Verwendung persönlicher Redewendungen im Rahmen von Feedbackmechanismen ausspricht, Morland gegen deren Benutzung argumentiert.

Trotz dieser Vorbehalte formuliert [Shneiderman 98] folgende zentralen Qualitätsmerkmale oder Grundziele guter Benutzerschnittstellen:

1. Geringe Einarbeitungszeit: Ein Anwender soll die Oberfläche nach kurzer Einarbeitungszeit benutzen können und auch die erweiterten Funktionen schnell einzusetzen lernen.
2. Hohe Bedienungstempo: Ein Anwender soll seine Aufgaben schnell durchführen können.
3. Geringe Fehlerrate: Ein Anwender soll bei der Benutzung der Oberfläche möglichst wenige Fehler machen (können).
4. Subjektive Zufriedenheit des Benutzers: Der Anwender soll mit der Oberfläche zufrieden sein und sie subjektiv als gelungen einschätzen.
5. Mittel- bis langfristige Beibehaltung des Wissens: Ein Anwender soll sein Wissen, wie die Oberfläche benutzt wird, möglichst lange möglichst weitgehend behalten.

Shneiderman's Kriterien für Benutzerschnittstellen finden in der Literatur häufig Anwendung (z.B. [Vanderdonck 95]) und sind akzeptiert; vor allem lassen sie sich effektiv evaluieren und sind unabhängig vom Verwendungszweck der

Benutzerschnittstelle. Somit ist die Zielstellung klar, offen ist nur, wie diese zu erreichen sind, vor allem im Rahmen einer allgemeinen Diskussion, bei der über die Zielgruppe und den speziellen Verwendungszweck nichts bekannt ist. Dennoch lassen sich hier konkrete Maßnahmen („Goldene Regeln“) aufführen, die nach [Shneiderman 98] generell zur Realisierung obiger Ziele beitragen:

Maßnahme	Ziele	Erläuterung	Beispiele für Umsetzung
Feedback	1, 3, 4	Benutzer wird ständig mittels verbaler und graphischer Meldungen über den Systemzustand informiert	Jede Aktion des Anwenders wird vom System kommentiert; Erfolg, Wirkung, Dauer, aktueller Zustand, Grund für Mißerfolg und Abschluß wird dargestellt; Visualisierung der durch einen Befehl hervorgerufenen Veränderung
Fehlervermeidung und Fehlerbehandlung	1, 3, 4	Benutzer soll kaum Fehler machen können; Fehler sollten abgefangen und Korrekturmöglichkeiten angeboten werden	Sorgfältige, klare Systemarchitektur; Konsistenz; präzise, konstruktive, positiv formulierte und benutzerzentrierte Fehlermeldungen
Hilfe	1, 4	Hilfsfunktion	Für alle Situationen steht eine ausführliche, kontextsensitive Hilfe zur Verfügung; Suchfunktion innerhalb der Hilfe; Tutorials
Individualisierung	2, 4, 5	das System soll an die konkrete Aufgabe, den Arbeitsplatz, die physischen Fähigkeiten und Präferenzen des Benutzers anzupassen sein	Adaptierbarkeit der Informationskodierung (z.B. Schrifttyp und -größe, Farbgebung, Formatierung von Ein-, Ausgabe und Darstellung von Daten einstellbar); Makros; Einstellungen können gespeichert werden (Benutzerprofile)
Intelligente Dateneingabe	2, 3, 4	Dateneingabe soll so komfortabel wie möglich sein; Redundanz soll vermieden werden	Defaultwerte; Möglichkeit, Werte für Gruppen von Eingabefeldern einzugeben; Übernahme bekannter Daten innerhalb des Systems; Kompletierungsfunktion bei Dateneingabe mittels Tatstatur
Konsistente Darstellung	1, 3, 4, 5	Darstellung und Bezeichnung von Daten und Kommandos soll einheitlich sein	Einheitliche Terminologie, Formatierung, Abkürzungen, Symbolik, graphische Darstellung; Darstellung von Daten soll der ursprünglichen Eingabe des Benutzers möglichst ähnlich sein
Konsistentes Verhalten	1, 3, 4, 5	Verhalten aller Programmteile soll einheitlich sein	In allen Teilen des Programmes stehen die selben Befehle zur Verfügung und haben den selben Effekt; ähnliche Situationen erfordern und ermöglichen ähnliche Vorgehensweise
Kontrolle beim Anwender	4	Benutzer soll das Gefühl haben, daß er die Software kontrolliert und nicht umgekehrt	Kontrolle über Start, Dauer, Pause, Ausmaß, Wiederaufnahme und Beendigung von Aktionen liegt beim Anwender; keine überraschenden, unvorhersehbaren (Re)Aktionen der Applikation

Minimale Belastung des Kurzzeitgedächtnis	2, 3, 4	Die begrenzte Kapazität des Kurzzeitgedächtnis soll nicht überschritten werden	Minimale Anzahl von Schritten zur Erreichung eines Ziels; wenige Abkürzungen, Mnemoniks und Codes, wenn, dann nur nach ausreichender Schulung und mit erklärender Hilfefunktion
Modellierung des Systems nach Welt des Anwenders	1, 3, 4, 5	Das System soll der gewohnten Umgebung des Benutzers nachempfunden sein	Terminologie, Datenformatierung und graphische Repräsentation, Vorgehensweise so wie es der Benutzer im Aufgabenkontext gewohnt ist;
Reversibilität	3, 4	Kommandos sollten rückgängig zu machen sein	Undo-Funktion; Auto-Save in regelmäßigen Abständen; explizite Speicherfunktion
Shortcuts	2, 4	Abkürzungen für gängige (Sequenzen von) Befehle(n)	Vorgegebene Shortcuts, Funktionalität für benutzerdefinierte Shortcuts; Makros
Suche		Suchfunktionalität	
Tastaturunterstützung	2, 4	Programm kann – neben anderen Eingabemitteln – auf jeden Fall auch mit der Tastatur bedient werden	Navigation mittels Tabulator-, Pfeil-, und anderer Tasten mit Navigationsfunktionalität; Dateneingabe immer mit Tastatur möglich, Shortcuts über die Tastatur aufrufbar

Tabelle 2 : Goldene Regeln des Interface-Designs

3.1.3 Fazit

Die allgemeinen Anforderungen an Software sollen als Evaluationskriterien für die im Rahmen dieser Arbeit vorgeschlagene Lösung (Kapitel 5) dienen.

In Bezug auf Shneiderman's Grundziele gilt, daß sich natürlich auch eine Oberfläche zur interaktiven Lösung von Zuordnungsproblemen an ihnen orientieren sollte. Deswegen sollten die Grundziele auch in den nächsten beiden Abschnitten bei der Diskussion der Darstellung des Zuordnungsproblems und der Interaktivität zur Lösung des Zuordnungsproblems im Vordergrund stehen; konkret sollten also, wo immer möglich, die goldenen Regeln zur Umsetzung der Grundziele zum Einsatz kommen. Stellt sich zum Beispiel die Frage, wie eine Zuordnung visualisiert werden sollte, dann sollte gemäß der Regel „Modellierung nach der Welt des Anwenders“ die traditionelle Darstellung, die ein Benutzer gewohnt ist, zur Darstellung der Zuordnung im Computer gewählt werden. Jede in den nächsten Abschnitten getroffene Entscheidung bezüglich der konkreten Anforderungen an die Benutzerschnittstelle sollte also auf ihre Konformität mit den Grundzielen, bzw. den goldenen Regeln überprüft werden.

3.2 Anforderungen an die Darstellung des Zuordnungsproblems

Die Hauptaufgabe einer Oberfläche eines jeden Zuordnungswerkzeugs ist es, die Zuordnung darzustellen, also – gemäß der Definition eines Zuordnungsproblems – darzustellen, welche Angebotsobjekte welchen Nachfrageobjekten zugeteilt sind (die Zuordnung) und welche Randbedingungen durch diese Zuordnung verletzt sind.

Darüber hinaus existiert eine Reihe allen Zuordnungsproblemen gemeinsamer peripherer Ansprüche wie die Verwaltung von Angebots- und Nachfragemengen, die Visualisierung von Statistiken und eine Anzeige der Zahl der Nachfrageobjekte pro Angebotsobjekt als Auslastungs- oder Belastungsanzeige, neben den zusätzlichen individuellen Ansprüchen der einzelnen Zuordnungsprobleme.

Allerdings sollen im Rahmen dieser Arbeit nur die oben genannten zentralen Ansprüche einer genauen Analyse unterzogen werden, zum einen da die Untersuchung sekundärer Aspekte den Rahmen dieser Arbeit sprengen würde, zum anderen weil die Annahme naheliegt, daß sich in eben diesen Kerngebieten eher Gemeinsamkeiten zwischen den Ansprüchen verschiedener Zuordnungsprobleme finden lassen und somit auf eine überschaubare Anzahl von Grundtypen reduzieren lassen.

3.2.1 Darstellung der Zuordnung

Prinzipiell sollte eine Zuordnung so dargestellt werden, wie es die menschlichen Benutzer gewohnt sind (Tabelle 2, Modellierung nach Welt des Anwenders) – im Fall der Schulstundenplanung sollte sie einer Wandtafel (siehe Kapitel 4), wie sie in fast allen Schulen zur Planung von Hand zum Einsatz kommt, nachempfunden sein. In der Dienstplanung im Krankenhaus sollte sie sich an den zur Planung von Hand benutzten Formularen orientieren. Allerdings kann unter anderem wegen der begrenzten Größe der Darstellungsmedien (im allgemeinen ein Monitor) nicht jede Darstellungsweise exakt auf den Computer übertragen werden; auch bietet der Computer Darstellungsmöglichkeiten, deren Verwendung bei der Planung von Hand nicht praktikabel ist – etwa Blinkzeichen oder aussagekräftige, ständig aktuelle und mehrfarbige Diagramme. Aus diesen Gründen ist eine ausführliche Betrachtung und Bewertung der verschiedenen Darstellungsmöglichkeiten durchaus angebracht.

Die Forderung, sich bei der Visualisierung der Zuordnung an der traditionellen Darstellungsweise zu orientieren, läßt sich prinzipiell nur schwer mit dem Ziel vereinbaren, Oberflächen automatisch zu generieren, es sei denn, daß sich all diese in der Praxis vorkommenden Darstellungsarten konsequent aus einer Menge von logischen Regeln ableiten ließe. Dies ist natürlich nicht der Fall, allerdings ähneln sich diese Darstellungen laut empirischen Untersuchungen (siehe [Busch 97] und auch Kapitel 4) und können deshalb, wie bereits im Falle der Randbedingungen geschehen, auf einige wenige abstrakte Grundtypen zurückgeführt werden.

3.2.1.1 Grundtypen der Zuordnungsdarstellung

Listendarstellung

Im einfachsten Fall lässt sich eine 1-zu-n Zuordnung wie in Abbildung 3 als Liste von Paaren von Angebots- und Nachfrageobjekten darstellen. Diese Darstellung ist sehr einfach zu realisieren, aber oft zu unübersichtlich und kommt deshalb in der Praxis im allgemeinen nicht zum Zuge.

Nachfrageobjekt	Angebotsobjekt
n_1	a_{347}
n_2	a_7
...	...
n_k	a_{l-2}

Abbildung 3: Liste von Zuteilungen

Komprimierte Listendarstellung

Da die Listendarstellung in einer langen, aber schmalen Liste resultiert, bietet es sich an, auch die Breite eines Darstellungsmediums zu nutzen. Es wäre möglich, mehrere Abschnitte der Liste nebeneinander darzustellen, dann wären aber Angebots- und Nachfrageobjekte bunt gemischt, und das Resultat wäre sehr unübersichtlich. Lassen sich die Nachfrageobjekte dagegen in gleichmäßig strukturierte Gruppen einteilen, so bietet sich eine komprimierte Listendarstellung wie in Abbildung 4 an. Allerdings muß dann eindeutig zu erkennen sein, welche Objekte zusammen eine Gruppe bilden und auf welches Element einer Gruppe sich die Zuteilung eines Angebotsobjekts bezieht. Der Dienstplan für das ärztliche Personal (siehe Kapitel 4) ist ein Beispiel für eine solche komprimierte Listendarstellung.

	$G_1=\{n_1, \dots, n_{10}\}$	$G_2=\{n_{11}, \dots, n_{20}\}$	...	$G_q=\{n_{r*10+1}, \dots, n_k\}$
$n_{x*10+1} (x? \mathbb{N})$	a_{17}	a_{191}	...	a_{123}
$n_{x*10+2} (x? \mathbb{N})$	a_7	a_{191}	...	a_5
...	...	...	...	...
$n_{x*10+10} (x? \mathbb{N})$	a_l	a_2	...	a_{701}

Abbildung 4: Komprimierte Listendarstellung

Funktionsdarstellung

Nach Kapitel 2 kann eine Zuordnung auch als Abbildung der Nachfragemenge auf die Angebotsmenge interpretiert werden. Somit bietet sich die Darstellung einer 1-zu-n Zuordnung als zweidimensionaler Graph mit der Menge der Nachfrageobjekte auf der Abszisse und der Menge der Angebotsobjekte auf der Ordinate an (Abbildung 5). Ein Punkt oder allgemeiner ein Markierungsobjekt an der Schnittstelle einer Parallele zur Ordinate durch ein Nachfrageobjekt n_i und einer Parallele zur Abszisse durch ein Angebotsobjekt a_j bedeutet dann, daß a_j dem Objekt n_i zugeteilt ist, also x_{ij} nach Definition 1 den Wert 1 hat.

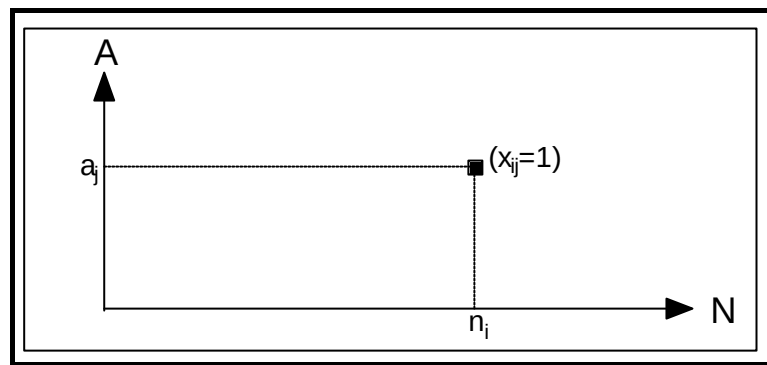


Abbildung 5: Prinzip der Funktionsdarstellung (nach [Busch 97])

Natürlich kann die Anordnung von Angebots- und Nachfragemenge beliebig variiert werden, also auch eine Darstellung mit den Angebotsobjekten auf der Ordinate, den Nachfrageobjekten auf der Abszisse und einer nach unten orientierten Abszisse gewählt werden, wie unter anderem in Abbildung 6 dargestellt.

Im Fall kontinuierlicher Angebotsmengen wie zum Beispiel Zeitintervallen eines Zeitstrahls kann bei der Funktionsdarstellung eine optional einblendbare Linie, wie in Abbildung 6 zu sehen, sehr hilfreich dabei sein, zugeteilte Angebotsobjekte zu identifizieren.

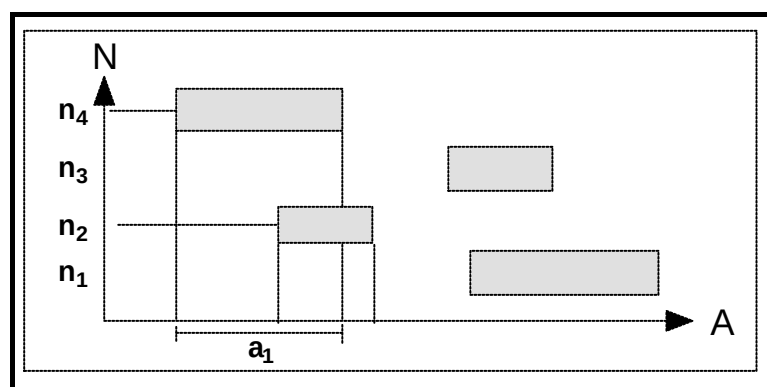


Abbildung 6: Funktionsdarstellung mit Hilfslinien (nach [Busch 97])

Tabellendarstellung

Bei diskreten Zuordnungen mit einer beschränkten Anzahl von Angebots- und Nachfrageobjekten ist es eventuell sinnvoll, über die Funktionsdarstellung wie in Abbildung 7 (mit dem Markierungssymbol x) ein permanentes Gitter zu legen, also eine Tabelle zu benutzen. Das zugrundeliegende Prinzip ist aber immer noch das der Funktionsdarstellung.

	a_1	a_2	...	a_l
n_1	x		...	
n_2			...	x
...	...	...	...	
n_k	x			

Abbildung 7: Darstellung als Tabelle (nach [Busch 97])

Gruppendarstellung

Die Benutzung von Tabellen wie in Abbildung 7 hat jedoch den gravierenden Nachteil, daß der vorhandene Platz nicht effektiv genutzt wird; besonders bei großen Angebots- oder Nachfragemengen ist die Darstellung wenig kompakt, fast alle Zellen der Tabelle bleiben leer – nur k der $k \cdot n$ Zellen haben einen Eintrag, da jedem der k Nachfrageobjekte nur ein Angebotsobjekt zugeordnet wird. Deshalb wird die Darstellung manchmal wie im Falle der Listendarstellung verdichtet, indem verschiedene Nachfrageobjekte zu Gruppen zusammengefaßt werden. Die genaue Information, welches der Nachfrageobjekte der Gruppe tatsächlich zugeteilt ist, muß dann zusätzlich visualisiert werden, in Abbildung 8 zum Beispiel über das Markierungsobjekt. Können den Objekten innerhalb einer Gruppe aufgrund der Randbedingungen keine identischen Angebotsobjekte zugeordnet werden, so bezeichnet man diese Gruppierung, bzw. die entsprechende Darstellung als injektiv⁹.

	$G_1=\{n_1, \dots, n_{10}\}$	$G_2=\{n_{11}, \dots, n_{20}\}$	...	$G_q=\{n_{r+10+1}, \dots, n_k\}$
a_1	n_1		...	n_k
a_2		n_{19}	...	
...	...	...	...	...
a_l	n_{10}	n_{12}	...	n_{r+10+7}

Abbildung 8: Tabelle mit injektiver Gruppierung

⁹ Da die Einschränkung der Zuordnung (interpretiert als Funktion) auf diese Gruppe dann eine injektive Funktion in die Menge der Angebotsobjekte darstellt.

Ein Beispiel für die injektive Gruppendarstellung ist die Dienstplanung im Krankenhaus mit den Pflegekräften als Angebotsobjekten, den Schichten als Nachfrageobjekten: Eine Pflegekraft wird immer nur einer Schicht eines Tages zugeteilt, niemals zwei oder mehr Schichten desselben Tages. Deshalb werden üblicherweise alle Schichten eines Tages zu einer Gruppe zusammengefaßt und dann mittels des Markierungsobjekts, im Beispielsfall üblicherweise der Anfangsbuchstaben des Schichttyps der Schicht, ausgedrückt, welcher dieser Schichten die Pflegekraft tatsächlich zugeteilt ist.

Verzichtet man auf die Forderung, nur solche Objekte zu Gruppen zusammenzufassen, denen nicht dasselbe Objekt zugeteilt werden kann, so müssen potentiell mehrere Zuteilungen in derselben Zelle visualisiert werden. Das ist zwar prinzipiell möglich, jedoch unter Umständen problematisch, da dann in jeder Zelle genügend Raum für die Darstellung mehrerer Markierungsobjekte vorhanden sein muß, wie in Abbildung 9 dargestellt ist.

	$G_1=\{n_1, \dots, n_{10}\}$	$G_2=\{n_{11}, \dots, n_{20}\}$	...	$G_q=\{n_{r*10+1}, \dots, n_k\}$
a_1			...	
a_2		$n_{11}, n_{13}, n_{15}, n_{16}, n_{17}, n_{18}, n_{19}$	...	n_{r*10+5}, n_{r*10+7}
...	...	...	...	...
a_i	n_{10}, n_7	n_{12}	...	n_k

Abbildung 9: Tabelle mit nicht-injektiver Gruppierung

Mehrfachzuordnung

Die Visualisierung von Mehrfachzuordnungen ist naturgemäß komplexer und weniger intuitiv – die Darstellung eines n -dimensionalen Graphen auf einem 2-dimensionalen Medium wie einen Bildschirm erscheint zu unübersichtlich. Üblicherweise werden in einem solchen Fall wieder über das Markierungsobjekt weitere Informationen vermittelt. Faßt man, wie in Kapitel 2.1 beschrieben, eine Mehrfachzuordnung als Komposition mehrerer Zuordnungen auf, so bedeutet das übertragen auf die Darstellung, daß die erste Zuordnung durch einen der bisher in diesem Abschnitt beschriebenen Grundtypen dargestellt werden kann, alle weiteren Zuordnungen dann aber über das Markierungsobjekt visualisiert werden müssen. Abbildung 10 zeigt die schematische Darstellung einer Zweifachzuordnung, wobei die Angebotsobjekte $(a_i)_1$ Elemente der Angebotsmenge A_1 , die Angebotsobjekte $(a_i)_2$ Elemente der Angebotsmenge A_2 sind. Dem Nachfrageobjekt n_i der Abbildung 10 sind somit die beiden Angebotsobjekte $(a_i)_1$ und $(a_i)_2$ zugeordnet.

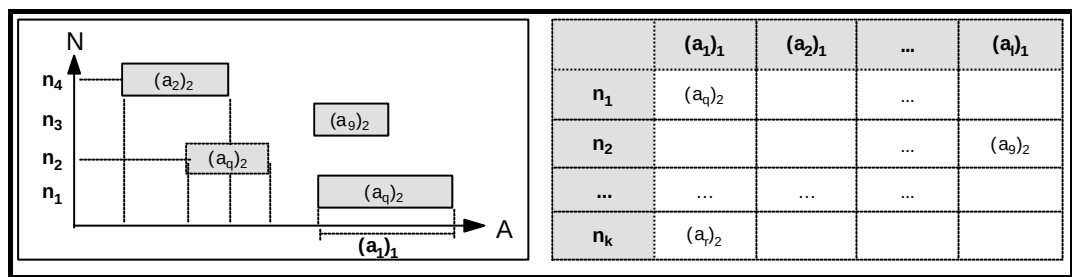


Abbildung 10: Darstellung einer Mehrfachzuordnung

Mehrfach- und Gruppendarstellung können natürlich auch kombiniert werden, was zum Beispiel bei der Schulstundenplanung (Kapitel 4) der Fall ist. Abbildung 11 zeigt die resultierende Darstellung:

	$G_1=\{n_1, \dots, n_{10}\}$	$G_2=\{n_{11}, \dots, n_{20}\}$	...	$G_q=\{n_{r*10+1}, \dots, n_k\}$
$(a_1)_1$	$n_1, (a_r)_2$		...	
$(a_2)_1$		$n_{19}, (a_{12})_2$	...	$n_{r*10+5}, (a_{412})_2$
...	...	...	...	...
$(a_i)_1$	$n_{10}, (a_{412})_2$		...	$n_{r*10+7}, (a_{999})_2$

Abbildung 11: Kombination von Mehrfach- und Gruppendarstellung

3.2.1.2 Kodierungsformen des Markierungsobjekts

Im letzten Abschnitt wurde deutlich, daß bei komplexeren Darstellungsweisen wie der Gruppendarstellung oder der Darstellung von Mehrfachzuordnungen dem Markierungsobjekt wesentliche Aufgaben wie die Identifikation eines Objekts der Angebotsmenge zukommen. Deswegen soll nun untersucht werden, welche prinzipiellen Möglichkeiten es gibt, Informationen über die Zuordnung mittels der Darstellung des Markierungsobjekts zu kodieren, und inwiefern diese Kodierungsformen für diese Aufgaben geeignet sind.

3.2.1.2.1 Beurteilungskriterien

Tabelle 3 nach [Herczeg 94] versucht, gängige Kodierungsmöglichkeiten in Kategorien einzuteilen und zu bewerten. Dabei gibt die Anzahl der Stufen an, wie viele Abstufungen derselben Kodierungsform simultan verwendet werden können – so sollten laut Herczeg höchstens 6 unterschiedliche Farben gleichzeitig verwendet werden. Unterscheidbarkeit zeigt an, wie gut sich die verschiedenen Abstufungen einer einzelnen Kodierungsform prinzipiell voneinander abheben. Nach Herczeg lassen sich also verschiedene Linienarten nur schwer, verschiedene Symbole dagegen sehr gut voneinander unterscheiden.

Kodierungsform	Einsatzgebiet	Stufen	Unterscheidbarkeit
Symbol	Buchstaben, Ziffern, Zeichen, Texte	Beliebig	++
Bildliche Form	Pictogramm / Icons	10	+
Position	Hilfe, Fehler, Datum, Überschrift	9	+
Winkel	Zeigerinstrumente	8	+
Farbton	Aufmerksamkeit erregen	6	+
Länge		6	+
Geometrische Form		5	+
Fläche		3	-
Schriftgröße		3	-
Linienart		3	-
Schraffierung		3	-
Schriftform		3	-
Blinkfrequenz	Aufmerksamkeit erregen	2	-
Helligkeit		2	-

Tabelle 3: Gängige Kodierungsformen und ihre Eigenschaften

Neben den von Herczeg bewerteten allgemeinen Gesichtspunkten spielen folgende zusätzliche Kriterien im Hinblick auf die konkrete Aufgabe – die Darstellung der Zuordnung – eine Rolle:

Platzbedarf

Zuordnungsprobleme sind oft sehr komplex, die Angebots- und Nachfragemengen entsprechend groß (siehe Kapitel 4), der gegebene Raum zur Darstellung des Problems auf dem Bildschirm ist jedoch starken Einschränkungen unterworfen. Dennoch sollte es das Ziel der Darstellung sein, einen möglichst großen Teil des Zuordnungsproblems gleichzeitig darstellen zu können; ansonsten müßte der Benutzer zwischen verschiedenen Darstellungen wechseln und zwischen ihnen sehr komplexe Informationen transferieren, was dem Prinzip der minimalen Belastung des Kurzzeitgedächtnis (Tabelle 2) widerspräche.

Effektive und intuitive Eingabe

Um direkte Manipulation zu ermöglichen (siehe dazu Kapitel 3.3), ist es wichtig, daß Stufen einer Kodierungsform leicht vom Benutzer spezifiziert werden können.

Ressourcenaufwand für die Ausgabe

Es muß möglich sein, die Kodierungsform ohne großen Ressourcenaufwand auch auf älteren Computern darzustellen.

Individualisierbarkeit

Der Benutzer soll in der Lage sein können, eigene Stufen der Kodierungsform zu kreieren oder zumindest aus einer großen Anzahl von Kodierungsstufen wählen können.

Skalierbarkeit

Die Kodierungsform sollte sich möglichst stufenlos in ihrer Größe verändern lassen, um sich unterschiedlichen räumlichen Rahmenbedingungen – etwa kleineren Darstellungsmedien oder Benutzern mit Sehbehinderungen – anpassen zu können.

3.2.1.2.2 Beurteilung der einzelnen Kodierungsformen

Symbole

Wie Tabelle 3 zeigt, sind Symbole prinzipiell sehr gut geeignet, Informationen zu vermitteln. Falls viele unterschiedliche Informationen zu vermitteln sind, sind sie sogar das einzige Mittel, das eine quasi unbeschränkte Anzahl von Abstufungen und somit an Inhalten vermitteln kann. Übertragen auf Zuordnungsprobleme führt diese Erkenntnis zu einer Darstellungsweise mit Symbolen als Markierungsobjekten wie in Abbildung 12 schematisiert dargestellt ist.

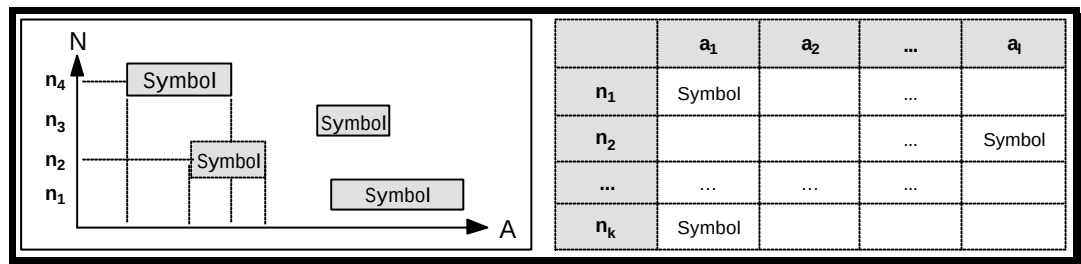


Abbildung 12: Symbole als Kodierungsform

Der Platzbedarf von Symbolen läßt sich nicht allgemein beurteilen – während etwa ein F als Symbol für eine Frühschicht bei der Dienstplanung im Krankenhaus nur einen äußerst geringen Platzbedarf hat, lassen sich manche Informationen nur aufwendig durch Symbole ausdrücken.

Die Dateneingabe für Symbole läßt sich effektiv und intuitiv über die Tastatur (deren Unterstützung nach Tabelle 2 eine Notwendigkeit ist) realisieren, da sich Symbole im Normalfall auf der Tastatur wiederfinden. Es gehört zur Standardfunktionalität eines Computers, eine Vielzahl von Symbolen (in allen Schriftgrößen) ohne großen Ressourcenaufwand darstellen zu können. Symbole sind sehr gut individualisierbar, da jedes moderne Betriebssystem über einen sehr großen Zeichensatz verfügt, der dann auch Anwendungsprogrammen zur Verfügung steht.

Bildliche und geometrische Formen

Gut zu unterscheiden sind bildliche Formen - Icons, bzw. allgemein Grafiken (Abbildung 13), allerdings ist es mit ihnen nicht möglich, eine unbeschränkte Anzahl von Informationen zu kodieren – nach Herczeg sollten maximal 10 verschiedene bildliche Formen verwendet werden. Sie können also nur in Situationen eingesetzt werden, in denen definitiv langfristig nur eine geringe Anzahl unterschiedlicher Inhalte kodiert werden muß. Neben einem potentiell höheren programmatischen Aufwand für die Darstellung von Graphiken etwa im Vergleich zur Darstellung von Symbolen ist auch die Auswahl der konkreten Icons und Bildinhalte problematisch – nach Tabelle 2 sollten sie aus dem gewohnten Kontext des Benutzers stammen. Auch Platzbedarf und Skalierbarkeit sind potentiell problematisch. Anders als für Symbole müssen für bildliche Formen indirekte, weniger intuitive Eingabeverfahren verwendet werden. Schließlich können bildliche Formen nicht so einfach vom Benutzer angepaßt werden, es sei denn,

eine große, strukturierte Bibliothek mit bildlichen Formen oder ein Graphikeditor stünden zur Verfügung.

All diese Überlegungen lassen sich auch auf geometrische Formen übertragen.

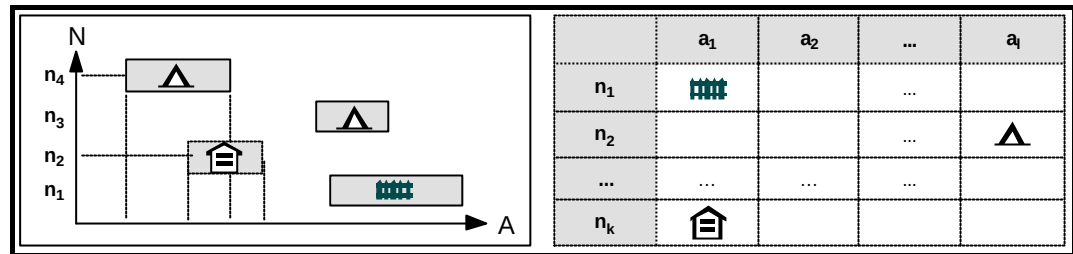


Abbildung 13: Icons als Kodierungsform

Position, Fläche und Länge

Position, Länge und Fläche können nur sehr beschränkt dazu verwendet werden, zusätzliche Information zu vermitteln, da diese Kodierungsformen schon für die Funktionsdarstellung verwendet werden.

Winkel

Das primäre Einsatzgebiet von Winkeln sind Zeigerinstrumente, deren Einsatz im Rahmen einer allgemeinen Analyse der Visualisierung von Zuordnungsproblemen nicht nahe liegt.

Farbe, Schraffierungen, Helligkeit und Blinken

Farbe, mit Einschränkungen Schraffierungen, Helligkeit und Blinken eignen sich prinzipiell zur effektiven Informationskodierung (Abbildung 14). Hauptmanko im Hinblick auf den Einsatzzweck ist in allen Fällen wieder die geringe Zahl der einsetzbaren Stufen; Platzbedarf und Darstellungsaufwand sind äußerst gering, Skalierbarkeit ist gegeben, Eingabe und Individualisierung sind im Falle von Farbe und Helligkeit über Standard-Auswahldialoge eines jeden modernen Betriebssystems realisierbar. Die Verwendung von Farbe und Helligkeit kann auf monochromen Darstellungsmedien problematisch sein, ebenfalls für die ca. 8% der Bevölkerung, die ganz oder teilweise farbenblind sind [Shneiderman 98].

Darüber hinaus ist laut Herczeg das primäre Einsatzgebiet für diese Darstellungsformen die Erregung von Aufmerksamkeit – deshalb ist es sinnvoll, wenn möglich diese Kodierungstechniken für eben diesen Zweck zu reservieren und nicht für die Visualisierung der Zuordnung selbst zu verwenden.

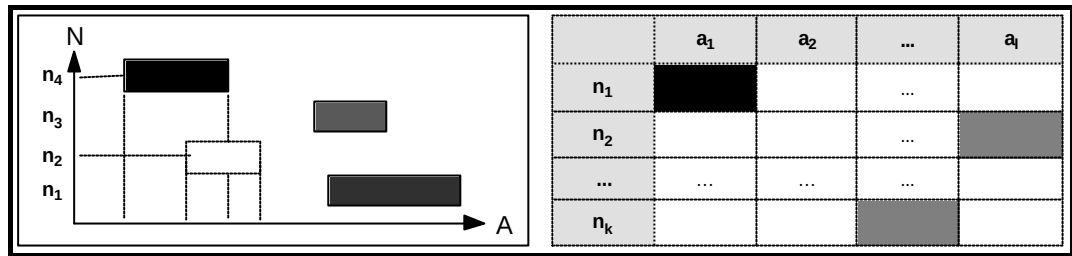


Abbildung 14: Farbe und Schraffierungen als Kodierungsform

Schriftgröße und Schriftart

Die weiteren in Tabelle 3 aufgeführten Kodierungsformen wie zum Beispiel Schriftart oder -größe (Abbildung 15) sind aufgrund der geringen Anzahl der gleichzeitig einsetzbaren Stufen (<4) und der mäßigen Unterscheidbarkeit prinzipiell wenig geeignet, eine Zuordnung darzustellen, auch wenn sie effektiv darzustellen sind und über die Standarddialoge eines jeden modernen Betriebssystems spezifiziert werden können. Außerdem bedürfen sie der Verwendung von Symbolen – nur dann können sie eingesetzt werden, sind also per se nicht zur Visualisierung von Zuordnungen geeignet. Sinnvoller scheint ihre Verwendung zur Vermittlung von Zusatzinformationen oder zur Erregung von Aufmerksamkeit, auch wenn sich nach Herczeg zu diesem Zweck Farben besser eignen.

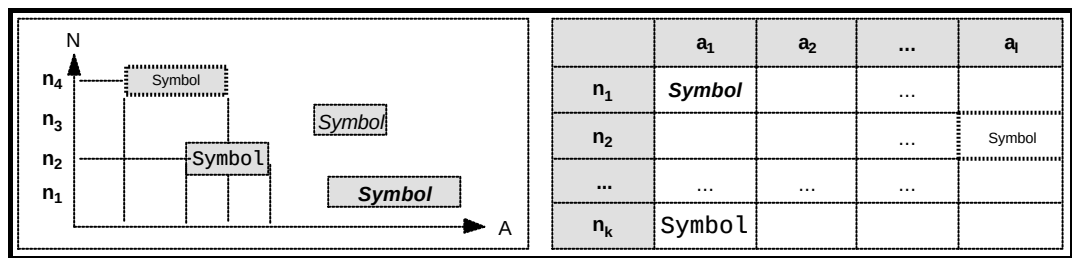


Abbildung 15: Schriftform als Kodierungsform

Linienart

Wieder gilt, daß die Linienart (Abbildung 16) aufgrund der geringen Anzahl der gleichzeitig einsetzbaren Stufen und der mäßigen Unterscheidbarkeit dieser Stufen im allgemeinen ungeeignet ist, die Zuordnung selbst darzustellen. Zusätzlich ist es die Konvention, den aktuellen Sitz des Tastaturfokus (siehe 3.3) durch eine stärkere Umrandung, also eine größere Linienbreite darzustellen, so daß auch eine Verwendung von Linienarten als Kodierungsform für andere Zwecke aus Konstistenzgründen verzichtet werden sollte.

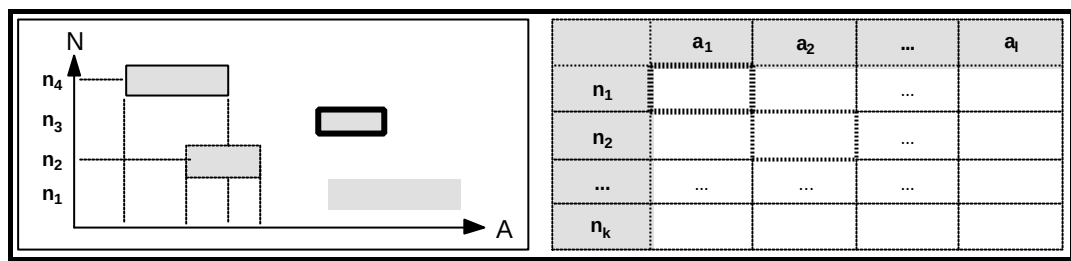


Abbildung 16: Linienart als Kodierungsform

3.2.1.3 Fazit

Der Anspruch an die Oberfläche eines Werkzeuges zur Lösung von Zuordnungsproblemen lautet, die beschriebenen Grundtypen der Zuordnungsdarstellung visualisieren zu können.

In Bezug auf das Markierungsobjekt bleibt als klares Fazit, daß generell Symbole das Mittel der Wahl repräsentieren, zusätzliche Informationen über dieses Objekt zu vermitteln. Bildliche Formen sind nur mit Einschränkungen für diesen Zweck geeignet; ihre Verwendung führt zu einer Vielzahl von Problemen. Die restlichen Mittel sind nur sehr bedingt zur Informationskodierung bezüglich der Zuordnung über das Markierungsobjekt anwendbar. Zwei besondere Eigenschaften von Symbolen unterstreichen diesen Eindruck noch weiter:

- ?? Ihre potentielle Generierbarkeit: es ist vorstellbar, Symbole als Markierungsobjekte mittels bestimmter Regeln automatisch zu generieren, so wird zum Beispiel bei der Dienstplanung im Krankenhaus meist jede Schicht mit den Anführungsbuchstaben ihres Schichttyps kodiert, also jede Frühschicht mit F, jede Spätschicht mit S, usw.
- ?? die traditionelle Rolle von Symbolen bei der Informationskodierung von Hand (z.B. als Schrift), d.h. die Vertrautheit menschlicher Benutzer mit dieser Form der Informationskodierung.

Es ist jedoch durchaus sinnvoll, die restlichen Kodierungsformen – wenn auch nicht als primäre Kodierungsform bezüglich der Zuordnung selbst – zumindest zur Unterstützung der Informationskodierung einzusetzen, also etwa farbige Symbole als Markierungsobjekte zu verwenden. Allerdings obliegt diese Entscheidung nur bedingt dem Designer, der sich, wie bereits erwähnt, an der traditionellen Darstellung des Zuordnungsproblems orientieren sollte, also auch bei der Informationskodierung über das Markierungsobjekt. Die Anforderung an die Oberfläche lautet also allein aus diesem

Grund, das Markierungsobjekt mittels aller in diesem Abschnitt beschriebenen Kodierungsformen darstellen können.

3.2.2 Visualisierung verletzter Randbedingungen

Ein Benutzer sollte aus der Darstellung der Zuordnung nicht nur ersehen können, welches Angebotsobjekt welchem Nachfrageobjekt zugeordnet ist, sondern auch Aufschluß über die Qualität der Lösung des Zuordnungsproblems erhalten, also sehen können, welche Randbedingungen durch die Zuordnung verletzt sind. Ziel der Zuordnungsdarstellung ist es, möglichst die Darstellungsweise nachzubilden, die der Benutzer für das Zuordnungsproblem gewohnt ist; diese Restriktion entfällt im allgemeinen bei der Visualisierung von Regelverletzungen, da letztere ein Vorzug der computergestützten Planung ist und bei der Planung von Hand üblicherweise nicht vorkommt. Somit sind der Kreativität des Designers keine so engen Grenzen gesetzt wie im Falle der Zuordnungsdarstellung. Prinzipiell sollte der Benutzer darüber entscheiden können, ob und wie Verletzungen von Randbedingungen dargestellt werden (Tabelle 2: Sitz der Kontrolle beim Anwender, Individualisierung). Die Oberfläche muß jedoch intuitiv verständliche Standarddarstellungen offerieren (Grundziel der leichten Erlernbarkeit), die der Benutzer dann nach Bedarf verändern kann. Somit sollen in diesem Kapitel erst die möglichen Visualisierungen von Restriktionsverletzungen erarbeitet werden; diese unterscheiden sich vor allem in folgenden grundsätzlichen Aspekten:

- ?? Relevante Informationen: Welche Informationen sind von Belang und sollten dargestellt werden?
- ?? Mögliche Orte der Visualisierung: Wo sollten diese Informationen dargestellt werden?
- ?? Mögliche Kodierungsformen: Wie sollten diese Informationen graphisch kodiert werden?

Nach den möglichen Antworten auf diese grundsätzlichen Fragen sollen zunächst eventuelle spezielle Aspekte einzelner Restriktionstypen und besondere Situationen, die bei der Visualisierung verletzter Randbedingungen auftreten können, untersucht werden. Anschließend sollen die Standarddarstellung von Restriktionsverletzungen festgelegt werden und die Möglichkeiten für Designer oder Benutzer vorgestellt werden, diese ihren Ansprüchen anzupassen.

3.2.2.1 Relevante Informationen

In der Praxis sind im Falle einer verletzten Randbedingung vor allem folgende Informationen von Interesse und sollten deshalb so weit möglich visualisiert werden:

- ?? Randbedingung: Welche Randbedingung ist verletzt? (z.B. Pflegeplanung: Abstand zweier Schichten einer Pflegekraft muß mindestens 10 Stunden betragen)
- ?? Grund: Warum ist die Randbedingung verletzt? (Abstand beträgt nur 8 Stunden)
- ?? Verursacher: Welche Zuteilungen sind an dieser Verletzung beteiligt? (Pflegekraft A, Spätschicht am 3.4 und Pflegekraft A, Frühschicht am 4.4)
- ?? Bewertung: Wie schwerwiegend ist die Verletzung? (Je nach verwendeter Bewertungsmetrik: z.B. Verstoß gegen wichtige Dienstvorschrift; Schwerer Regelverstoß; 108.3)

3.2.2.2 Mögliche Orte der Visualisierung

Es gibt zwei prinzipiell verschiedene Stellen, an denen Informationen über eine verletzte Randbedingung visualisiert werden können – zum einen kann dies mittels einer von der Darstellung der Zuordnung separierten, im folgenden als extern bezeichneten Anzeige erfolgen, zum anderen kann versucht werden, diese Informationen in die Darstellung der Zuordnung zu integrieren (interne oder integrierte Visualisierung).

3.2.2.2.1 Externe Visualisierung

Eine externe Visualisierung von Regelverstößen, zum Beispiel in einem eigenem Fenster, ist unabhängig von der Darstellung der Zuordnung. Ein gravierender Nachteil der externen Visualisierung ist, daß der Benutzer seine Aufmerksamkeit gleichzeitig auf zwei Darstellungen richten und dabei relativ komplexe Informationen zwischen ihnen transferieren muß, was dem Prinzip der minimalen Belastung des Kurzzeitgedächtnis widerspricht (Tabelle 2). Deshalb liegt es nahe zu versuchen, die Darstellung der Regelverstöße in die Darstellung der Zuordnung zu integrieren. Siehe Abbildung 26 für ein Beispiel für die externe Visualisierung.

3.2.2.2.2 Interne Visualisierung

Entscheidender Vorteil der internen Visualisierung ist, daß die per Definition wichtigsten Informationen eines Zuordnungsproblems in eine einzige Darstellung integriert sind. Allerdings stellt sich die Frage, wo innerhalb der Darstellung der Zuordnung ein Regelverstoß angezeigt werden sollte. Wie in Kapitel 2.3 erläutert, werden Randbedingungen über einzelne Zuteilungen, Zuteilungskombinationen und über Attribute an diesen Zuteilungen beteiligter Objekte formuliert. Somit sind die Punkte der Zuordnungsdarstellungen, die mit eben diesen Zuteilungen, Attributen und Objekten assoziiert sind, auch als Stellen der integrierten Visualisierung der betroffenen Randbedingung prädestiniert, wobei für eine einzelne Restriktionsverletzung durchaus mehrere Stellen gleichzeitig gewählt werden können.

Zuteilungen als Stellen der Visualisierung

Im Falle der Zuteilung(en) sind das üblicherweise die entsprechenden Markierungsobjekte, oder die Stellen, an denen die Markierungsobjekte eigentlich stehen sollten (im Falle der Verletzungen von Regeln, die Zuteilungen durchsetzen sollen).

Attribute als Stellen der Visualisierung

Zusätzlich bieten sich genau die Attribute der an dem Regelverstoß beteiligten Objekte an, die auch zur Formulierung der verletzten Randbedingung benutzt wurden. Zum Beispiel schreibt eine Regel der Dienstplanung im Krankenhaus vor, daß Pflegekräfte im Planungszeitraum eine bestimmte Zahl von Stunden arbeiten sollten, nicht mehr und nicht weniger, sie bezieht sich also auf die Attribute „Ist-Stunden“ und „Soll-Stunden“ einer Pflegekraft. Arbeitet die Pflegekraft A jedoch deutlich mehr und liegt deshalb ein Verstoß gegen diese Regel vor, so kann man einerseits alle Zuteilungen dieser Pflegekraft als Visualisierungsstellen auswählen, schließlich tragen sie alle zu dem Regelverstoß bei.

Andererseits könnte man auch die Attribute „Ist-Stunden“ und „Soll-Stunden“ der Pflegekraft A auswählen – falls sie in die Zuordnungsdarstellung integriert sind –, und somit zusätzlich die Begründung für die Restriktionsverletzung zumindest andeuten.

Objekte als Stellen der Visualisierung

Schließlich bleiben noch die an dem Regelverstoß beteiligten Objekte; verstößt die Zuteilung a_i zu n_j gegen eine Randbedingung, so kann man die entsprechende Verletzung an den Stellen darstellen, die mit eben diesen beiden Objekten identifiziert werden.

Aus diesen Überlegungen folgt, daß es sinnvoll sein kann, nicht nur Zuteilungen und Objekte, sondern auch Objektattribute in die Zuordnungsdarstellung zu integrieren, ein Aspekt, der insbesondere bei der interaktiven Planung an Bedeutung gewinnt (siehe 3.3) und der in Abbildung 17 schematisiert dargestellt wird. Dabei stehen schraffierte Zellen für Bereiche, in denen das Attribut nicht definiert ist.

	Att ₁	...	n ₁	n ₂	n ₃	n ₄	...	n ₇	Att _s (a)
Att _{s+1}			Att _{s+1} (n ₁)	Att _{s+1} (n ₂)	Att _{s+1} (n ₃)	Att _{s+1} (n ₄)		Att _{s+1} (n ₇)	
a ₁	Att ₁ (a ₁)		x						Att _s (a ₁)
...			...	...	...	...	...	...	...
a _k	Att ₁ (a _k)				x			x	Att _s (a _k)

Abbildung 17: Darstellung von Attributen im Rahmen der Zuordnungsdarstellung

Attribute lassen sich sogar im Rahmen der Gruppendarstellung einsetzen; allerdings muß dann zusätzlich visualisiert oder zumindest eine Konvention bezüglich der Frage gefunden werden, welches Element der Gruppe mit welchem Attributwert assoziiert wird oder ob das Attribut eventuell für die ganze Gruppe gilt. Abbildung 18 zeigt die resultierende Darstellung. Das Formular der Dienstplanung für das Pflegepersonal des Juliusspitals (siehe Kapitel 4) ist ein Beispiel für eine solche Darstellung.

	G ₁ ={n ₁ , ..., n ₁₀ }	G ₂ ={n ₁₁ , ..., n ₂₀ }	...	G _i ={n _{i*10+1} , ..., n _{i*10+10} }
a ₁	n ₁		...	
a ₂	n ₇	n ₁₉	...	n _{i+5}
...	...	...	...	
a _k	n ₁₀	n ₁₂	...	n _{i+7}
Att ₁	Att ₁ (n ₁)	Att ₁ (n ₁₁)	...	Att ₁ (n _{i*10+1})
...	...	...	...	...
Att ₁	Att ₁ (n ₁₀)	Att ₁ (n ₂₀)	...	Att ₁ (n _{i*10+10})

Abbildung 18: Visualisierung von Attributen bei der Gruppendarstellung von Zuordnungen

Abbildung 19 faßt die möglichen Stellen, an denen Regelverletzungen im Rahmen der internen Darstellung dargestellt werden können, zusammen.

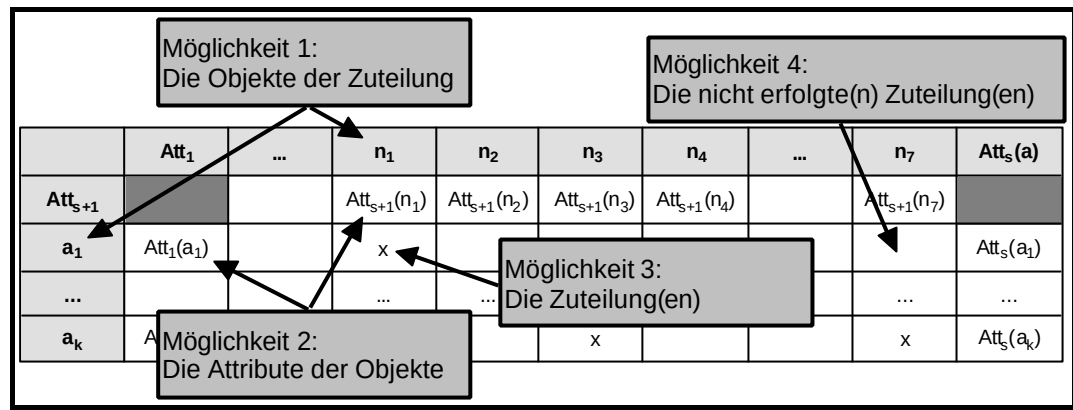


Abbildung 19: Mögliche Stellen zur internen Visualisierung von Regelverstößen

3.2.2.3 Mögliche Kodierungsformen

Prinzipiell stehen für die Kodierung von Restriktionsverletzungen wieder alle Kodierungsformen aus Tabelle 3 zur Verfügung, im Falle der externen Visualisierung auch uneingeschränkt, sowohl in Bezug auf den verfügbaren Platz als auch auf die Kodierungsformen. Deshalb kann bei der externen Visualisierung Text (also Symbole) verwendet werden, also das im allgemeinen geeignetste Mittel, komplexe Sachverhalte effektiv, präzise und verständlich zu vermitteln.

Die Kodierung im Falle der integrierten Visualisierung erweist sich als komplizierter, da hier eine Vielzahl von Kriterien zu berücksichtigen ist. Mangelnder Platz macht eine ausführliche Verbalisierung eines jeden einzelnen Regelverstößen wie im Falle der externen Visualisierung im allgemeinen unmöglich – da Überschneidungen zu vermeiden sind, müßte man entweder eine sehr kleine und damit unlesbare Schriftgröße wählen, oder man könnte nur wenige Restriktionsverletzungen gleichzeitig darstellen, was wiederum der ursprünglichen Motivation zur integrierten Visualisierung widerspräche, möglichst viel Information simultan sichtbar zu machen. Deshalb sollen nun alle in Tabelle 3 aufgeführten Kodierungsformen auf ihre Eignung zur integrierten Visualisierung von Restriktionsverletzungen überprüft werden.

3.2.2.3.1 Beurteilungskriterien

Zusätzlich zu den bereits erläuterten Kriterien Platzbedarf, Ressourcenaufwand für die Ausgabe, Individualisierbarkeit und Skalierbarkeit müssen bei der integrierten Visualisierung von Regelverstößen zwei weitere Kriterien berücksichtigt werden:

Kombinierbarkeit

Die Visualisierung von Restriktionsverletzungen findet meistens an Stellen der Zuordnungsdarstellung statt, die bereits Informationen kodieren und sollte diese primären Informationen nicht überschreiben, bzw. verdecken oder schwerer erkennbar machen, sondern sich – im optimalen Fall – ohne zusätzlichen Platzbedarf in die bereits vorhandene Darstellung integrieren lassen. Dabei ist die Kombinierbarkeit mit den bevorzugten Kodierungsformen der Zuordnungsdarstellung (Symbole, bildliche und geometrische Formen, mit Einschränkungen Farbe) von besonderem Interesse.

Eignung zur Erregung von Aufmerksamkeit

Die Visualisierung von Regelverstößen soll diese nicht einfach nur darstellen, sondern die Aufmerksamkeit des Benutzers auf diese Problemstelle n innerhalb der Zuordnung lenken.

3.2.2.3.2 Beurteilung der einzelnen Kodierungsformen

Symbole

Symbole lassen sich nicht ohne zusätzlichen Platzbedarf mit anderen Symbolen und im allgemeinen nicht mit bildlichen oder geometrischen Formen, dafür gewöhnlich gut mit Farben kombinieren. Sie sind nur bedingt dazu geeignet, Aufmerksamkeit zu erregen.

Bildliche und geometrische Formen

Bildliche (und geometrische) Formen sind zur Erregung von Aufmerksamkeit geeignet, z.B. in der Form von Fahnen oder Stopzeichen, lassen sich jedoch im allgemeinen nicht ohne zusätzlichen Platzbedarf mit anderen bildlichen Formen oder Symbolen, jedoch bedingt mit Farben kombinieren.

Farbe, Schraffierung und Blinken

Laut Tabelle 3 nach Herczeg ist Farbe das beste graphische Mittel, um Aufmerksamkeit zu erregen, auch [Shneiderman 98] betont die Eignung von Farbe für diesen Zweck: „Color can ... draw attention to warnings“. Zugleich läßt sich Farbe meist sehr gut ohne zusätzlichen Platzbedarf mit anderen Kodierungsformen kombinieren, vor allem mit Symbolen, wobei jedoch manche Farbkombinationen wie zum Beispiel Blau und Rot oder Magenta und Grün nach [Shneiderman 98] problematisch sind. Falls eine Tabelle für die Darstellung einer Zuordnung gewählt wurde, läßt sich auch die Hintergrundfarbe einer Zelle farbig markieren und das eigentliche Kodierungssymbol für die Zuteilung kann unverändert gelassen werden.

Die Wahl der geeigneten Farbe kann nur im konkreten Problemkontext geschehen, nicht im Rahmen einer allgemeinen Diskussion, da je nach Kontext unterschiedliche und sich teilweise widersprechende Konventionen über die Bedeutung bestimmter Farben existieren. Die Farbe Rot indiziert nach [Shneiderman 98] im allgemeinen „Halt!“ oder eine Warnung, im Zusammenhang mit Finanzen Verluste, hat jedoch bei elektrischen Geräten die positive Konnotation „an“, „funktionstüchtig“. Schraffierungen haben den großen Nachteil, nicht gut kombinierbar zu sein, Blinken zieht zu viel Aufmerksamkeit auf sich und wirkt störend.

Schriftform und -größe

Schriftform und -größe lassen sich nur auf Symbole anwenden, bzw. mit diesen kombinieren. Somit können diese Kodierungsformen nur dann zur Visualisierung von Restriktionsverletzungen verwendet werden, wenn Zuteilungen mittels Symbolen visualisiert werden. Werden dann Regelverstöße durch eine weitere Stufe dieser Kodierungsform visualisiert, so erregt dies, wegen der nur leidlichen Unterscheidbarkeit dieser Stufen, kaum Aufmerksamkeit. Schriftform und -größe lassen sich aber, wenn überhaupt, dann sehr gut mit nur geringen zusätzlichen Platzbedarf (im Falle der Schriftgröße) kombinieren.

Linienarten

Linienarten sind bedingt zur Erregung von Aufmerksamkeit geeignet, sollten jedoch nicht benutzt werden, da üblicherweise der aktuelle Sitz des Tastaturfokus (siehe 3.3) durch eine stärkere Umrandung, also eine besondere Linienart dargestellt wird. Somit assoziiert ein Benutzer mit dieser Kodierungsform traditionell eine andere Bedeutung, auch ließen sich aufgrund der nur mäßigen Unterscheidbarkeit verschiedener Linienarten das Tastaturfokus und Darstellungen von Restriktionsverletzungen nur schwer unterscheiden.

Fazit

Insgesamt ist Farbe das beste Mittel, um Regelverstöße zu visualisieren. Prinzipiell ist es angebracht, unterschiedliche Kodierungsformen für die Darstellung der Zuordnung und für die integrierte Darstellung von Randbedingungen zu wählen, um keine Verwirrung zu erzeugen und dem Benutzer einfache Richtlinien geben zu können (z.B. Symbole werden für die Darstellung der Zuordnung, Farbe für die der Regelverstöße benutzt) und diese Trennung möglichst konsequent einzuhalten (Tabelle 2: Prinzip der Konsistenz). Falls dies auf der Ebene der Kodierungsform nicht praktikabel ist, sollte eine solche Trennung zumindest auf der Ebene der Stufen einer Kodierungsform durchgeführt werden – zum Beispiel könnten Rottöne systematisch für die Visualisierung von Regelverstößen, Grautöne für die Darstellung der Zuordnung selbst verwendet werden. Zumindest eine Kodierungsform oder eine Stufe einer Kodierungsform sollte also für die interne Visualisierung von Randbedingungen reserviert sein, Konsistenz ist hier essentiell.

3.2.2.4 Spezielle Aspekte einzelner Regeltypen

Vermeidung einzelner Zuteilungen

Das ist der einfachste Fall einer Verletzung – eine einzelne Zuteilung verstößt gegen eine Randbedingung. Somit sollte die Darstellung des Regelverstoßes am Ort des entsprechenden Markierungssymbols mit einer der in 3.3.2.3 erwähnten Kodierungen durchgeführt werden (Abbildung 20). Dadurch wird sowohl der Fakt, daß es einen Regelverstoß gibt, als auch die Information über die Zuteilung, die ihn verursacht, effektiv und vollständig kodiert. Diese Visualisierung ist immer möglich, da die Zuteilung in der Zuordnungsdarstellung vorhanden sein muß.

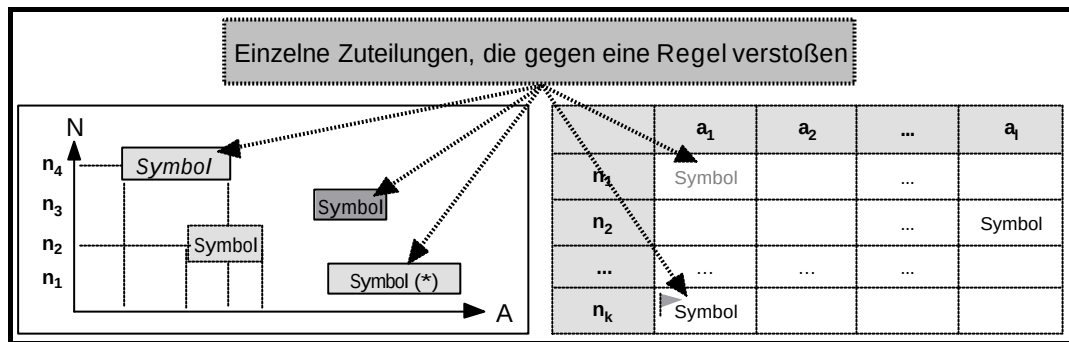


Abbildung 20: Mögliche Visualisierungen des Typs Vermeidung

Durchsetzung einzelner Zuteilungen

Dieser Fall ist etwas diffiziler zu visualisieren, da hier eine Zuteilung nicht erfolgt ist, die durchgesetzt werden sollte. Somit kann auch nicht das entsprechende Markierungsobjekt gekennzeichnet werden, da es nicht vorhanden ist. Als Alternative kann wie in Abbildung 21 ein besonders gekennzeichnetes Markierungsobjekt an der Stelle der erwünschten Zuteilung dargestellt werden oder die tatsächlich erfolgten Zuteilungen der beteiligten Objekte (des Angebots- und des Nachfrageobjekts, die einander zugeteilt sein sollten) als unerwünscht angesehen und wie im Falle der Vermeidung markiert werden. Ersteres erscheint aussagekräftiger, ist jedoch nicht immer möglich, da der Platz bereits besetzt sein könnte und eventuell kein Raum für zwei oder mehr Markierungsobjekte ist.

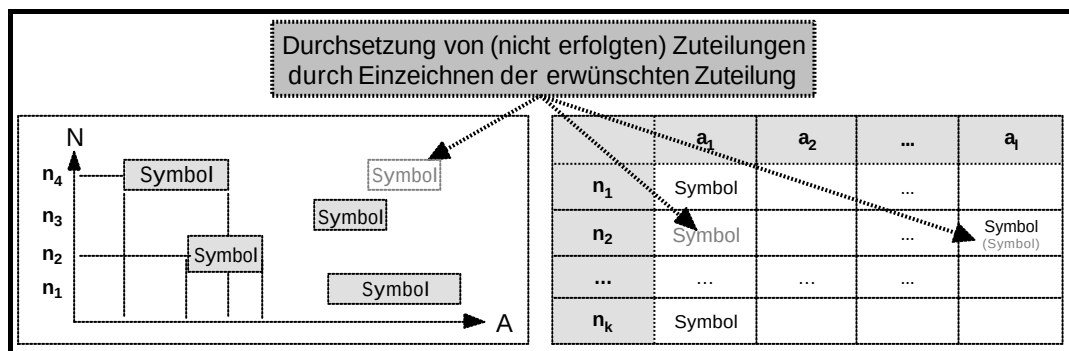


Abbildung 21: Mögliche Visualisierungen des Typs Durchsetzung

Komplexe Vermeidung

Bei diesem Typ führen verschiedene Zuteilungen zusammen zu einem Regelverstoß. Somit liegt es nahe, diese Zuteilungen auch als Orte der Visualisierung auszuwählen und durch eine identische Kodierungsform als zusammengehörig zu kennzeichnen (Abbildung 22). So kann der Benutzer eine unzulässige Kombination mit einem Blick erkennen. Diese Art der Visualisierung ist immer möglich, da die Zuteilungen auf jeden Fall in der Zuordnungsdarstellung visualisiert sind.

Allerdings wird schnell die Obergrenze an gleichzeitig darstellbaren Kodierungsformen, bzw. Stufen erreicht; bei Farben liegt die Zahl gleichzeitig darstellbarer Stufen zum Beispiel bei 6 (Tabelle 3), nach [Shneiderman 98] bei 4. Ist jedoch sichergestellt, daß sich in Bezug auf Regelverstöße zusammengehörige Zuteilungen auf einzelne Spalten, bzw. Zeilen beschränken, und hat auch der Benutzer diese Erwartung, so muß die Wahl der Kodierungsform nur innerhalb der jeweiligen Spalte, bzw. Zeile eindeutig sein. Das ist unter anderem dann der Fall, wenn die jeweilige Randbedingung nur Zuteilungen eines Objektes betrifft und Zuteilungen dieses Objekts immer nur innerhalb einer Zeile, bzw. Spalte der Tabelle auftreten können. Bei den in Kapitel 4 untersuchten Beispielen der Dienstplanung im Krankenhaus trifft diese Einschränkung zum Beispiel auf alle Regeln zu.

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	...	n_l
a_1	x		x							...	▶ x
a_2		x **		x	x **			▶ x			
...	...	...	...	...	...	...	...		▶ x	...	...
a_k							x	x		...	

Abbildung 22: Mögliche Visualisierungen des Typs komplexe Vermeidung

Komplexe Durchsetzung

Auf den ersten Blick bietet es sich wie im Falle der einfachen Vermeidung an, die Zuteilungen darzustellen, die durch die Regel durchgesetzt werden sollen. Allerdings ist das nicht immer eindeutig möglich, da diese Zuteilungen bei der komplexen Durchsetzung oft nicht absolut, sondern relativ zu anderen Zuteilungen spezifiziert sind: wenn zum Beispiel zwei Pflegekräfte derselben Schicht zugeteilt werden sollen, so ist anders als bei der einfachen Durchsetzung keine eindeutige Anzeige der konkreten erwünschten Zuteilung möglich. Ist die Pflegekraft A der Frühschicht (F), die Pflegekraft B der Spätschicht (S) zugeteilt, sollten aber beide derselben Schicht zugeteilt werden, so könnte man willkürlich festlegen, daß die Zuteilung (B, F) oder die Zuteilung (A, S) durchgesetzt werden soll. Dies würde gegen das Prinzip der Konsistenz (Tabelle 2) und, falls diese Entscheidung willkürlich vom System getroffen wird, gegen die Forderung nach Sitz der Kontrolle beim Anwender (Tabelle 2) verstoßen. Wählt und visualisiert man beide gleichzeitig, so könnte dies den Benutzer in die Irre führen, schließlich führt die Umsetzung zu den neuen Zuteilungen (B, F) und (A, S) und die beiden Pflegekräfte arbeiten wieder in unterschiedliche Schichten. Deswegen erscheint es besser, für diesen Restriktionstyp die tatsächlich erfolgten Zuteilungen der beteiligten Objekte wie im Falle

der komplexen Vermeidung zu markieren, also im Beispielsfall die ursprünglichen Zuteilungen (A, F) und (B, S).

Ziel- oder Grenzwert einer arithmetischen Funktion

Arithmetische Werte haben den großen Vorteil, daß sie sich sehr anschaulich und effektiv kodieren lassen, etwa bildlich durch Balkendiagramme und Zeigerinstrumente oder traditionell durch Ziffern. Anders als im allgemeinen Fall, für den Symbole das Mittel der Wahl darstellen, können Zahlenwerte also generell auch bildlich visualisiert werden, solange nicht absolut exakte Werte benötigt werden.

Oft ist es möglich, einen Zahlenwert eindeutig mit einem bestimmten Punkt der Zuordnungsdarstellung zu assoziieren, also einer Zuteilung, einem Attribut oder Objekt. Beim Zahlenwert Ist-Stunden einer Pflegekraft, der möglichst nahe an dem Wert Soll-Stunden liegen sollte, wären dies zum Beispiel das Attribut Ist-Stunden, das Objekt Pflegekraft und alle Zuteilungen dieser Pflegekraft. In einem solchen Fall kann eine Verletzung einer Regel bezüglich eines Ziel- oder Grenzwerts dann durch die Darstellung des tatsächlichen Werts und des Ziel- oder Grenzwerts an eben dieser Stelle visualisiert werden. Dabei erscheint die Verwendung eines Balkendiagramms mit dem Ziel- oder Grenzwert als Vergleich sinnvoller als die Verwendung einer Skala oder von Ziffern, da erstere Platz spart und eventuelle Abweichungen zum Ziel- oder Grenzwert auf den ersten Blick zu erkennen läßt. Abbildung 23 zeigt die unterschiedlichen Visualisierungsmöglichkeiten.

Als Alternative können die Zuteilungen, über die der Funktionswert berechnet wird, wie im Fall der komplexen Vermeidung markiert werden. Dies ist vor allem dann angebracht, wenn eine Assoziation des arithmetischen Werts mit einer Stelle der Zuteilung nicht möglich, für den Benutzer nicht nachvollziehbar oder insgesamt nicht sinnvoll ist.

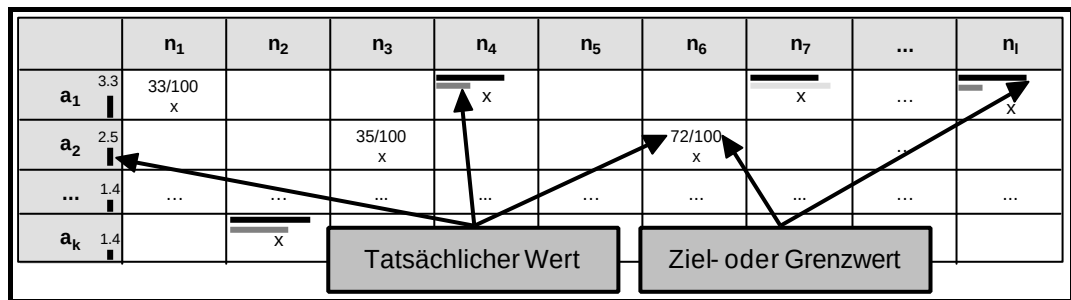


Abbildung 23: Mögliche Visualisierungen des Typs Arithmetischer Ziel- oder Grenzwert

Reihenfolge

Zur Visualisierung des Typs Reihenfolge können prinzipiell dieselben Mittel angewandt werden wie beim Typ komplexe Vermeidung, bzw. Durchsetzung. Zusätzlich bietet sich hier jedoch das Einblenden der richtigen Reihenfolge (Abbildung 24) oder eines Hierarchiegraphen an. Diese Hilfestellung sollte jedoch nur auf Wunsch des Benutzers gewährt werden, also nicht zur Standarddarstellung gehören, da sie andere Teile der Zuordnungsdarstellung verdecken könnte.

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	...	n_l
a_1	x	x		x	x		x	...	x
a_2	x		x		x		x	...	
...	...	...	...	...	...	...	...	...	...
a_k								...	

Abbildung 24: Mögliche Visualisierung des Typs Reihenfolge

3.2.2.5 Besondere Situationen

Besondere Situationen bei der Darstellung von Restriktionsverletzungen treten unter anderem auf, wenn

- ?? verschiedene Verletzungen an derselben Stelle visualisiert werden sollen.
- ?? der Benutzer es bevorzugt, daß nicht alle Verletzungen gleichzeitig visualisiert werden.
- ?? der Benutzer die Visualisierung einzelner Verletzungen hervorheben möchte.

3.2.2.5.1 Konflikte bei der Visualisierung

Treten bei einer Zuordnung viele Restriktionsverletzungen gleichzeitig auf, so kann es vorkommen, daß an derselben Stelle einer Zuordnungsdarstellung mehrere Verletzungen gleichzeitig dargestellt werden sollen. Folgende Möglichkeiten gibt es, einen solchen Konflikt zu lösen:

- ?? Nur der als schwerer bewertete Regelverstoß wird visualisiert, die restlichen Regelverstöße werden an dieser Stelle nicht dargestellt und somit eventuell nur extern visualisiert.
- ?? Eine besondere Kodierungsform wird für diesen Fall reserviert, die dann nur die Information „Mehrere Regelverstöße gleichzeitig“ kodiert.
- ?? Falls möglich werden die Regelverstöße in Hinblick auf die Darstellung zu einer entsprechend schwerer bewerteten Verletzung zusammengefaßt.
- ?? Es wird versucht, die verschiedenen Kodierungsmittel der einzelnen Regelverstöße gleichzeitig darzustellen. Dies ist jedoch nicht immer möglich, bei Symbolen oder

Icons vor allem aus Platzgründen. Manchmal wäre die resultierende Darstellung zwar technisch möglich, ist aber wenig sinnvoll, etwa wenn als Kodierungsform für Regelverstöße die Vordergrundfarbe gewählt wurde und die mit den Restriktionsverletzungen assoziierte Stelle der Zuordnungsdarstellung dann vielfarbig darzustellen wäre („Symbol“).

3.2.2.5.2 Filterungsmöglichkeiten

Es ist nicht immer sinnvoll, alle Verletzungen von Randbedingungen auf einmal anzuzeigen, da der Benutzer überlastet oder frustriert werden könnte. Deswegen sollte eine graphische Oberfläche dem Benutzer die Möglichkeit offerieren festzulegen, welche Verletzungen von Randbedingungen dargestellt werden sollen (Tabelle 2: Individualisierung, Kontrolle beim Anwender). Hier sind folgende Ansätze denkbar, wobei der Benutzer differenzieren können sollte, ob die getroffene Wahl auf die interne, die externe oder auf beide Visualisierungen angewendet werden sollte.

- ?? Anzeige nach absoluter Schwere einer Verletzung (nur Regelverstöße, die schwer oder noch massiver sind).
- ?? Anzeige nach relativer Schwere einer Verletzung – (etwa „die n schwersten“ oder „die schwersten n%“).
- ?? Anzeige nur bestimmter Regeln (etwa ausgewählt nach ihrem Namen oder ihrer Beschreibung).
- ?? Anzeige nur der Verletzungen, an denen bestimmte Objekte beteiligt sind (alle Verletzungen, an denen Nachfrageobjekt a_i beteiligt ist).
- ?? Explizites Ausschalten der Visualisierung einzelner Regelverstöße.
- ?? Explizites Ausschalten der Visualisierung aller Regelverstöße.

3.2.2.5.3 Detaillierte Visualisierung einzelner Restriktionsverletzungen

Da sich bei der integrierten Visualisierung eventuell nicht alle relevanten Informationen gleichzeitig darstellen lassen, sollte die Oberfläche auf Wunsch¹⁰ des Benutzers durch die Kodierung der Randbedingung zu einer ausführlichen verbalen Meldung gelangen können. Diese kann durchaus temporär andere Teile der Zuordnungsdarstellung verdecken. Gleichzeitig sollte eine nur für diesen Zweck reservierte Kodierungsform alle mit dieser Regelverletzung assoziierten Stellen der Zuordnungsdarstellung kennzeichnen (Abbildung 23). Das Kodierungsmittel Blinken scheint hier besonders geeignet, da es ohnehin wegen seiner zu stark ablenkenden Wirkung nur temporär und lokal begrenzt eingesetzt werden sollte. Allerdings sollte auch diese Kodierungsform wieder einstellbar sein.

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	...	n_l
a_1	x		x							...	x
a_2		x **		x	x **			x		...	
...	...	...	...								
a_k											
Randbedingung				Zuteilung(en)				Grund	Bewertung		
R1				$(n_1, a_1), (n_3, a_3), (n_4, a_2)$				...	200.5		

Abbildung 25: Ausführliche Visualisierung einer ausgewählten Restriktionsverletzung (mit Schraffierungen als Ersatz für Blinken)

3.2.2.6 Standarddarstellung

Nachdem bisher die möglichen Antworten auf die Grundfragen bezüglich der Visualisierung von Randbedingungen erörtert wurden, soll nun die problemunabhängige Standarddarstellung von Restriktionsverletzungen festgelegt werden. Allerdings ist diese in der Regel einer speziell auf das Problem angepassten Darstellung unterlegen, vor allem bei der internen Visualisierung. Schließlich existieren, wie bereits am Beispiel der Farben erläutert, in unterschiedlichen Bereichen widersprüchliche Konventionen und Erwartungen bezüglich der Darstellung, denen eine Standarddarstellung nur schwer gerecht werden kann. Auch wenn es eventuell möglich wäre, solches Wissen in umfangreichen Wissensbasen zu akkumulieren und dann anzuwenden, ist es bei weitem praktikabler, wie in 3.2.2.7 beschrieben, eine einfache direkte Adaption der Visualisierung zu ermöglichen. Somit kann ein Designer dann eine problemspezifische Standarddarstellung implementieren oder ein Anwender seine eigenen Visualisierungswünsche umsetzen (Tabelle 2: Individualisierung). Die problematische Forderung nach einer Standarddarstellung, die immer allen Ansprüchen gerecht wird, kann dann entfallen.

¹⁰ Siehe Abschnitt 3.3 über Interaktivität zu den Möglichkeiten eines Benutzer, einen solchen Wunsch zu artikulieren.

3.2.2.6.1 Externe Standarddarstellung

Definitiv sollte eine externe Visualisierung Bestandteil der Standarddarstellung sein, die sämtliche Regelverstöße in einem eigenen Fenster mit allen relevanten Informationen explizit mittels Symbolen, bzw. Text anzeigt. Um diese Daten strukturiert darzustellen, bietet sich eine Tabelle mit der in Tabelle 2 aufgeführten Funktionalität an (z.B. nach beliebigen Spalten sortierbar, mit beliebiger Spaltenreihenfolge, Schriftgröße). Abbildung 26 zeigt am Beispiel der Dienstplanung im Krankenhaus, wie die externe Standardvisualisierung aussehen sollte. Externe und interne Visualisierung sollten insofern miteinander verknüpft sein, als es möglich sein sollte, über die Selektion einer Randbedingung in der externen Visualisierung die dazugehörigen Zuteilungen und Objekte in der Darstellung der Zuordnung zu lokalisieren. Ebenso sollte man vice versa über Zuteilungen oder Objekte zur externen Darstellung der Restriktionsverletzungen gelangen, an denen diese beteiligt sind.

Randbedingung	Zuteilung(en)	Grund	Bewertung
...	...	...	...
Mindestabstand zweier Schichten	Pflegekraft A,, Spätschicht am 3.4 und Pflegekraft A,, Frühschicht am 4.4	Abstand zweier Schichten einer Pflegekraft muß mindestens 10 Stunden betragen	Verstoß gegen wichtige Dienstvorschrift
...	...	...	...

Abbildung 26: Externe Standarddarstellung

3.2.2.6.2 Interne Standarddarstellung

Zusätzlich sollte es eine interne Visualisierung geben, bei der das primäre Ziel Einfachheit durch Konsistenz sein sollte, um einen Erstbenutzer nicht zu überfordern, sondern ihm systematisch ein schnelles Verständnis der Oberfläche zu ermöglichen (Grundziele der subjektiven Zufriedenheit und leichten Erlernbarkeit). Schließlich kann ein Benutzer, sobald er die Oberfläche subjektiv beherrscht, die erweiterte Funktionalität selektiv nach seinen Vorstellungen einsetzen (siehe 3.3.2.7). Die Diskussion der internen Standarddarstellung soll wieder anhand der Grundfragen (Was? Wo? Wie?) strukturiert werden:

1. Relevante Information

Bei der Frage, welche Informationen dargestellt werden sollten (Was?), muß wieder zwischen den vier zentralen Inhalten dieser Informationen differenziert werden:

?? Bewertung

Die Kodierung der Bewertung ist prinzipiell möglich, allerdings müßte eine entsprechende Konvention gefunden werden. Denkbar wäre zum Beispiel im Falle der Dienstplanung im Krankenhaus: „blau, #, ✂ entspricht einem Verstoß gegen gesetzliche Vorschriften, gelb, &, ✂ einem Verstoß gegen Präferenzen des Pflegepersonals“.

Für Symbole und bildliche Formen läßt sich die Bewertung zumindest andeutungsweise intuitiv verständlich über die (Schrift-)Größe ausdrücken – schwerere Verletzungen sollen deshalb durch die Standarddarstellung über entsprechend größere Symbole, bzw. bildliche Formen ausgedrückt, leichtere über kleinere Symbole bzw. bildliche Formen. Bei Farbe soll die Grundeinstellung niedriger bewertete Restriktionsverletzungen mit einer helleren Schattierung, höher bewertete Regelverstöße mit einer entsprechend dunkleren Farbabstufung derselben Grundfarbe visualisieren, zum Beispiel „hellrot für einen leichten Regelverstoß, rot für einen mittelschweren, dunkelrot für einen schweren“.

?? **Betroffene Randbedingung**

Eine bestimmte Kodierung oder Stufe einer Kodierungsform könnte für eine gewisse Randbedingung stehen, bei der Dienstplanung im Krankenhaus zum Beispiel: „rot, *“, eine Fahne entspricht einer Verletzung der Regel „Mindestabstand zwischen zwei Schichten einhalten“. Allerdings haben Zuordnungsprobleme oft eine große Zahl von Randbedingungen – im Falle der Schwesternplanung etwa 20, in der Schulstundenplanung noch mehr, so daß nur Symbole über eine ausreichende Anzahl an Stufen verfügen, um zwischen allen Randbedingungen eines komplexen Zuordnungsproblems zu differenzieren. Somit würde eine solche Kodierung zu Problemen führen - entweder müßte eine Vielzahl von Abkürzungen eingeführt werden, was dem Grundprinzip der leichten Erlernbarkeit entgegenstünde, oder die Kodierung müßte so ausführlich sein und damit so viel Platz einnehmen, daß sie sich nicht mehr in die Zuordnungsdarstellung integrieren ließe. Allenfalls eine spezielle Kodierung nur der wichtigsten oder häufigsten Randbedingungen könnte sinnvoll sein; diese Entscheidung sollte aber dem Benutzer überlassen werden, die Standarddarstellung sollte nicht versuchen, einzelne Regeln besonders zu kodieren.

?? **Kodierung des Grundes**

Falls eine Regelverletzung mit einem Attribut assoziiert werden kann und dieses Attribut Bestandteil der Zuordnungsdarstellung ist, sollte – zusätzlich zu den sonstigen Orten – das Attribut als Ort der Visualisierung gewählt werden und so der Grund für die Restriktionsverletzung zumindest angedeutet werden.

?? **Kodierung der Verursacher**

Soweit möglich sollen immer die Zuteilungen, die zu der Restriktionsverletzung führen, entsprechend gekennzeichnet werden (siehe Kodierungsort).

2. Kodierungsort

Als Kodierungsort sollten immer tatsächlich durchgeführte Zuteilungen dienen. Das sind bei Randbedingungen vom Typ

?? Vermeidung genau die Zuteilungen, die durchgeführt wurden, obwohl sie vermieden werden sollten.

- ?? Durchsetzung genau die Zuteilungen der entsprechenden Angebots- und Nachfrageobjekte, die anstelle der erwünschten Zuteilungen durchgeführt wurden.
- ?? Arithmetischer Ziel- oder Grenzwert die Zuteilungen, über die der Wert berechnet wurde.
- ?? Reihenfolge/Muster: Die Zuteilungen, die gegen die erforderliche Reihenfolge verstoßen, bzw. nicht dem Muster entsprechen.

3. Kodierungsform

Als Standardkodierungsform sollte aus den erläuterten Gründen eine einzelne Farbe dienen, so weit möglich rot oder eine andere Grundfarbe. Kann keine Farbe verwendet werden, so sollte ein allgemein mit einer Warnung assoziiertes Icon wie ein Stoppschild oder ein Ausrufezeichen zum Einsatz kommen. Es sollte also nur eine Kodierungsstufe einer Kodierungsform zur internen Visualisierung von Restriktionsverletzungen benutzt werden.

3.2.2.6.3 Standardverhalten bei besonderen Situationen

Die Oberfläche sollte sich bei den in 3.2.2.5 beschriebenen besonderen Situationen wie folgt verhalten:

- ?? Restriktionsverletzungen werden nicht gefiltert. Die Entscheidung über diese Frage sollte dem Benutzer überlassen werden und nicht vom System getroffen werden (Tabelle 2: Sitz der Kontrolle beim Anwender), da das eigenmächtige Weglassen von Regelverstößen das Vertrauen in das System mindern könnte; schließlich werden beim Filtern offensichtliche Regelverstöße nicht angezeigt. Das Grundziel, subjektive Zufriedenheit bezüglich der Oberfläche beim Anwender zu erzeugen, wäre verletzt.
- ?? Das System sollte versuchen, auch bei Konflikten alle Restriktionsverletzungen gleichzeitig darzustellen. Nur falls dies nicht möglich ist, sollte es Verletzungen zusammenzufassen, falls dies wiederum nicht praktikabel ist, sollte eine für diesen Zweck reservierte Kodierungsform eingesetzt werden. Nur die schwerste Restriktionsverletzung anzuzeigen sollte nur die letzte Option sein.
- ?? Das System sollte einzelne Regelverstöße auf die in Abbildung 25 gezeigte Weise besonders hervorheben. Die mit der Verletzung assoziierten Stellen der Zuordnung sollten durch Blinken besonders hervorgehoben werden.

3.2.2.7 Benutzer- und problemspezifische Adaption

Nach Tabelle 2 soll der Benutzer alle Aspekte der Oberfläche individualisieren können, also auch die Visualisierung von Regelverstößen. Eine Pflegekraft, die einen Dienstplan für das Krankenhaus erstellt, sollte also ihre persönlichen Einstellungen vornehmen können und die Darstellung so zum Beispiel an ihre Arbeitsbedingungen (eventuell nur monochromer Bildschirm) anpassen können.

Zugleich sollte ein Designer die Möglichkeit haben, problemspezifische Standardeinstellungen zu spezifizieren, die dann die in 3.2.2.6 beschriebenen allgemeinen Standardeinstellungen ersetzen. Somit kann er sein Wissen über problemspezifische Darstellungskonventionen einbringen und normalen Benutzern von vornherein eine bessere Oberfläche bieten.

Dabei sollten beiden die gleichen Möglichkeiten offenstehen, mit dem Unterschied, daß die Anpassungen des Designers zu den neuen problemspezifischen Standarddarstellungen für alle Normalbenutzer werden sollen, die eines Normalbenutzers nur für ihn selbst gelten sollten. Für jede Restriktion sollten folgende Optionen bezüglich der Visualisierung von Verletzungen bestehen:

Visualisierungsaspekt Mögliche Werte	
Orte der Visualisierung (mehrere Werte möglich)	?? Zuteilungen ?? Attribute ?? Beteiligte Objekte
Kodierungsform	?? Farbe ?? Symbol ?? Bildliche oder geometrische Form ?? Pfeile, bzw. Linienart (nur bei komplexen Randbedingungen) ?? Schriftform und Größe (nur wenn die Zuordnung durch Symbole visualisiert wird) ?? Schraffierung ?? Blinken
Kodierungsstufe	Siehe Tabelle 5 (Abhängig von Kodierungsform)
Visualisierung der Bewertung	Siehe Tabelle 5 (Abhängig von Kodierungsform)
Weitere Optionen:	?? Einblenden der Reihenfolge (bei Regeln bezüglich der Reihenfolge) ?? Anzeige der erwünschten Zuteilung (bei Regeln bezüglich der Durchsetzung von Zuteilungen) ?? Ständige integrierte Anzeige der Werte (bei Regeln bezüglich arithmetischen Zielwerten)

Tabelle 4: Allgemeine Einstellungsmöglichkeiten des Benutzers

Dabei sind folgende Kodierungsstufen, bzw. Möglichkeiten, die Bewertung zu visualisieren, in Abhängigkeit von der Kodierungsform auswählbar:

Kodierungsform	Spezifikation der Kodierungsstufe durch	Anzeige der Bewertung durch
Farbe	Farbauswahl-Dialog	Helligkeit/keine Anzeige
Symbol	Texteingabefeld	Größe/keine Anzeige
Bildliche und geometrische Form	Auswahl aus vorhandenen bildlichen und geometrischen Formen	Größe/keine Anzeige
Pfeile, bzw. Linienart	Auswahl aus vorhandenen Pfeilen/Linienarten	Pfeil-, bzw. Liniendicke/keine Anzeige
Schriftform und -größe	Auswahl aus vorhandenen Schriftformen und Größen	Keine Anzeige
Schraffierung	Auswahl aus vorhandenen Schraffierungen	Keine Anzeige
Blinken	-	Blinkfrequenz/ Keine Anzeige

Tabelle 5: Spezielle Einstellungsmöglichkeiten des Benutzers in Abhängigkeit von der Kodierungsform

Abbildung 27 zeigt, wie ein solcher Einstellungsdialog bei der Dienstplanung im Krankenhaus aussieht:

Randbedingung	Ort	Kodierungsform	Kodierungsstufe	Bewertung
...	...	...		...
Mindestabstand zweier Schichten	Zuteilungen	Farbe	Rot	Keine Anzeige der Bewertung
Einhalten der Soll-Stunden	Attribut Soll-Stunden	Farbe	Grün	Helligkeit
Nicht mehr als 5 Nachtschichten hintereinander	Zuteilungen	Symbol	***	Größe
...	...	...		...

Abbildung 27: Dialog zur Spezifikation der Darstellung von Regelverstößen

Wie in 3.2.2.4 beschrieben, muß bei komplexen Randbedingungen eventuell ein Zusammenhang zwischen den Zuteilungen, die den Regelverstoß verursachen, dargestellt werden. Treten mehrere solche Verstöße gleichzeitig auf, so müssen auch verschiedene Zuteilungskombinationen simultan als zusammengehörig gekennzeichnet werden, wozu im allgemeinen eine jeweils unterschiedliche Visualisierung gewählt werden muß (siehe Abbildung 22). Mittels der beschriebenen Einstellungsmöglichkeiten kann ein Benutzer nun sicherstellen, daß er gleichzeitig darzustellende Verstöße gegen unterschiedliche Randbedingungen immer auseinanderhalten kann, indem er für diese Regeln jeweils unterschiedliche Visualisierungen wählt.

Um auch gleichzeitig auftretende Verstöße gegen dieselbe Regel auseinanderhalten zu können, sollte ein Benutzer zusätzlich nicht nur eine, sondern beliebig viele Visualisierungen für jede einzelne Regel festlegen können, die dann in einem solchen Fall zum Einsatz kommen würden.

So kann die Regel „Mindestabstand zweier Schichten soll 10 Stunden betragen“ der Dienstplanung im Krankenhaus für eine Pflegekraft gleichzeitig mehrmals verletzt sein. Legt ein Benutzer nun für diese Regel als erste Visualisierung das Symbol „*“, als zweite das Symbol „&“ fest, und wäre die Regel für die Pflegekraft A sowohl beim Übergang der Nachtschicht des 13.3 auf die Spätschicht des 14.3 als auch beim Übergang von der Spätschicht des 14.3 auf die Frühschicht des 15.3 verletzt, so würde folgende Darstellung resultieren:

...	...	13.3	14.3	15.3	...
...	..				...
A	...	N*	S*&	F&	...
...	...				...

Abbildung 28: Zwei simultane Verstöße gegen dieselbe Regel

3.2.2.8 Fazit

Die Anforderung an die Oberfläche lautet, all die in diesem Abschnitt beschriebenen Möglichkeiten der Visualisierung zu unterstützen und darüber hinaus die geschilderte Standardvisualisierung (sowohl die externe als auch die interne) zu implementieren. Weiter muß sie die beschriebenen Möglichkeiten zur Adaption der Visualisierung zur Verfügung stellen.

3.3 Anforderungen an die Interaktivität

In diesem Kapitel sollen nun die Ansprüche an die Benutzeroberfläche erarbeitet werden, die sich zusätzlich zu den bisher aufgeführten Anforderungen aus der Unterstützung der interaktiven Lösung ergeben. Dabei sollen zuerst die funktionellen Aspekte (Welche interaktiven Eingriffsmöglichkeiten sollte es mindestens geben?) behandelt werden, dann die Frage, wie diese – vor allem graphisch – repräsentiert werden sollten (Wie soll der Benutzer diese aufrufen können?).

3.3.1 Funktionelle Grundanforderungen

Wie im Einführungskapitel erwähnt, läßt der interaktive Ansatz zur Lösung von Zuordnungsproblemen den Lösungsvorgang auf Wunsch vom menschlichen Benutzer steuern. Dies zu ermöglichen ist Aufgabe des Benutzerinterface. [Stary 96] bezeichnet solche Computerprogramme als interaktiv, „die für Benutzer Eingriffsmöglichkeiten in ihren Kontroll- und Datenfluß aufweisen“. Die „Benutzungsschnittstelle“ definiert er als „Summe interaktiver Eingriffsmöglichkeiten für Benutzer“. Allerdings gilt, daß ein Prädikat wie „interaktiv“ einem Programm, oder besser der Benutzung eines Programms, nur abgestuft zukommen kann; nach obiger Definition könnte man jedes Programm, das dem Anwender auch nur wenige Eingriffsmöglichkeiten mittels kryptischer Tastenkombinationen bietet, bereits als interaktiv bezeichnen. Deswegen soll hier – ausgehend von der Definition eines Zuordnungsproblems ein funktionelles Anforderungsprofil erstellt werden, dem ein wirklich interaktives oder hochgradig interaktives Programm zur Lösung von Zuordnungsproblemen genügen sollte [vgl. Busch 97].

Bei Zuordnungsproblemen geht es – gemäß ihrer Definition - prinzipiell darum, Nachfrageobjekten Angebotsobjekte zuzuweisen. Ein interaktives Werkzeug muß deshalb dem Benutzer zumindest folgende Eingriffsmöglichkeiten bieten:

- ?? Zuweisung eines Angebotsobjekts an ein Nachfrageobjekt durch den Benutzer
- ?? Zuweisung eines Nachfrageobjekts an ein Angebotsobjekt durch den Benutzer
- ?? Rücknahme vom Benutzer ausgewählter Zuteilungen

Weiter sollte die Oberfläche den Benutzer bei der manuellen Zuordnung unterstützen können, falls dieser es wünscht, also Vorschläge machen oder Bewertungen abgeben können. Die Anforderungen an die Funktionalität lauten somit:

- ?? Vorschlag von Angebotsobjekten für ein vom Benutzer ausgewähltes Nachfrageobjekt/eine vom Benutzer ausgewählte Gruppe von Nachfrageobjekten durch das System
- ?? Vorschlag von Nachfrageobjekten für ein vom Benutzer ausgewähltes Angebotsobjekt/eine vom Benutzer ausgewählte Gruppe von Angebotsobjekten durch das System
- ?? Anzeigen von Alternativen für eine vom Benutzer ausgewählte Zuordnung durch das System
- ?? Bewertung der aktuellen Zuordnung durch das System zu einem vom Benutzer bestimmten Zeitpunkt (falls keine ständige Bewertung in Echtzeit möglich ist)

Schließlich sollte der Anwender Bereiche des Gesamtproblems auswählen können, die das System selbständig bearbeiten soll, ohne daß der Benutzer eingreifen muß, so daß der Benutzer eventuell nur kritische Teilbereiche selbst planen und unproblematische vom System planen lassen kann. Dabei sollte der Benutzer festlegen können, ob das System bereits durchgeführte Zuteilungen als feststehend auffassen sollte oder ob es solche Zuteilungen eventuell zurückziehen und durch andere Zuteilungen ersetzen können sollte. Eine solche Einteilung sollte sowohl generell („Existierende Zuteilungen *sollen auf keinen Fall / können bei Bedarf* aufgehoben werden“) als auch differenziert („Bestimmte Zuteilungen *sollen auf keinen Fall / können bei Bedarf* zurückgezogen werden“) artikuliert werden können.

- ?? Zuweisung von Angebotsobjekten für ein vom Benutzer ausgewähltes Nachfrageobjekt/eine vom Benutzer ausgewählte Gruppe von Nachfrageobjekten durch das System
- ?? Zuweisung von Nachfrageobjekten für ein vom Benutzer ausgewähltes Angebotsobjekt/eine vom Benutzer ausgewählte Gruppe von Angebotsobjekten durch das System
- ?? Einstellung der prinzipiellen Vorgehensweise des Systems: Bestehende Zuteilungen sind fest, bzw. können aufgehoben werden.
- ?? Einstellung der konkreten Vorgehensweise des Systems: Bestimmte Zuteilungen können als unveränderlich, bzw. als veränderlich gekennzeichnet werden.

Es erscheint klar, daß ein Programm zur Lösung von Zuordnungsproblemen, das obige Funktionalität bietet, in seiner Struktur eingeschränkter¹¹ und deutlich komplexer¹² als ein System ohne diese Eingriffsmöglichkeiten ist. Dennoch ist dieser Mehraufwand gegenüber konventionellen Systemen angebracht, da ein interaktives Vorgehen bei der Lösung von Zuordnungsproblemen große Vorteile bietet, so

- ?? können manche Randbedingungen nicht oder nur schwer in eine dem Computer verständliche Form gebracht werden, etwa weil ein menschlicher Experte sie intuitiv berücksichtigt und nicht formal spezifizieren kann (Bsp. Ärzteplanung: Bilden eines Ärzteteams für einen Operationssaal) oder weil sie aus rechtlichen oder politischen Gründen nicht explizit in den Computer eingegeben werden können (Bsp. Krankenhausplanung: detaillierte Beurteilung der beruflichen Fähigkeiten des Personals). Bei interaktiven Systemen kann der Benutzer die Einhaltung dieser Regeln dann manuell sicherstellen.
- ?? können einmalig auftretende Randbedingungen im Plan berücksichtigt werden, ohne daß für sie eigens eine Regel implementiert oder bestehende Regeln geändert werden müssen (Bsp. Krankenhausplanung: Pfleger A kann diesen Montag nicht in der Frühschicht arbeiten, obwohl er diese Schicht an diesem Tag als bevorzugte Schicht in seinen Arbeitszeitpräferenzen angegeben hat).
- ?? kann der Anwender schnell selbst erkennen, wenn seine Aufgabenspezifikation zu keiner oder nur einer ungünstigen Lösung führen kann, und diese dann eventuell ändern.

¹¹ So muß der zugrundeliegende Algorithmus die entsprechende Funktionalität aufweisen - es muß zum Beispiel möglich sein, eine Lösung aus einer vorgegebenen Teillösung heraus zu entwickeln oder vorgegebene Lösungen unter Beibehaltung bestimmter Zuteilungen zu verbessern.

¹² Schließlich müssen sowohl die zugrundeliegende Applikation als auch das Benutzerinterface diese zusätzliche Funktionalität unterstützen.

- ?? Werden Fehler in der Spezifikation der Regeln oder Daten leichter erkannt, etwa wenn der Experte interaktiv eine nach seinem Wissen akzeptable Zuordnung ausführt und das System eine Regelverletzung anzeigt, die auf eine falsche Problemspezifikation zurückzuführen ist.
- ?? erzeugt ein interaktives System, anders als ein System, bei dem der Benutzer nur mit einer fertigen Lösung konfrontiert wird, kein subjektives Gefühl der Bevormundung durch den Computer (Grundkriterium der subjektiven Zufriedenheit).
- ?? sieht ein Benutzer eines interaktiven Systems die Auswirkungen eines jeden seiner Schritte sofort; dieses unmittelbare Feedback führt zu einem schnelleren Lern- und Einarbeitungsprozeß (Grundkriterium der geringen Einarbeitungszeit).
- ?? entspricht ein interaktives Vorgehen oft dem traditionellen menschlichen Ansatz zur Problemlösung; ein ursprünglicher Lösungsvorschlag wird Schritt für Schritt erweitert und verbessert, bis er den Ansprüchen genügt. Die Umstellung auf eine computerbasierte Lösungsmethode fällt somit leichter, Ängste vor dem Computer werden schneller abgebaut, da wie gewohnt gearbeitet werden kann. (Grundkriterien der geringen Einarbeitungszeit und der subjektiven Zufriedenheit)

3.3.2 Präsentation der Interaktionsmöglichkeiten

3.3.2.1 Navigation innerhalb der Zuordnungsdarstellung

Die Navigation sollte sich am üblichen Verhalten ähnlicher Programme, etwa von Tabellenkalkulationsprogrammen, orientieren.

Der Benutzer sollte mit Zeigeinstrumenten und der Tastatur innerhalb der Zuordnungsdarstellung navigieren können. Dabei zeigt traditionell eine stärkere Umrandung den Sitz des aktuellen Tastaturfokus an, d.h. der Stelle, an die jegliche Eingaben mittels der Tastatur weitergeleitet werden. Das Fokus des Zeigeegeräts wird üblicherweise durch eine entsprechende Markierung, im allgemeinen einen Mauszeiger, dargestellt. Hat eine Stelle der Zuordnungsvisualisierung das Fokus, so sollten Aktionen wie ein Mausklick oder das Drücken einer Taste an die entsprechende Stelle der Zuordnungsdarstellung weitergeleitet werden und die auf dem jeweiligen Betriebssystem üblichen Reaktionen hervorrufen. Bei Microsoft Windows evoziert die Eingabetaste oder die linke Maustaste im allgemeinen Möglichkeiten zur direkten Manipulation, während die rechte Maustaste oder die Menu-Taste auf der Tastatur selektionsorientierte Interaktionsmöglichkeiten aufruft.

3.3.2.2 Selektion innerhalb der Zuordnungsdarstellung

Auch die Selektion von Bereichen sollte sich an dem üblichen Verhalten ähnlicher Programme orientieren. Unter Microsoft Windows ist es zum Beispiel üblich, daß durch das Anklicken mehrerer Objekt bei gedrückter Steuerungstaste all diese Objekte selektiert werden. Selektierte Objekte sollen durch eine besondere Farbe markiert werden.

3.3.2.3 Selektionsorientierte Interaktionsmöglichkeiten

Unter selektionsorientierten Interaktionsmöglichkeiten versteht man in erster Linie die Menüauswahl, bei der ein Benutzer aus einer Reihe gegebener Optionen auswählen kann [Busch 97]. Sie erfüllt vor allem das Grundkriterium der leichten Erlernbarkeit, da nach [Shneiderman 98] nur ein geringer Lern- und Erinnerungsbedarf besteht, vorausgesetzt, die aufgeführten Handlungsoptionen sind vollständig und verständlich formuliert.

Ruft der Benutzer innerhalb der Zuordnungsdarstellung die Menüauswahl auf, so sollte der Aufruf auf das mit der gegenwärtigen Position des entsprechenden Eingabegeräts assoziierte Objekt bezogen werden und es sollten auch nur Möglichkeiten auswählbar sein, die auf das Objekt anwendbar sind. Bezieht sich der Aufruf also auf eine bestehende Zuteilung, so sollte die Option „Zuteilung durchführen“ zwar aufgeführt, aber nicht auswählbar sein, da sie bei einer bereits durchgeführten Zuteilung nicht sinnvoll wäre. Es sollten aber aus Gründen der Konsistenz immer alle Optionen aufgeführt werden. Nicht anwendbare, aber dennoch aufgeführte Menüoptionen sollten so, wie auf dem jeweiligen Betriebssystem üblich, gekennzeichnet werden, im Regelfall durch eine hellere Schriftfarbe (siehe Abbildung 29).

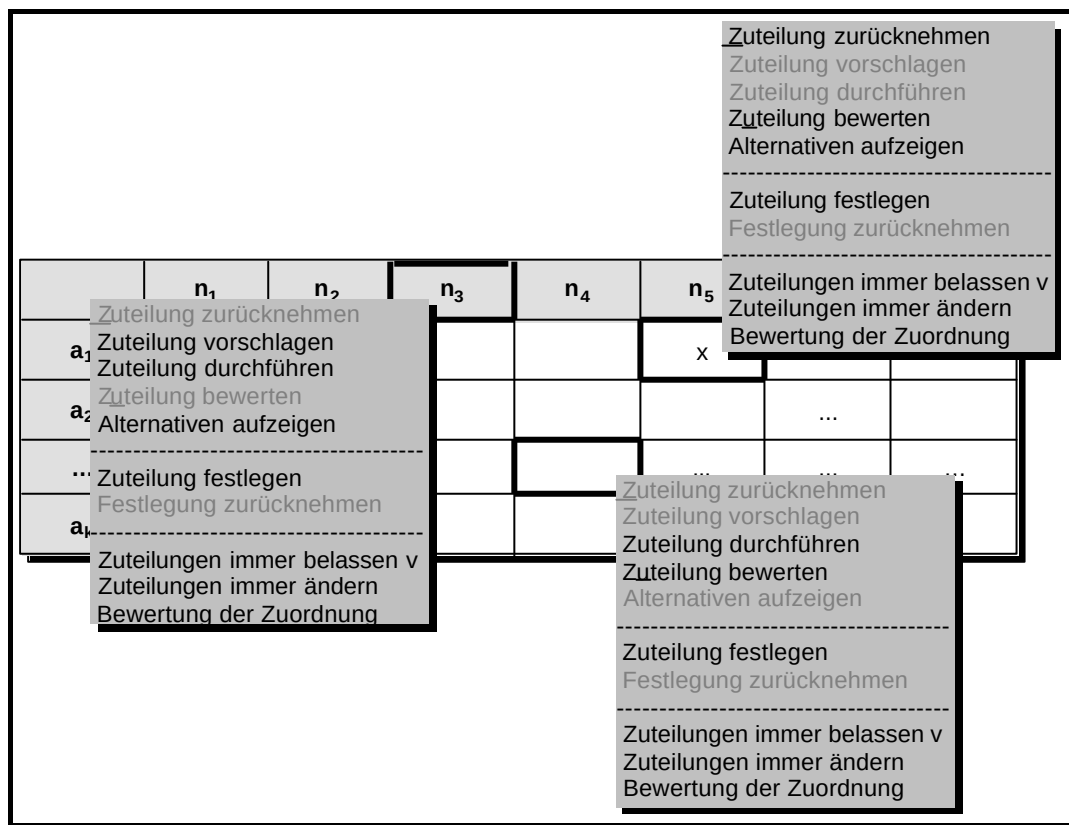


Abbildung 29: Selektionsorientierte Präsentation der Interaktionsmöglichkeiten

3.3.2.4 Direkte Manipulation

Neben der selektionsorientierten Interaktion, die charakteristisch für Computersysteme ist und kein direktes Pendant in der realen Welt hat, steht der Begriff der direkten Manipulation für den Grundsatz, daß der Benutzer eine Aufgabe so lösen kann, wie er sie ohne Computer lösen würde [Shneiderman 98]. Dazu stellt die Oberfläche die Objekte der gewohnten Arbeitsumgebung (oder Metaphern für diese Objekte) dar und ermöglicht es dem Benutzer, diese Objekte in der üblichen Weise zu manipulieren (siehe [Busch 97]), also etwa zu markieren oder zu verschieben.

Übertragen auf Zuordnungsprobleme bedeutet das, daß der Benutzer die wesentlichen Bestandteile der Zuordnung manipulieren kann, also Angebots- und Nachfrageobjekte, Attribute und Zuteilungen. Jeder dieser Bestandteile sollte also je nach Kodierungsform mittels der in Tabelle 5 aufgeführten Spezifikationsmöglichkeiten¹³ verändert werden können. Das Resultat einer solchen Veränderung sollte sofort sichtbar sein. Eine Zuteilung der Dienstplanung für das Pflegepersonal eines Krankenhauses, die zum Beispiel durch ein „N“, also ein Symbol, kodiert wird, sollte zum Beispiel nach Tabelle 4 durch ein Texteingabefeld zu manipulieren sein. Möglichkeiten der Manipulation wären zum Beispiel das Zurücknehmen durch Löschen des Symbols oder Veränderung der Zuteilung durch Ersetzen des Symbols durch ein anderes.

Wichtiger Bestandteil der direkten Manipulation ist die Unterstützung von „Drag&Drop“, d.h. der Möglichkeit, beliebige Objekte der Oberfläche zu verschieben und dadurch Aktionen auszuführen, deren konkrete Bedeutung vom jeweiligen Problem abhängt. Allgemein sollte das Verschieben eines Nachfrageobjekts zu einem Angebotsobjekt die beiden einander zuteilen, bzw. das Verschieben eines Markierungsobjekts einer Zuteilung die alte Zuteilung aufheben und eine neue Zuteilung erzeugen. Bei der Dienstplanung für das Pflegepersonal eines Krankenhauses würde das Verschieben eines Nachtschichtsymbols von der Pflegekraft A zu der Pflegekraft B dann folgende intuitiv verständliche Bedeutung haben: Mache die Zuteilung von A zu der Nachtschicht rückgängig und teile statt dessen die Pflegekraft B für diese Nachtschicht ein.

3.3.3 Weitere Maßnahmen zur Unterstützung der interaktiven Planung

Aus dem interaktiven Ansatz ergibt sich zusätzlich zu den Grundanforderungen an die interaktive Funktionalität eine Reihe von zusätzlichen Möglichkeiten, die interaktive Planung zu unterstützen. Zwei exemplarische Möglichkeiten sollen nun vorgestellt werden, die jeweils nur auf expliziten Wunsch des Benutzers zum Einsatz kommen sollten.

3.3.3.1 Darstellung der Existenz von Randbedingungen

Eine Visualisierung von Randbedingungen – also nicht von Verstößen gegen Randbedingungen – könnte einem Benutzer bei der interaktiven Planung dabei helfen,

¹³ Eine solche Manipulation fällt auch teilweise in die Interaktionskategorie „Form fill-in“, da für direkte Manipulation primär nur Zeigeeinstrumente benutzt werden sollte [Shneiderman 98]. Diese Unterscheidung ist für die Praxis jedoch nicht von Bedeutung.

Regelverstöße von vornherein zu vermeiden. Bei der Dienstplanung im Krankenhaus könnte man zum Beispiel all die Tage kennzeichnen, an denen eine Pflegekraft nicht arbeiten will, so daß ein Planer eine entsprechende Zuteilung gar nicht erst durchführt. Allgemeiner bieten sich differenziert nach dem Typ der Randbedingung folgende Möglichkeiten:

Vermeidung einzelner Zuteilungen:

Die mit dieser Zuteilung identifizierten Stelle der Zuordnungsdarstellung kann als problematisch gekennzeichnet werden (Abbildung 30).

Durchsetzung einzelner Zuteilungen

Die erwünschte Zuteilung kann – als solche besonders gekennzeichnet – in die Zuordnungsdarstellung eingezeichnet werden (Abbildung 30).

Vermeidung von Zuteilungskombinationen

Solche Randbedingungen sind nur schwer zu visualisieren; eine derartige Darstellung kann nur dynamisch erfolgen – nur einzelne Zuteilungen, die zusammen mit bereits durchgeführten Zuteilungen zu einer unerlaubten Kombination führen würden, können entsprechend gekennzeichnet werden. Die Visualisierung ganzer Kombinationen ist nicht sinnvoll, da es meist extrem viele solcher Kombinationen gibt. Bei der Dienstplanung im Krankenhaus besagt eine Regel, daß eine Pflegekraft keine 3 Wochenenden in Folge Dienst haben sollte. Betrachtet man nur 3 zusammenhängende Wochenenden, so gibt es allein 27 Kombinationen von Tagen dieser 3 Wochenenden, die zu einer Verletzung dieser Regel führen könnten: Dienst nur an allen Samstagen, Dienst nur an allen Sonntagen, Dienst an allen Samstagen und allen Sonntagen usw. Bezieht man die Schichten mit ein, so erreicht man bereits bei dieser vergleichsweise einfachen Regel schnell astronomisch hohe Werte. Somit bleibt nur die beschriebene dynamische Kennzeichnung einzelner Zuteilungen; wurde die Pflegekraft bereits zum Dienst an zwei der drei Wochenenden eingeteilt, so kann dann jede mögliche einzelne Zuteilung für das dritte, noch freie Wochenende wie bei der Vermeidung einzelner Zuteilungen gekennzeichnet werden.

Durchsetzung von Zuteilungskombinationen

Die erwünschten Zuteilungen können – als solche besonders gekennzeichnet – in die Zuordnungsdarstellung eingezeichnet werden. Allerdings ist das aus den in 3.2.2.4 erläuterten Gründen meist nicht eindeutig möglich, nur die letzte fehlende Zuteilung einer durchzusetzenden Kombination läßt sich als erwünscht kennzeichnen. Sollen also zwei Pflegekräfte bei der Dienstplanung im Krankenhaus zur selben Schicht eingeteilt werden, so kann erst nachdem eine der beiden Pflegekräfte zu einer Schicht zugeteilt wurde, die Zuteilung der anderen Pflegekraft zu eben dieser Schicht wie im Falle der Durchsetzung einer einzelnen Zuteilung besonders gekennzeichnet werden. Eventuell können noch Objekte, die von einer solchen Regel betroffen sind (gekoppelt sind), als zusammengehörig gekennzeichnet werden, zum Beispiel durch einen Kreis um ihre Kodierungen wie in Abbildung 30. Im Beispielsfall wären das die beiden Pflegekräfte.

Arithmetischer Ziel- oder Grenzwert

Der arithmetische Zielwert, auf den sich die Regel bezieht, kann ständig/auf expliziten Wunsch des Benutzers überall/an vom Benutzer ausgewählten Stellen in die Zuordnungsdarstellung integriert werden.

Reihenfolge

Die richtige Reihenfolge wird wie in Abbildung 24 eingeblendet.

Probleme der Darstellung von Randbedingungen

Allerdings treten bei komplizierteren Zuordnungsdarstellungen eventuell Komplikationen ein, etwa bei der Gruppendarstellung – schließlich steht eine Zelle nun für mehrere mögliche Zuteilungen, es muß zusätzlich visualisiert werden, auf welche der in Frage kommenden Zuteilungen die Visualisierung der Randbedingung bezogen ist. In diesem Fall wird die Darstellung von Regeln schnell unübersichtlich.

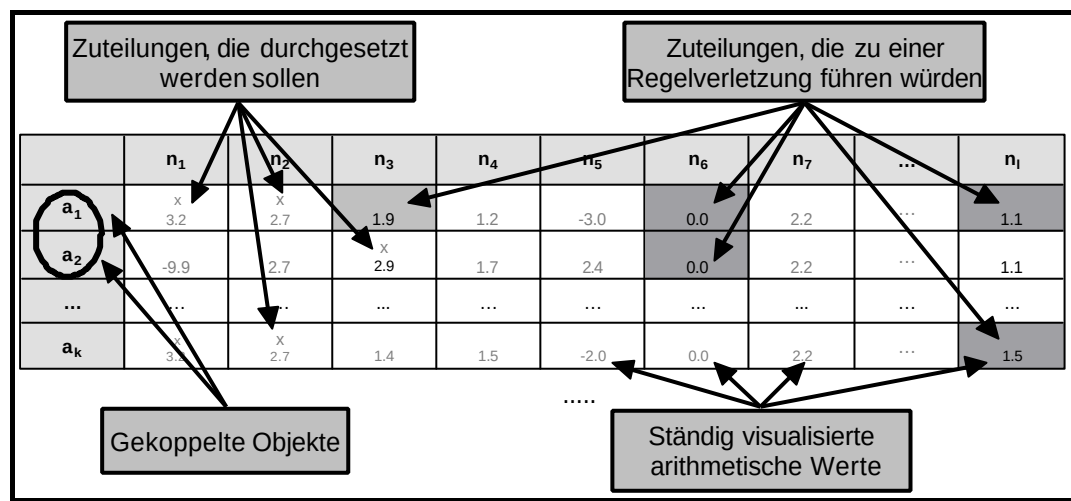


Abbildung 30: Visualisierung der Existenz von Randbedingungen

3.3.3.2 Darstellung von Veränderungen

In manchen Situationen, etwa wenn das System eine vom Benutzer erstellte Zuordnung automatisch optimiert, kann es für den Benutzer hilfreich sein, die gemachten Änderungen zu visualisieren. Prinzipiell sind hier wieder obige Darstellungsmittel anwendbar, so können alle geänderten Zuteilungen farbig markiert oder mit einem Icon versehen werden. Besonders geeignet sind hier jedoch Pfeile, die exakt anzeigen, wie sich einzelne Zuteilungen verändert haben (siehe Abb. 31). Dabei können Veränderungen

natürlich – außer bei der Mehrfachzuordnung – immer nur innerhalb einer Spalte (oder Zeile, je nach Wahl der Achsen) auftreten, bei der Gruppendarstellung zusätzlich noch innerhalb einer Zelle. Im Falle der Mehrfachzuordnung können Veränderungen über die gesamte Darstellung der Zuordnung hinweg auftreten, so daß die Darstellung von Veränderungen schnell unübersichtlich wird.

	n_1	n_2	n_3	n_4	n_5	n_6	n_7	...	n_l
a_1	x ↑			x ↑			x	...	x ↑
a_2					↓			...	
...	...	...	...	...	↓	...	...	...	...
a_k		x			x ↓			...	

Abbildung 31: Visualisierung von Veränderungen der Zuordnung

3.3.4 Fazit

Die Grundanforderungen an die Oberfläche bezüglich der Unterstützung der interaktiven Lösung sind somit, dem Anwender über die verschiedenen Interaktionsformen die in 3.3.1 aufgeführten Eingriffsmöglichkeiten zu bieten. Zusätzlich sollte die Oberfläche die aufgeführten Beispiele zur Unterstützung der interaktiven Planung implementieren.

4 Beschreibung einzelner Zuordnungsprobleme

In diesem Kapitel sollen verschiedene reale Zuordnungsprobleme beschrieben werden und insbesondere ihre individuellen Ansprüche an eine graphische Benutzeroberfläche analysiert werden. Dabei soll für jedes der drei untersuchten Zuordnungsprobleme, der Dienstplanung für das Pflegepersonal im Krankenhaus, der Dienstplanung für das ärztliche Personal im Krankenhaus und der Schulstundenplanung, beschrieben und untersucht werden, inwiefern sich ihre Anforderungen auf die im vorhergehenden Kapitel beschriebenen Grundtypen zurückführen lassen. Der Schwerpunkt liegt dabei wieder auf der Darstellung der Zuordnung und der Visualisierung von Restriktionsverletzungen.

4.1 Dienstplanung für das Pflegepersonal des Juliusspitals Würzburg

4.1.1 Beschreibung der Dienstplanung

4.1.1.1 Aufgabe der Dienstplanung

Aufgabe der Dienstplanung des Juliusspitals ist es, für jeden Tag eines 4-wöchigen Planungszeitraums in den einzelnen Stationen des Krankenhauses bestimmte Schichten entsprechend ihrer Anforderungen mit Pflegepersonal zu besetzen.

4.1.1.2 Pflegepersonal

Insgesamt haben die Stationen einen festen Personalstand in der Größenordnung von 10 bis zu 50 Pflegekräften (zum Beispiel zirka 40 Pflegekräfte auf der Intensivstation), wobei dieser natürlich gewissen Fluktuationen unterworfen ist, etwa aufgrund von Kündigungen oder Mutterschutzurlaub. Im allgemeinen findet zwischen den einzelnen Stationen kein Austausch von Personal statt, so daß auch nur mit den Pflegekräften der Station geplant wird. Ein Teil des Pflegepersonals arbeitet Vollzeit (gegenwärtig 38.5 Stunden pro Woche), der Rest Teilzeit unterschiedlicher Ausprägung (zum Beispiel 20.0 Stunden pro Woche).

Pflegekräfte können unterschiedlich qualifiziert sein und werden gemäß ihrer Ausbildung in Kategorien eingeteilt, wobei Anzahl und Bezeichnung dieser Kategorien von Station zu Station verschieden sein können. So wird auf der Intensivstation zwischen den Kategorien I, II, III, Nicht – Examinierte (NE) und dem Stationsleiter (SL), bzw. seinem Stellvertreter (SSL) unterschieden.

4.1.1.3 Schichten

Auf jeder Station gibt es verschiedene Schichtarten, die durch ihren Namen, ihre Anfangszeit, ihre Endzeit, und die enthaltene Pause charakterisiert sind. Um Platz zu sparen, werden sie im Dienstplan nur mit Kürzeln bezeichnet. In der Intensivpflege handelt es sich dabei zum Beispiel um folgende sechs Schichtarten:

?? Die Frühschicht 1 (Kürzel F1) von 6:00 bis 14:12 (mit ½ Stunde Pause).

- ?? Die Frühschicht 2 (Kürzel F2) von 7:00 bis 15:12 (mit ½ Stunde Pause).
- ?? Die Zwischenschicht (Kürzel Z) von 7.42 bis 15:54 (mit ½ Stunde Pause).
- ?? Die Spätschicht 1 (Kürzel S1) von 11:48 bis 20:00 (mit ½ Stunde Pause).
- ?? Die Spätschicht 2 (Kürzel S2) von 13:30 bis 21:42 (mit ½ Stunde Pause).
- ?? Die Nachtschicht (Kürzel N) von 21:06 bis 6:36 (mit ½ Stunde Pause).

Die Schichten müssen jeden Tag mit der geforderten Anzahl ausreichend qualifizierten Personals besetzt werden. Dabei wird der Personalbedarf mittels der beschriebenen stationsspezifischen Kategorien formuliert; auf der Intensivstation sind typischerweise für jeden Tag eines Planungszeitraums folgende Schichten zu besetzen, wobei der konkrete Bedarf jedoch prinzipiell von Tag zu Tag variieren kann:

Schichtname	Schichtleiter	Kategorie I	Kategorie II	Kategorie III	Nicht-Examierte
Frühschicht 1		1	1	2 oder	
				1	1
Zwischen-schicht	Wochentags 1				
Spätschicht 2		1	1	2 oder	
				1	1
Nachtschicht		1	2		

Tabelle 6: Typischer Personalbedarf der Intensivstation

4.1.1.4 Urlaubsplanung

In den Dienstplan geht der Urlaubsplan mit ein, der im allgemeinen vor dem Dienstplan festgelegt wird. Die einzige Konsequenz für den Dienstplan ist, daß Pflegekräfte, die sich im Urlaub befinden, zwar zu bestimmten Schichten eingeteilt, aber nicht zur Deckung des Personalbedarfs der Schicht hinzugerechnet werden.

4.1.1.5 Randbedingungen

Bei der Erstellung des Dienstplanes müssen bestimmte Randbedingungen berücksichtigt werden, die sich teils aus gesetzlichen Vorschriften, teils aus klinik- und stationsinternen Anforderungen ergeben.

4.1.1.5.1 Allgemeine Richtlinien

Folgende allgemeinen Regeln sollten beachtet werden:

- ?? Der Personalanforderungen jeder Schicht müssen erfüllt werden.

- ?? Der Dienstplan einzelner Pflegekräfte sollte auf Wunsch möglichst weitgehend einer vorgegebenen Schichtenfolge entsprechen.
- ?? Gewisse Pflegekräfte sollten immer zu denselben Schichten eingeteilt werden, also immer zusammen arbeiten.
- ?? Freie Tage sollten möglichst zusammenhängend sein.
- ?? Häufige Schichtwechsel sollten vermieden werden.

4.1.1.5.2 Regeln bezüglich der Arbeitszeit des Pflegepersonals

Folgende Regeln, die alle die Arbeitszeit der Pflegekräfte betreffen, sollten beachtet werden:

Allgemeine Regeln

- ?? Die Differenz der tatsächlichen Arbeitszeit und der Sollarbeitszeit (154 Stunden für eine volle Planstelle innerhalb von 4 Wochen) innerhalb eines Planungszeitraumes sollte höchstens 15 Stunden betragen und möglichst gering sein. Dabei muß auch der Übertrag aus dem letzten Dienstplan berücksichtigt werden.
- ?? Auf zwei Dienstwochenenden sollte immer ein freies Wochenende folgen.
- ?? Eine Pflegekraft kann höchstens zehn Tage in Folge arbeiten.
- ?? Eine Pflegekraft nach einem Dienstblock, also mehreren direkt aufeinander folgenden Arbeitstagen, mindestens zwei Tage frei haben.
- ?? Eine Pflegekraft sollte immer mindestens 2, höchstens 5 Nachtdienste in Folge arbeiten.
- ?? Zwischen zwei Schichten einer Pflegekraft müssen mindestens 10 Stunden Pause liegen. Somit ist z.B. die Konstellation Spätdienst 2 – nächster Frühdienst 1 nicht zulässig, ebenso wie die Kombination zweier direkt aufeinanderfolgender Schichten. Falls sich die Kombination Spätdienst 2 – Frühdienst 1 nicht vermeiden läßt, kann eine Pflegekraft ihre Spätschicht um 1½ Stunden vorziehen (also die Spätschicht 1 arbeiten).
- ?? Die Wünsche und besonderen Umstände des Pflegepersonals sind bei der Dienstplanung so weit wie möglich zu berücksichtigen, wobei verschiedene existieren (zum Beispiel „nicht“, „ungern“, „gern“, „unbedingt“). Insgesamt gibt es folgende Grundtypen, die sowohl einzelne Pflegekräfte als auch Gruppen von Pflegekräften betreffen können:
 - Grundsätzliche Bevorzugung/Vermeidung einer bestimmten Schicht (nicht/ungern/gerne/unbedingt Frühschicht), eines bestimmten Wochentags („nicht/ungern/gerne/unbedingt montags“) oder einer Kombination Wochentag-Schicht für den gesamten Planungszeitraum („nicht/ungern/gerne/unbedingt die Spätschicht am Freitag“)
 - Bevorzugung/Vermeidung konkreter Schichten oder Tage im Planungszeitraum. („nicht/ungern/gerne/unbedingt Dienstag, 03.11.98 Frühschicht“, „nicht/ungern/gerne/unbedingt Donnerstag, 05.11.98“)

Spezielle Regeln nach dem Mutterschutzgesetz

Für all die Pflegekräfte, die das Mutterschutzgesetz betrifft, gelten zusätzlich folgende Regelungen:

- ?? Ein Einsatz im Nachtdienst (zwischen 20 Uhr und 6 Uhr) ist nicht zulässig.
- ?? Werdende und stillende Mütter dürfen nicht mit Mehrarbeit beschäftigt werden. Mehrarbeit ist jede Arbeit, die von Frauen unter 18 Jahren über 8 Stunden täglich oder 80 Stunden in der Doppelwoche, und von Frauen ab dem 18. Lebensjahr über 8 ½ Stunden täglich oder 90 Stunden in der Doppelwoche geleistet wird.

Spezielle Regeln nach dem Jugendarbeitsschutzgesetz

Für all die Pflegekräfte, die das 18. Lebensjahr zum Zeitpunkt der jeweiligen Schichten noch nicht vollendet haben, gelten zusätzlich folgende Regelungen:

- ?? Die wöchentliche Arbeitszeit darf 40 Stunden nicht überschreiten.
- ?? Die tägliche Arbeitszeit ohne Pausen darf acht Stunden nicht überschreiten.
- ?? Die tägliche Arbeitszeit muß zwischen 7 und 20 Uhr absolviert werden.
- ?? Es darf nur an maximal fünf Tagen in der Woche gearbeitet werden
- ?? Zwei Sonntage im Monat müssen beschäftigungsfrei bleiben.
- ?? Die Beschäftigung von Jugendlichen an den Feiertagen 25. Dezember, 1. Januar, erster Osterfeiertag und 1. Mai, sowie am 24. Und am 31. Dezember nach 14 Uhr ist ausgeschlossen.

4.1.1.6 Dienstplanung als Zuordnungsproblem

Die Dienstplanung kann als diskretes Zuordnungsproblem aufgefaßt werden, wobei die Pflegekräfte als Anbieter und die einzelnen Schichten als Nachfrager fungieren. Es handelt sich hier um eine $n \times m$ Zuordnung, da eine Pflegekraft, also ein Angebotsobjekt, mehreren Nachfrageobjekten, den Schichten, zugeteilt werden kann und eine Schicht gleichzeitig einen Personalbedarf von mehreren Pflegekräften hat. Jedoch kann dieses Problem, wie in Kapitel 2 beschrieben, in eine $1 \times n$ Zuordnung umgewandelt werden, indem man einzelnen Schichten mit einem Personalbedarf von n Pflegekräften in n unterschiedliche Arbeitsplätze mit einem Personalbedarf von einer einzelnen Pflegekraft umwandelt. Dies spielt jedoch für die Darstellung keine Rolle, der Benutzer wird im allgemeinen die Schichten als Nachfrageobjekte identifizieren und sollte mit der internen Repräsentation des Zuordnungsproblems nicht konfrontiert werden.

4.1.2 Traditionelle Darstellung

Die manuelle Planung geschieht durch den Stationsleiter per Hand für einen Zeitraum von vier Wochen. Tabelle 7 zeigt ein Beispiel eines ausgefüllten Formulars. Bei jeder Pflegekraft wird für jeden Tag das Kürzel der eingeteilten Schicht eingetragen, bzw. ein leerer Kreis (O) für einen dienstfreien Tag.

Die Bedeutung der einzelnen Spalten und Zeilen ist dabei:

- ?? Die linke Übertragsspalte (1) führt den Übertrag aus dem letzten Planungszeitraum.
- ?? Die rechte Übertragsspalte (2) führt den neuen Übertrag, der dann in den nächsten Planungszeitraum übernommen wird.
- ?? Die Spalte „Wochenstunden“ führt die Arbeitszeit der vergangenen Woche.
- ?? „Soll-Stunden“ gibt an, wie viele Stunden die Pflegekraft im Planungszeitraum arbeiten soll.
- ?? „Ist-Stunden“ gibt an, wie viele Stunden die Pflegekraft im Planungszeitraum tatsächlich arbeitet, inklusive des Übertrags aus dem letzten Planungszeitraum.
- ?? Die zwei meist leeren Zeilen (3) unter der Zeile mit den Schichteinträgen dienen dazu, Tage als Urlaub zu markieren oder sonstige Anmerkungen zu machen.
- ?? Die auf einzelne Schichten bezogene Zeile (4) gibt im Format x/y an, wie viele Pflegekräfte der Schicht an diesem Tag tatsächlich zugeteilt sind (x) und wie viele zugeteilt sein sollten (y). Nach Qualifikationsstufen wird hier nicht differenziert.

Dienstplan Intensivstation 8.3.99 - 28.3.99

Name	Aus- bildung	Über- trag	Mo	Di	Mi	Do	Fr	Sa	So	Wo- Std.	Mo	Di	Mi	Do	Fr	Sa	So	Wo- Std.	...	So	Wo- Std.	Soll- Std.	Ges- Std.	Über- trag
			8	9	10	11	12	13	14		15	16	17	18	19	20	21		...	28				
Heinz M.	SL	28.8	Z	Z	Z	Z	Z	S	S	53.9	Z	Z	Z	Z	O	O	O	30.8	...	O	38.5	115.5	123.2	
Dieter K.	SL	20.7	S	S	S	N	N	N	N	53.0	N	N	N	O	F	O	O	13.8	...	S	53.9	115.5	120.7	
Helga A.	I	34	O	O	O	N	N	N	N	29.9	O	O	O	O	O	F	F	21.1	...	N	20.9	60	72.3	
Jörg S.	I	56	O	S	S	S	S	S	S1	46.2	O	O	F	F	F	O	O	23.1	...	S	38.5	115.5	113.9	
				U	U	U	U	U	U				U	U	U									
Heide L.	II	21.5	N	N/0	O	O	S		F	38.2	F	F	F	F	O	F	F	46.2	...	F	23.1	115.5	107.7	
3 ... 4																								
F			4/4	4/4	4/4	4/4	4/4	4/4	4/4		4/4	4/4	4/4	4/4	4/4	4/4	4/4			4/4				
Z			1/1	1/1	1/1	1/1	1/1	0/0	0/0		1/1	1/1	1/1	1/1	1/1	0/0	0/0			0/0				
S1			1/1	1/1	1/1	1/1	1/1	1/1	1/1		0/0	0/0	1/1	1/1	1/1	1/1	0/0			0/0				
S2			3/3	3/3	4/3	3/3	3/3	3/3	3/3		4/4	4/4	3/3	3/3	3/3	3/3	4/4			4/4				
N			3/3	3/3	3/3	3/3	3/3	3/3	3/3		3/3	3/3	3/3	3/3	3/3	3/3	3/3			3/3				

Tabelle 7: Formular der Dienstplanung im Krankenhaus

4.1.3 Einordnung in die Grundtypen

4.1.3.1 Zuordnungsdarstellung

Die Zuordnungsdarstellung der Dienstplanung des Pflegepersonals im Krankenhaus entspricht einer injektiven Gruppendarstellung mit Attributen. Jede Gruppe besteht dabei aus den Schichten (genauer den Arbeitsplätzen) eines Tages, wobei die Anfangszeit einer Schicht die Zuordnung eines Tages zu einer Schicht bestimmt, eine Nachtschicht zum Beispiel wird dem Tag zugeordnet, an dem sie beginnt.

Für die Angebotsobjekte, die Pflegekräfte, werden die Attribute Name, Ausbildung, Übertrag aus dem letzten Planungszeitraum, die Ist-Stunden für jede einzelne Woche des Planungszeitraums, die Soll-Stunden, die Ist-Stunden des gesamten Planungszeitraums und der Übertrag in den nächsten Planungszeitraum visualisiert. Für die Nachfrageobjekte, die Schichten, wird nur das Attribut Ist-Besetzung/ Soll-Besetzung dargestellt.

4.1.3.2 Randbedingungen

In Tabelle 8 sollen nun die Randbedingung der Dienstplanung in die in Kapitel 2 vorgestellten Grundtypen eingeordnet werden.

Randbedingung	Typ	Zuteilungs-kombinationen	Gemeinsames Objekt
Ist-Zeit möglichst nahe an Soll-Zeit	Arithmetischer Zielwert	Ja	Pflegekraft
Personalbedarf einer Schicht muß erfüllt werden	Arithmetischer Zielwert	Ja	Schicht
Mindestens 10 Stunden Pause zwischen zwei Schichten	Arithmetischer Grenzwert	Ja	Pflegekraft
Keine Mehrarbeit (Mutterschutz)	Arithmetischer Grenzwert	Ja	Pflegekraft
Wöchentliche Arbeitszeit < 40h (Jugendschutz)	Arithmetischer Grenzwert	Ja	Pflegekraft
Tägliche Arbeitszeit ohne Pausen < 8h (Jugendschutz)	Arithmetischer Grenzwert	Nein	-
Koppelung von Pflegekräften	Durchsetzung	Ja	Schicht
Nach einem Dienstblock zwei Tage frei	Durchsetzung	Ja	Pflegekraft
Qualifikation muß stimmen	Durchsetzung	Ja	Schicht
Dienstpräferenzen des Pflegepersonals beachten	Durchsetzung/ Vermeidung	Nein	-
Vorgegebene Schichtfolge	Muster/ Reihenfolge	Ja	Pflegekraft
Mindestens 2 (keine einzelnen), höchstens 5 Nachtschichten in Folge (keine 6)	Vermeidung	Ja	Pflegekraft

Höchstens 10 Tage in Folge Dienst	Vermeidung	Ja	Pflegekraft
Kein Nachtdienst (Mutterschutz)	Vermeidung	Nein	-
Tägliche Arbeitszeit zwischen 7 und 20 Uhr (Jugendschutz.)	Vermeidung	Nein	-
Maximal fünf Arbeitstage pro Woche (Jugendschutz)	Vermeidung	Ja	Pflegekraft
Zwei Sonntage im Monat müssen beschäftigungsfrei bleiben.	Vermeidung	Ja	Pflegekraft
Kein Dienst am 25. 12, 1. 1, erster Osterfeiertag und 1. 5, sowie am 24. Und 31. 12 nach 14 Uhr (Jugendschutz)	Vermeidung	Nein	Pflegekraft
Keine häufigen Schichtwechsel	Vermeidung	Ja	Pflegekraft
Freie Tage möglichst zusammenhängend	Vermeidung	Ja	Pflegekraft

Tabelle 8: Einordnung der Randbedingungen

4.1.4 Beispiele für die resultierende Darstellung

In diesem Abschnitt soll an einem Beispiel die Visualisierung gezeigt werden, die aus der Einordnung in die Grundtypen resultiert. Bei der Zuordnungsdarstellung ist diese Einordnung problemlos möglich, somit müssen bei der resultierenden Darstellung keine Abstriche gemacht werden, sie entspricht genau dem Beispielformular.

Der einzige Unterschied bei der Darstellung resultiert aus der Einordnung der Randbedingungen; die Regel, daß die Ist-Zeit einer Pflegekraft möglichst der Soll-Zeit entsprechen sollte, ist eindeutig vom Typ Ziel- oder Grenzwert. Somit bietet es sich an, die Ist-Stunden und Soll-Stunden in einer Anzeige zusammenzufassen und über ein Balkendiagramm zu visualisieren, gemäß dem Grundsatz, daß arithmetische Ziel- oder Grenzwerte graphisch und nicht durch Ziffern dargestellt werden, solange nicht absolut genaue Werte benötigt werden (siehe 3.3).

Tabelle 9 zeigt die unterschiedlichen Visualisierungen einer komplexen Restriktionsverletzung; dabei werden, wie in 3.2 diskutiert, die Zuteilungen, die zu dieser Verletzung führen, besonders gekennzeichnet. Dazu werden Symbole, Hintergrundfarbe, Schraffierung, Icon und Pfeile als Kodierungsformen eingesetzt.

Name	Aus- bildung	Über- trag	Mo 15	Di 16	Mi 17	Do 18	Fr 19	Sa 20	So 21	...	So 28	Wo- Std	Soll-Std	Über- trag
Heinz M.	FKP SL	28.8	Z*	Z	Z*	Z*	O	O	O	...	O	38.5	117	
													110	
Dieter K.	FKP SL	20.7	N	N	N	O	F	O	O	...	S	53.9		
Helga A.	KS FWBI	34	0	O	O	O	O						60	
Jörg S.	FKS	56	O	O	F	F	F	O	O	...	S	38.5	115	
Heide L.	KP	21.5	F	F	F	F	O	F	F	...	F	23.1		
...														

Tabelle 9: Beispiel für die resultierende Visualisierung der Dienstplanung für das Pflegepersonal des Juliusspitals

4.1.5 Fazit

Die Dienstplanung im Krankenhaus lässt sich komplett in das System der Grundtypen einordnen. Die resultierende Darstellung scheint brauchbar, wie das Beispiel zeigt. Allerdings hat ein solches Beispiel prinzipiell nur eine sehr begrenzte Aussagekraft, unter anderem weil nur ein kleiner schwarz-weißer Ausschnitt des Dienstplans und wenige Restriktionsverletzungen gezeigt werden können. Sinnvolle Aussagen über die Qualität der resultierenden Darstellung, bzw. der gesamten Oberfläche können nur mittels Untersuchungen des praktischen Einsatzes der erzeugten Oberfläche durch Endbenutzer durchgeführt werden. Deswegen soll hier in Bezug auf die Qualität der resultierenden Darstellung nur auf Kapitel 6 verwiesen werden, das eine solche Evaluation beschreibt.

4.2 Dienstplanung für das Pflegepersonal der anästhesiologischen Klinik der Universität Würzburg

Zusätzlich zur Dienstplanung im Juliusspital wurde im Rahmen dieser Arbeit auch die Dienstplanung für das Pflegepersonal der anästhesiologischen Klinik der Universität Würzburg untersucht. Auch wenn es sich dabei um zwei vollkommen unabhängige Kliniken mit unterschiedlichen Trägern handelt, wurde eine weitgehende Übereinstimmung in Fragen der Dienstplanung gefunden. Deswegen soll hier nur kurz auf die Unterschiede zwischen den Dienstplanungen eingegangen werden und untersucht werden, ob sie in Hinblick auf die Oberfläche eines Werkzeuges zur Dienstplanung relevant sind.

4.2.1 Unterschiede zur Dienstplanung des Juliusspitals

4.2.1.1 Aufgabe der Dienstplanung

Der Planungszeitraum an der anästhesiologischen Klinik beträgt 6 Wochen, im Juliusspital nur 4 Wochen.

4.2.1.2 Pflegepersonal

Generell in der anästhesiologischen Klinik alle Pflegekräfte gleich qualifiziert, jedoch kommt nur ein Teil des Personals für die Tätigkeit als Schichtleiter in Frage; das Juliusspital unterscheidet zwischen einer Reihe von Qualifikationsstufen.

4.2.1.3 Schichten

Unterschiedlich ist die Zahl und die Dauer der einzelnen Schichten. In der anästhesiologischen Klinik gibt es folgende Schichttypen:

- ?? Die Frühschicht (Kürzel F) von 6:30 bis 14:45 (mit ½ Stunde Pause) mit mindestens 7 Pflegekräften. Jeden zweiten Dienstag im Monat findet eine Stationsbesprechung statt, an der die Frühschicht teilnimmt und somit eine Stunde länger arbeitet.
- ?? Die Spätschicht (Kürzel S) von 14:00 bis 22:15 (mit ½ Stunde Pause) mit mindestens 7 Pflegekräften.
- ?? Die Nachtschicht (Kürzel N) von 21:30 bis 7:00 (mit ½ Stunde Pause) mit mindestens 6 Pflegekräften.
- ?? Der Funktionsdienst von 7.30 bis 15:45 (mit ½ Stunde Pause) mit genau einer Pflegekraft, einem/einer Schichtleiter/in

4.2.1.4 Randbedingungen

Folgende Restriktionen unterscheiden sich oder kommen neu dazu:

- ?? Die Differenz der tatsächlichen Arbeitszeit und der Sollarbeitszeit (231.0 Stunden für eine volle Planstelle innerhalb von 6 Wochen) innerhalb eines Planungszeitraumes darf höchstens 6 Stunden betragen (Juliusspital 15 Stunden).
- ?? Jede Schicht muß ein Schichtleiter anwesend sein (neu).
- ?? Auf ein Dienstwochenende sollte immer ein freies Wochenende folgen, falls notwendig kann aber auch zwei Wochenenden hintereinander gearbeitet werden, auf die dann zwei freie Wochenenden folgen müssen (Juliusspital:.. Zwei Arbeitswochenenden, ein freies Wochenende).

4.2.2 Traditionelle Darstellung

Die benutzten Formulare unterscheiden sich nur in folgenden Details:

- ?? Die Ist-Arbeitszeit wird in der Anästhesiologie nicht für jede Woche angezeigt.
- ?? Das U für Urlaub wird in der Anästhesiologie farbig (rot) gekennzeichnet.

4.2.3 Fazit

Die Dienstplanung der beiden Kliniken unterscheidet sich nur in Details, prinzipiell kann dieselbe Darstellung und damit dieselbe Oberfläche für beide verwendet werden. Da die beiden Kliniken in keiner Weise affiliert sind, besteht die berechtigte Hoffnung, daß sich dieses Resultat auf weitere Kliniken übertragen läßt. Allerdings könnte eine solche Behauptung nur durch eine umfassende Untersuchung bewiesen werden, die über den Rahmen dieser Arbeit hinausgeht.

4.3 Schulstundenplanung:

4.3.1 Beschreibung der Schulstundenplanung

4.3.1.1 Aufgabe der Schulstundenplanung

Für jede Jahrgangsstufe eines Schultyps schreibt der vom Kultusministerium des jeweiligen Bundeslandes festgelegte Lehrplan vor, welches Fach wie viele Stunden pro Woche unterrichtet werden soll (die Unterrichtsplanung gilt im allgemeinen für eine Woche und wiederholt sich dann). Anhand des Lehrplanes und der Qualifikation, der Verfügbarkeit und der persönlichen Wünsche der einzelnen Lehrer legt dann die Lehrerkonferenz einer Schule in der sogenannten Deputatsliste fest, welcher Lehrer welche Klasse in welchem Fach unterrichtet. Ein Eintrag in der Deputatsliste (z.B. Lehrer: Müller, Klasse: 5a, Fach: Deutsch) heißt Kurs. Eine Wochenstunde ist ein Zeitintervall an einem bestimmten Wochentag, etwa Montag, 10-11 Uhr. Eine Unterrichtsveranstaltung besteht aus einem Kurs und der im Lehrplan vorgeschriebenen Anzahl der Wochenstunden (z.B. Lehrer: Müller, Klasse: 5a, Fach: Deutsch, 5 Stunden), eine Unterrichtsstunde ist eine konkrete Stunde einer Unterrichtsveranstaltung (z.B. Lehrer: Müller, Klasse: 5a, Fach: Deutsch, erste von 5 Wochenstunden). Bereits die Erstellung der Deputatsliste ist ein Zuordnungsproblem, das jedoch im allgemeinen nicht komplex ist und daher besser ohne Computereinsatz gelöst wird. Im Rahmen dieser Diskussion wird die Deputatsliste daher als gegeben betrachtet.

Das eigentliche Problem der Schulstundenplanung ist es, den Unterrichtsstunden als Nachfrageobjekten sowohl Räume als auch Wochenstunden als Angebotsobjekte zuzuweisen. Dabei muß eine Vielzahl von Randbedingungen beachtet werden, auf die bei der Beschreibung der einzelnen Objekte der Zuordnung und in einem gesonderten Abschnitt eingegangen werden soll. Eine ausführliche Darstellung des Problems Schulstundenplanung mit vielen Beispielen findet man bei [Busch 97].

4.3.1.2 Wochenstunden

Im Rahmen der Schulstundenplanung ist eine Wochenstunde ein Zeitintervall an einem bestimmten Wochentag, etwa Montag, 10.00 - 10.45 Uhr. Ein Wochentag wiederum ist ein Tag des Planungszeitraumes, oft Montag bis Freitag für einen Planungszeitraum von fünf Tagen. Allerdings sind auch beliebige andere Planungszeiträume denkbar.

Wochenstunden eines Tages überschneiden sich nicht, somit kann man sie eindeutig nach ihrer Anfangszeit ordnen und dann kurz mit der entsprechenden Ordnungszahl identifizieren, zum Beispiel die Stunde mit der frühesten Anfangszeit am Montag mit Montag, 1. Stunde. Im allgemeinen sind alle Schulstunden gleich lang; der Begriff Schulstunde ist insoweit irreführend, als eine Schulstunde keineswegs 60 Minuten dauern muß. Üblich sind 45 Minuten, aber auch andere Längen sind vorstellbar. Meistens ist die Aufteilung der einzelnen Wochentage in Wochenstunden identisch, in diesem Fall ist zum Beispiel die 5. Stunde an allen Wochentagen das Zeitintervall von 11.30 bis 12.15 Uhr. Eine Doppelstunde besteht aus zwei unmittelbar aufeinanderfolgenden Wochenstunden.

Schulpausen werden meistens implizit angegeben, indem zwei Stunden zwar gemäß obiger Ordnung direkt aufeinanderfolgen, aber dennoch eine Pause zwischen Ende der früheren und Anfang der späteren Stunde liegt. Als Alternative kann man sie auch

explizit als spezielle Wochenstunden aufführen, dann haben allerdings nicht mehr alle Wochenstunden dieselbe Dauer.

4.3.1.3 Räume

Unterrichtsstunden finden in Räumen statt, wobei nicht jeder Raum für jedes Fach gleich gut geeignet ist; so sollte Sport in der Turnhalle stattfinden, Physik im Physiksaal. Klassen haben oft ihren eigenen Stammraum, in dem sie dann bevorzugt Unterricht haben. Bei Räumen muß eventuell auch die Entfernung vom Hauptgebäude der Schule berücksichtigt werden; Unterrichtsstunden in weiter entfernten Räumen sollten nicht unmittelbar vor, bzw. anschließend an Unterrichtsstunden in anderen Räumen stattfinden, sondern es sollte eine Pause dazwischen liegen. Räume können im allgemeinen nur eine Klasse auf einmal aufnehmen; Ausnahmen wie zum Beispiel ein Sportplatz, auf dem mehrere Klassen gleichzeitig Unterricht haben können, sollten in entsprechend viele logische Räume untergliedert werden, z.B. Sportplatz1, Sportplatz2 usw. Darüber hinaus sollte mehreren Klassen nur auf expliziten Wunsch gleichzeitig derselbe Raum zugewiesen werden, z.B. wenn der Religionsunterricht nach Konfessionen getrennt wird und Teile von verschiedenen Klassen in einem einzelnen Raum zusammen in einer Konfession unterrichtet werden sollen. Auch Räume können von Zeitpräferenzen betroffen sein, etwa wenn sie nur an manchen Wochentagen zur Verfügung stehen. Die auftretenden Typen dieser Zeitpräferenzen werden noch ausführlicher diskutiert.

4.3.1.4 Lehrer

Lehrer verfügen – wie Pflegekräfte – über unterschiedliche Qualifikationen, wobei diese über die Fächer, die ein Lehrer unterrichtet, ausgedrückt wird. Lehrer können einen Stammraum haben, in dem sie bevorzugt unterrichten, zusätzlich können Lehrer eine Stammklasse haben. Allerdings sind Stammklasse und Qualifikation von Lehrern primär für die Erstellung der Deputatsliste von Interesse und müssen deshalb im Rahmen dieser Diskussion nicht weiter berücksichtigt werden. Lehrer können eine Vielzahl von Zeitpräferenzen haben, die im Rahmen der Diskussion der Randbedingungen detailliert werden.

4.3.1.5 Klassen

Klassen bestehen aus einer festen Gruppe von Schülern, ein Kurssystem mit wechselnder Zusammensetzung der Klassen wie in der gymnasialen Oberstufe soll also im Rahmen dieser Arbeit nicht berücksichtigt werden. Klassen können für manche Unterrichtsveranstaltungen geteilt werden, z.B. findet Sport für Jungen und Mädchen getrennt statt; auch für Klassen können Zeitpräferenzen formuliert werden. In diesem Zusammenhang kann die Zahl der männlichen, bzw. weiblichen Schüler der Klasse von Interesse sein. Klassen können wie Lehrer einen Stammraum haben, in dem sie dann bevorzugt Unterricht haben sollen. Zur leichteren Formulierung von Randbedingungen werden Klassen zu Gruppen zusammengefaßt, zum Beispiel alle 5. Klassen einer Schule zu der 5. Klassenstufe.

4.3.1.6 Fächer

Bei Fächern wird oft zwischen Haupt- und Nebenfächern differenziert, was in Bezug auf einige Randbedingungen wichtig ist. Auch Fächer können Zeitpräferenzen haben, zum Beispiel sollen Hauptfächer oft nicht nachmittags unterrichtet werden. Wie bereits erwähnt brauchen bestimmte Fächer besonders ausgestattete Räume, z.B. Sport oder Chemie.

4.3.1.7 Unterrichtsveranstaltungen

Eine Unterrichtsveranstaltung besteht aus einem Lehrer, einer Klasse, einem Fach und einer Anzahl von Stunden, die besagt, wie viel Stunden diese Unterrichtsveranstaltung in einer Woche gehalten werden soll. Unterrichtsveranstaltungen sollen manchmal in Doppelstunden gehalten werden, was bedeutet, daß zwei der Unterrichtsstunden der Veranstaltung eine Doppelstunde, also zwei direkt aufeinanderfolgende Wochenstunden, zugeteilt bekommen sollen. Auch können Unterrichtsveranstaltungen gekoppelt werden, zum Beispiel soll oft der Unterricht eines Faches für eine ganze Jahrgangsstufe gleichzeitig stattfinden, was auch als Parallelunterricht bezeichnet wird. Dieser Parallelunterricht kann für einzelne Unterrichtsveranstaltungen explizit verboten werden, um Schülern den Besuch beider Veranstaltungen zu ermöglichen.

4.3.1.8 Unterrichtsstunde

Eine Unterrichtsstunde ist eine konkrete Stunde einer Unterrichtsveranstaltung, also z.B. die erste Stunde Latein in der 5. Klasse, gehalten von Lehrer A.

4.3.1.9 Randbedingungen der Stundenplanung

Im folgenden sollen nun die Randbedingungen aufgeführt werden, die bei der Schulstundenplanung vorkommen:

Triviale Randbedingung:

Eine triviale Randbedingung der Schulstundenplanung ist, daß ein Lehrer oder eine Klasse nicht zur selben Zeit an zwei verschiedenen Orten (Räumen) sein kann.

Zeitpräferenzen:

Es gibt eine ganze Reihe von Präferenzen, welche die Zuteilung von Klassen, Lehrern, Fächern zu Wochenstunden und auch die Kombination der beiden Klassen von Angebotsobjekten (Räume und Wochenstunden) beeinflussen. Diese Präferenzen können dann jeweils folgende Zeiten betreffen:

?? Bestimmte Wochenstunden¹⁴ (nicht/ungern/gern/unbedingt). (Raum 103 nicht Mittwochs, 1. Stunde, Lehrer A gerne am Donnerstag oder Hauptfächer nicht am Nachmittag).

¹⁴ Solche Präferenzen werden oft für Gruppen von Wochenstunden eingegeben, was allerdings in Bezug auf Klassifizierung und Visualisierung keine Rolle spielt; solche Gruppierungen dienen nur der leichteren Formulierung von Randbedingungen und sollen deshalb in dieser Diskussion nicht

- ?? Unbestimmte Wochenstunden (Lehrer Müller ist nur zwei Tage in der Woche an der Schule oder die Klasse 5a soll nur 4 Halbtage in der Woche Unterricht haben).

Randbedingungen bezüglich der Zuordnung Unterrichtsstunde – Wochenstunde

Folgende weitere Randbedingungen betreffen die Zuordnung von Unterrichtsstunden zu Wochenstunden, wobei die entsprechenden Parameter beliebig spezifiziert werden können:

- ?? Eine spezifizierte maximale Anzahl von Unterrichtsstunden ohne Pause für Lehrer oder Klassen sollte nicht überschritten werden.
- ?? Eine spezifizierte maximale Anzahl von wöchentlichen Freistunden für Lehrer oder Klassen sollte nicht überschritten werden.
- ?? Eine spezifizierte minimale und maximale Länge der Mittagspause für Lehrer oder Klassen sollte eingehalten werden.
- ?? Eine spezifizierte minimale oder maximale Stundenzahl pro Tag für Lehrer oder Klassen sollte eingehalten werden.
- ?? Eine spezifizierte maximale Zahl von Hauptfächern pro Tag für Klassen sollte nicht überschritten werden.
- ?? Eine spezifizierte maximale Zahl der Hauptfächer in Folge für Klassen sollte nicht überschritten werden.
- ?? Jedes Fach (oder jede Unterrichtsveranstaltung) sollte nur einmal pro Tag in einer Klasse unterrichtet werden, außer im Falle von Doppelstunden.
- ?? Bestimmte Unterrichtsveranstaltungen sollten in Doppelstunden unterrichtet werden.
- ?? Bestimmte Unterrichtsveranstaltungen sollten in einer spezifizierten Reihenfolge unterrichtet werden.
- ?? Bestimmte Unterrichtsveranstaltungen sollten gekoppelt, also ihre jeweiligen Unterrichtsstunden derselben Wochenstunde zugewiesen werden.
- ?? Die Koppelung bestimmter Unterrichtsveranstaltungen sollten vermieden werden, also ihre jeweiligen Unterrichtsstunden nicht derselben Wochenstunde zugewiesen werden.

Randbedingungen bezüglich der Zuordnung Unterrichtsstunde – Raum

Für jeden Raum können folgende Parameter und damit auch Regeln spezifiziert werden:

- ?? Der Raum muß für das entsprechende Fach geeignet sein.
- ?? Ein Raum kann zu jedem Zeitpunkt nur eine Unterrichtsveranstaltung aufnehmen, es sei denn, dies ist explizit erwünscht, dann sollte dieser Wunsch umgesetzt werden.
- ?? Die Kapazität eines Raumes muß eine Klasse, bzw. die Klassen aufnehmen können.

weiter berücksichtigt werden. Ein Tag oder Halbtage wird somit als Gruppe von Wochenstunden interpretiert.

?? Stammräume von Klassen und Lehrer sollten diesen auch bevorzugt zugewiesen werden.

Randbedingungen bezüglich der Zuordnung Unterrichtsstunde – Wochenstunde und zugleich der Zuordnung Unterrichtsstunde – Raum

Unterrichtsstunden in weiter entfernten Räumen sollten nicht unmittelbar vor, bzw. anschließend an Unterrichtsstunden in anderen Räumen stattfinden, sondern es sollte eine Pause dazwischen liegen.

4.3.1.10 Schulstundenplanung als Zuordnungsproblem

Die Schulstundenplanung ist eine Mehrfachzuordnung. Sie besteht zum einen aus der 1-n Zuordnung der Unterrichtsstunden zu den Wochenstunden, zum anderen aus der 1-n Zuordnung der Unterrichtsstunden zu den Räumen. Die Zuordnung der Unterrichtsstunden zu den Wochenstunden wird als primäre Zuordnung bezeichnet und bei der interaktiven Planung zuerst durchgeführt, da für sie viele komplexe Randbedingungen gelten. Weniger schwierig ist die als sekundäre Zuordnung bezeichnete Zuordnung der Unterrichtsstunden zu den Räumen. Hier existieren nur wenige Regeln, Räume stehen in einer Schule meist in genügender Anzahl zur Verfügung (nach [Busch 97]).

4.3.2 Traditionelle Darstellung

Nach [Busch 97] wird zur Stundenplanerstellung üblicherweise eine Wandtafel eingesetzt, wie sie in Abbildung 32 dargestellt ist. Diese ist wie eine Tabelle aufgebaut, wobei die Zeilen die Wochenstunden darstellen und die Spalten in drei Gruppen unterteilt sind: die Klassen, die Lehrer und die Räume. Dabei bedeutet nach [Busch 97] ein Eintrag in einer Zeile die Belegung der Klasse, des Lehrers oder des Raums zur entsprechenden Stunde. Die Information über den beteiligten Lehrer wird über die Farbe des Eintrags, die über das beteiligte Fach über Symbole kodiert, wobei ersteres nur an kleinen Schulen möglich ist, da für viele Lehrer die Zahl der eindeutig unterscheidbaren Farben nicht ausreicht. Dann müssen auch für die Kodierung des Lehrers Symbole eingesetzt werden. Dabei müssen sich die Einträge nicht an den Zeilen- oder Spaltengrenzen orientieren, sondern können diese überschreiten wie etwa im Fall von Doppelstunden. Auch können in einer Zelle zwei Einträge eingezeichnet werden; in Abbildung 32 wird zum Beispiel bei derselben Klasse zwischen den Unterrichtsveranstaltungen Sport männlich und Sport weiblich differenziert, die beide gleichzeitig stattfinden.

	5a	5b	5c	...	KARL	
Mo 1	Ge	Ma	Mu			
Mo 2	Deu	Deu	Deu			
Mo 3	Ma	T Z	T A	Ge		
Mo 4	Bio		Bio			
Mo 5	Eng	Eng	Eng			
Mo 6	Mu	Mu	Ek			
Mo 7						
Mo 8	S P K N	S P M A	S P M A	S P K N		
Mo 9						
Mo10						
Mo 11						
Di 1						

Abbildung 32: Traditionelle Darstellung der Schulstundenplanung durch eine Plantafel
(nach [Busch 97])

4.3.3 Einordnung in die Grundtypen

4.3.3.1 Zuordnungsdarstellung

Die Schulstundenplanung ist ein Beispiel für eine Gruppendarstellung kombiniert mit einer Mehrfachzuordnung, wobei die primäre Zuordnung die der Nachfrager, der Unterrichtsstunden, zu den Wochenstunden und die sekundäre die der Nachfrager zu den Räumen ist. Werden Klassen für einzelne Unterrichtsveranstaltungen getrennt, so ist die Darstellung nicht injektiv, wie in Abbildung 32 im Falle des Sportunterrichts zu sehen. Somit müssen in einer Zelle der Tabelle eventuell mehrere Zuteilungen kodiert werden.

Alle Nachfrageobjekte, die in ihrer Klasse, bzw. ihrem Lehrer oder Raum übereinstimmen, werden zu einer Gruppe zusammengefaßt. Welches der Nachfrageobjekte der Gruppe dem primären Angebotsobjekt zugeteilt ist, wird über das Markierungssymbol ausgedrückt. Die sekundäre Zuordnung wird im allgemeinen ebenfalls über das Markierungssymbol visualisiert, auch wenn sie in Abbildung 32 nicht eingezeichnet ist.

4.3.3.2 Randbedingungen

In Tabelle 10 sollen nun die Randbedingung der Schulstundenplanung in die in Kapitel 2 vorgestellten Grundtypen eingeordnet werden.

Randbedingung	Typ	Zuteilungs- kombi- nationen	Gemein- sames Objekt
Triviale Randbedingungen			
Ein Lehrer oder eine Klasse nicht zur selben Zeit in zwei verschiedenen Räumen	Vermeidung	Ja	Wochen- stunde
Zeitpräferenzen			
Präferenzen für bestimmte Wochenstunden	Durchsetzung/ Vermeidung	Nein	-
Präferenzen für unbestimmte Wochenstunden	Arithmetischer Ziel- oder Grenzwert	Ja	Lehrer, bzw. Klasse, Fach oder Raum
Randbedingungen bezüglich Unterrichtsstunde – Woche			
Maximale Anzahl von Unterrichtsstunden ohne Pause für Lehrer oder Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Lehrer, bzw. Klassen
Maximale Anzahl von wöchentlichen Freistunden für Lehrer oder Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Lehrer, bzw. Klasse
Minimale und maximale Länge der Mittagspause für Lehrer oder Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Lehrer, bzw. Klasse
Minimale oder maximale Stundenzahl pro Tag für Lehrer oder Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Lehrer, bzw. Klasse
Maximale Zahl von Hauptfächern pro Tag für Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Klasse
Maximale Zahl der Hauptfächer in Folge für Klassen	Arithmetischer Ziel oder- Grenzwert	Ja	Klasse
Jedes Fach nur einmal pro Tag in einer Klasse (außer Doppelstunden)	Vermeidung	Ja	Fach und Klasse
Bestimmte Unterrichtsveranstaltungen in Doppelstunden	Durchsetzung	Ja	Unterrichts- veranstaltung
Bestimmte Unterrichtsveranstaltungen in bestimmter Reihenfolge	Reihenfolge/ Muster	Ja	Klasse

Koppelung bestimmter Unterrichtsveranstaltungen	Durchsetzung	Ja	Wochenstunde
Explizites Verbot der Koppelung bestimmter Unterrichtsveranstaltungen	Vermeidung	Ja	Wochenstunde
Randbedingungen bezüglich Unterrichtsstunde – Raum			
Eignung des Raums für das Fach	Vermeidung	Nein	-
Nur eine Unterrichtsveranstaltung gleichzeitig in einem Raum	Vermeidung	Ja	Raum
Falls explizit gefordert, bestimmte Unterrichtsveranstaltungen in einem Raum	Durchsetzung	Ja	Raum
Ausreichende Kapazität für Klasse(n)	Vermeidung	Nein	-
Stammräume von Klassen und Lehrer	Durchsetzung	Nein	-

Tabelle 10: Einordnung der Randbedingungen der Schulstundenplanung in die Grundtypen

4.3.4 Beispiel für die resultierende Darstellung

Anders als bei der Dienstplanung im Krankenhaus ist bei der Schulstundenplanung die resultierende Darstellung nicht zu 100% mit der traditionellen Darstellung identisch; für eine Integration der zellübergreifenden Visualisierung müssten die Grundtypen erweitert werden. Allerdings ist die in Tabelle 11 zu sehende resultierende Darstellung durchaus akzeptabel und wurde bereits erfolgreich in der Praxis angewendet [Busch 97]. Zur Kodierung der einzelnen Zuteilungen wurden in Tabelle 11 ausschließlich Symbole eingesetzt, allerdings wäre im Rahmen der Grundtypen auch eine Kodierung partieller Informationen mittels Farbe wie in Abbildung 32 möglich.

	5a HAL	5b WAG	5c KARL	KARL 5c	R04 5a
Mo0					
Mo1	GE - FU R04	M R05	MU R06	MU	5a - FU
Mo2	*DEU R04	*DEU R05	*DEU R06	*DEU	5a - HAL
Mo3	M - FKA R04	*TA - ROS HA1	GE - FU R06		5a - FU
Mo4	BIO R04	*TA - ROS HA1	BIO R06	BIO	5a - FKA
Mo5	*E R04	*E R05	*E R04		5a - HAL
Mo6	MU M1	MU R05	EK - WEIG		
Mo7	*SPK5a SH1		*SPK5ac SH1	SPK5ac	
Mo8	*SPK5a SH1		*SPK5ac SH1		
Mo9					
Di1				...	...

Tabelle 11: Beispiel für die resultierende Visualisierung der Schulstundenplanung

4.3.5 Fazit

Anders als im Falle der Dienstplanung für das Pflegepersonal eines Krankenhauses müssen bei der Darstellung der Schulstundenplanung gewisse Abstriche gemacht werden, die Wandtafel läßt sich nicht vollkommen in das System der Grundtypen einordnen. Allerdings ist der Qualifikationsverlust nur gering, die resultierende Darstellung insgesamt brauchbar [Busch 97]. Die resultierende Visualisierung von Restriktionsverletzungen wurde nicht ausführlich untersucht, da sie, wie bereits bei der Dienstplanung erläutert, einer ausführlichen praktischen Evaluation bedarf, die im Rahmen dieser Arbeit nicht möglich ist. Allerdings ließen sich alle Randbedingungen problemlos entsprechenden Regeltypen zuteilen, die resultierende Visualisierung sollte also angemessen sein.

4.4 Dienstplanung für das ärztliche Personal der Klinik für Anästhesiologie der Universität Würzburg

4.4.1 Beschreibung der Dienstplanung

Die Dienstplanung für das ärztliche Personal der Klinik für Anästhesiologie der Universität Würzburg setzt sich aus zwei Bestandteilen zusammen:

- ?? Der als Dienstplan bezeichnete Bestandteil garantiert, daß Nacht- und Bereitschaftsdienst besetzt sind.
- ?? Der als Verteilungsplan bezeichnete Bestandteil legt für das ärztliche Personal der Klinik für Anästhesiologie für einen Zeitraum von vier Wochen fest, in welcher Klinik¹⁵ es eingesetzt wird. Die Zuweisung eines Arztes an eine Klinik erfolgt dabei in Wochenabschnitten. Primär muß der Verteilungsplan die klinischen Belange erfüllen, d.h. sicherstellen, daß jede Klinik im normalen Tagdienst über die erforderliche Anzahl ausreichend qualifizierten Personals verfügt. Zusätzlich muß die Ausbildung eines Teiles des Personals berücksichtigt werden, im Rahmen derer einzelne Ärzte verschiedene Stationen durchlaufen müssen.

Beide Pläne müssen die Urlaubs- und Dienstpräferenzen der einzelnen Ärzte berücksichtigen, die im Rahmen dieser Beschreibung als gegeben aufgefaßt werden und somit nicht weiter beschrieben werden sollen.

¹⁵ Die Klinik für Anaesthesiologie ist im Grunde genommen keine eigentliche Klinik, Anästhesiologen sind an fast jeder Operation beteiligt und werden somit allen Kliniken der Uniklinik zur Durchführung von Operationen zugeteilt.

4.4.1.1 Der Dienstplan

Für die Erstellung des Dienstplans muß die Qualifikation der einzelnen Ärzte berücksichtigt werden. Für den Nacht- und Bereitschaftsdienst wird zwischen 7 Diensten unterschieden, für die jeweils eine bestimmte Qualifikation notwendig ist und die in Tabelle 12 dargestellt sind. Jeder dieser Dienste muß jede Nacht mit einem entsprechend ausgebildeten Arzt besetzt werden, wobei die einzelnen Dienste möglichst gleichmäßig auf das vorhandene Personal verteilt werden sollten. Anders als für den Tagdienst erfolgt hier eine Zuweisung eines Arztes an einen Dienst immer nur für eine Nacht; dabei darf kein Arzt zwei Nächte hintereinander arbeiten.

Bezeichnung	Qualifikation	Einsatzort
1. Dienst	Facharzt, Aufsicht im OP	Chirurgie
2. Dienst	Fortgeschritten in der Ausbildung zum Facharzt, selbständige Narkose	Kopfkl. n. k
3. Dienst	In der Ausbildung zum Facharzt, noch keine selbständige Narkose	Chirurgie
4. Dienst	Fortgeschritten in der Ausbildung zum Facharzt, selbständige Narkose	Frauenklinik
5. Dienst	Facharzt	THG Klinik
6. Dienst	Beginn der Ausbildung zum Facharzt	Chirurgie
Orthopädischer Dienst	Fortgeschritten in der Ausbildung zum Facharzt, selbständige Narkose	Orthopädische Klinik

Tabelle 12: Beschreibung der ärztlichen Dienste

Generell kann ein Arzt in der Ausbildung zum Facharzt nur dann für den Nacht- und Bereitschaftsdienst in einer Klinik eingeteilt werden, wenn er diese Station im Rahmen seiner Ausbildung schon durchlaufen hat. Ein Arzt, der die Station Frauenklinik noch nicht absolviert hat, kann also nicht im 4. Dienst eingesetzt werden. Die Qualifikation für einen bestimmten Dienst setzt sich aus formalen Kriterien wie der Anzahl durchgeführter Narkosen, dem Stand der Ausbildung und aus einer subjektiven Einschätzung durch den Vorgesetzten zusammen. Im Laufe seiner Ausbildung ist ein Arzt erst nur im 6. Dienst einsatzfähig, die weitere Reihenfolge lautet 3. Dienst, 2. Dienst und orthopädischer Dienst, 4. Dienst, 5. Dienst, 1. Dienst (oder 1. Dienst, dann 5. Dienst). Wenn im Bezug auf Ärzte von einem Dienst gesprochen wird, so steht diese Bezeichnung informell für einen Arzt, der zu eben diesem Dienst qualifiziert ist, ein 2. Dienst ist dann ein mindestens zum 2. Dienst befähigter Arzt.

4.4.1.2 Der Verteilungsplan

Der Verteilungsplan umfaßt zwei Aspekte, den normalen Dienst für den täglichen Operationsbetrieb an den einzelnen Kliniken und die Dienstausslösung.

Normaler Dienst

Folgende Kliniken müssen für den normalen Operationsbetrieb mit anästhesiologischen Personal besetzt werden:

Bezeichnung	Personalbedarf
Intensiv	7
Schmerzambulanz	2
Neurochirurgie (NCH)	6
Hals, Nasen, Ohren (HNO)	5
Augen	4
Frauen	4
Orthopädie	5
Thorax, Herz, Gefäße (THG)	5
Zahn, Mund, Kiefer (ZMK)	3
Zentralbereich	10
Urologie	4

Tabelle 13: Personalbedarf der einzelnen Kliniken

Bis auf die Schmerzambulanz muß jeder Klinik eine Aufsicht, also ein zum 1. Dienst befähigter Arzt zugeteilt werden. Wenn dann noch weiteres zum 1. Dienst geeignetes Personal vorhanden ist, wird es in der Regel im Zentralbereich eingesetzt. Neben diesen formalen Kriterien werden bei der Planung auch informelle Faktoren berücksichtigt, die allerdings nicht spezifiziert werden können oder dürfen (z.B. Wissen, welche Ärzte ein gutes Team bilden). Die Zuweisung eines Arztes an eine Klinik erfolgt immer für ganze Wochen, eine Zuteilung für einzelne Tage einer Woche ist nicht möglich. Grundlegendes Prinzip bei der Planung des Tagdienstes ist die Vermeidung häufiger Wechsel, von Woche zu Woche sollte sich bei der Diensterteilung möglichst wenig ändern.

Schließlich sollten noch die Wünsche des Personals und Ausbildungserfordernisse berücksichtigt werden; Ärzte müssen im Rahmen ihrer Ausbildung zum Facharzt bestimmte Stationen durchlaufen. Während einer solchen Station, von denen jede ca. 1 Jahr dauert, müssen sie ihren normalen Dienst immer in derselben Klinik absolvieren, können also nicht frei eingesetzt werden.

Dienstausslösung

Aufgabe der Dienstausslösung ist es, den Nacht- und Bereitschaftsdienst der vergangenen Nacht in ihrem Tagdienst zu ersetzen. Deshalb müßte sie aus 6 Personen zu bestehen, um den 1. bis 6. Dienst ablösen zu können (der orthopädische Dienst wird im allgemeinen nicht abgelöst), wie es an sich vom Gesetzgeber vorgeschrieben ist. Aufgrund von Personalmangel ist das jedoch derzeit nicht konsequent möglich, und die Dienstausslösung umfaßt nur 3 oder 4 Ärzte, dann meist vom 1. bis zum 3. bzw. 4. Dienst, auf jeden Fall aber ein Mitglied des 1. Dienstes. Mitglieder der Dienstausslösung können im allgemeinen nicht im Nacht- und Bereitschaftsdienst eingesetzt werden. Nur die Intensivstation regelt ihren Nacht- und Bereitschaftsdienst intern.

4.4.1.3 Dienst- und Verteilungsplanung als Zuordnungsproblem

Der Dienstplan ist eine einfache 1-n Zuordnung mit den einzelnen Diensten eines Tages als Nachfragern, also z.B. dem 1. Dienst des ersten Tags des Planungszeitraums. Die Ärzte sind die Angebotsobjekte.

Der Verteilungsplan entspricht der Dienstplanung für das Pflegepersonal eines Krankenhauses, nur daß die Nachfrager nicht Schichten eines Tages, sondern Schichten über den Zeitraum einer Woche sind und die Angebotsobjekte Ärzte, nicht Pfleger. Für den Verteilungsplan werden auch Dienstausslösung und Urlaub als Kliniken interpretiert, so daß Ärzte wie für alle anderen Kliniken auch Präferenzen zum Ausdruck bringen können. Der einzige Unterschied ist, daß die Besetzung des Urlaubs nicht garantiert sein muß.

4.4.2 Traditionelle Darstellung

4.4.2.1 Traditionelle Darstellung des Dienstplans

Der Darstellung des Dienstplans ist sehr einfach gehalten, in der Kopfzeile stehen die 7 Dienste, die für jede Nacht besetzt werden müssen, in der ersten Spalte steht die jeweilige Nacht des Planungsmonats, zur besseren Orientierung wird noch zusätzlich der Wochentag angegeben, an dem diese Nacht beginnt. In Tabelle 14 ist also für den 1. Dienst der 1. Nacht des Monats, einem Montag, der Arzt Anetseder eingeteilt, für den der 2. Nacht des Monats der Arzt Rübsam, usw.

		1. Dienst	2. Dienst	3. Dienst	4. Dienst	5. Dienst	6. Dienst	Ortho
1	Mo	Anetseder	Englert	Wurm	Brandt	Eberlein	Schmied	Geier
2	Di	Rübsam	Zimmer	Otremba	Fischer	Grün	Benkwitz	Müller
3	Mi	Butterhof	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...
31	Mi	Anetseder	...	...	...	...	...	...

Tabelle 14: Traditionelle Darstellung des Dienstplans für das ärztliche Personal

4.4.2.2 Traditionelle Darstellung des Verteilungsplans

Auch die Darstellung des Verteilungsplans ist vergleichsweise einfach. In der Kopfzeile wird die Woche spezifiziert, in der ersten Spalte die jeweilige Klinik, bzw. der Urlaub. Die Zahlen geben für jede Klinik an, wie viele Ärzte benötigt werden. Somit ist zum Beispiel in Tabelle 15 Kraus in der Woche vom 8-14.3.99 für die Intensivstation eingeteilt, Englert für die Schmerzklinik usw.

		8-14.3.99	15-21.3.99	22-28.3.99	29.3-4.4.99
Intensiv	1	Kraus	Kraus	...	...
	2	Mildenberger	Mildenberger		
	3	Nenninger	Herrler		
	4	Gießelmann	Gießelmann		
	5	Geier	Müller		
	6	Hofmann	Hofmann		
	7	Horbaschek	Horbaschek		
Schmerz	1	Englert	Englert	Englert	Englert
	2	Klüber	Klüber	Klüber	Klüber
...		...	...	...	...
Urlaub		Anetseder Schwemmer ...	...	...	...

Tabelle 15: Traditionelle Darstellung des Verteilungsplans für das ärztliche Personal

4.4.3 Einordnung in die Grundtypen

4.4.3.1 Zuordnungsdarstellung

Der Dienstplan ist eine komprimierte Listendarstellung; alle Dienste desselben Typs des Planungszeitraums werden zu einer Gruppe zusammengefaßt, also der z.B. der 1. Dienst des ersten Tags, der 1.Dienst des zweiten Tags, usw. Das Datum in der ersten Spalte zeigt dann an, welcher der Dienste einer Gruppe tatsächlich gemeint ist.

Auch der Verteilungsplan ist eine komprimierte Listendarstellung, bei der alle normalen Dienste einer Woche zu einer Gruppe zusammengefaßt werden. Der Klinikname in der ersten Spalte zeigt dann an, welcher dieser normalen Dienste tatsächlich gemeint sind.

4.4.3.2 Randbedingungen

Dienstplan

In Tabelle 16 sollen nun die Randbedingung der Dienstplanung in die in Kapitel 2 vorgestellten Grundtypen eingeordnet werden.

Randbedingung	Typ	Zuteilungs-kombinationen	Gemeinsames Objekt
Urlaubsplan beachten	Vermeidung	Nein	-
Keine Mitglieder der Dienstausslösung einsetzen	Vermeidung	Nein	-
Dienstpräferenzen beachten	Durchsetzung/ Vermeidung	Nein	-
Keine zwei Dienste hintereinander (sowieso nicht gleichzeitig)	Vermeidung	Ja	Arzt
Qualifikation beachten	Durchsetzung	Nein	-
Gleichmäßige Verteilung der Dienste	-	Ja	-

Tabelle 16: Klassifizierung der Randbedingungen des Dienstplans für das ärztliche Personal

Verteilungsplan

In Tabelle 17 sollen nun die Randbedingung des Verteilungsplans in die in Kapitel 2 vorgestellten Grundtypen eingeordnet werden.

Randbedingung	Typ	Zuteilungs-kombinationen	Gemeinsames Objekt
Ausbildung eines Arztes berücksichtigen	Durchsetzung	Nein	-
Personalbedarf der Kliniken muß erfüllt werden	Arithmetischer Zielwert	Ja	Klinik
Arzt kann in einer Woche nur in einer Klinik eingesetzt werden	Vermeidung	Ja	Arzt
Möglichst wenige Veränderungen in der Besetzung einer Klinik	Vermeidung	Ja	Klinik
Keine im Dienstplan für den Nachtdienst vorgesehene Personal in die Dienstausslösung einteilen.	Vermeidung	Nein	-

Tabelle 17: Klassifizierung der Randbedingungen des Verteilungsplans für das ärztliche Personal

4.4.4 Fazit

Die Dienstplanung für das ärztliche Personal ist ein relativ einfaches Zuordnungsproblem, bei dem jedoch die ständig wechselnden Qualifikationen des Personals – ein großer Teil der Ärzte befindet sich in der Ausbildung – eine konstante Datenpflege erfordern. Auch informelle Aspekte (Welche Ärzte bilden ein gutes Team?), die ein erfahrener Planer intuitiv berücksichtigt, spielen eine große Rolle. Somit ist fraglich, ob eine Oberfläche eine so große Hilfe bei der Dienstplanung sein kann wie sie es im Falle der Dienstplanung für das Pflegepersonal sicher ist. In Bezug auf die Darstellung und Visualisierung von Randbedingungen sollte die Generierung einer entsprechenden Oberfläche jedoch unproblematisch sein.

5 Entwurf einer Lösung

5.1 Verwandte Arbeiten

Das Design und die Implementierung graphischer Benutzeroberflächen ist ein komplexer Prozeß, der viele Ressourcen in Anspruch nimmt [Maybury & Wahlster 98]. Deshalb wurden und werden viele Anstrengungen unternommen, diesen Prozeß zu unterstützen, angefangen von der Bereitstellung von Standardkomponenten bis hin zum Versuch der automatischen Generierung von Oberflächen. In diesem Abschnitt sollen einige maßgebliche Ansätze kurz vorgestellt werden.

5.1.1 Toolkits

Toolkits sind Bibliotheken, die eine Reihe von Standardkomponenten von Benutzeroberflächen wie etwa Fenster und Menüs zur Verfügung stellen, die dann vom Benutzer an seine Ansprüche adaptiert werden können. Für diese Toolkits gibt es häufig eine Art Aufsatz, die als Interface-Builder (z.B. Microsoft Visual C++) bezeichnet wird. Ein Interface-Builder erlaubt es, die Grundbausteine – meist mit der Maus – wie in einem Zeichenprogramm anzupassen und zu kombinieren, so daß der Designer zu jedem Zeitpunkt erkennen kann, wie die Oberfläche aussieht. Diese Komponenten verfügen über eine bestimmte Grundfunktionalität, so können Fenster in ihrer Größe verändert werden, eine Combo-Box reagiert auf einen Mausklick usw. Somit ist es möglich, ohne Programmierkenntnisse in kurzer Zeit das Gerüst eines Benutzerinterface zu erstellen, allerdings ohne weitergehende Funktionalität, die ein Entwickler selbst per Hand implementieren muß.

Toolkits und insbesondere Interface-Builder reduzieren den Entwicklungsaufwand für Oberflächen drastisch – mittlerweile ist die Verwendung von Toolkits bereits Standard geworden, eine Implementation eigener Grundkomponenten, wie sie noch vor wenigen Jahren durchaus üblich war (etwa unter MS-DOS, siehe z.B. [gp-UNTIS 96]), ist heute kaum noch vorstellbar. Ein weiterer Grund für ihre Popularität ist, daß Toolkits Design-Konsistenz innerhalb einer Oberfläche und zwischen verschiedenen Oberflächen, die auf demselben Toolkit beruhen, fördern, da immer dieselben Grundbausteine benutzt werden.

In Bezug auf die Art und Weise, wie diese Komponenten benutzt werden sollten, gibt es für viele Toolkits schriftliche Stilleitfäden (z.B. [APPLE 86]). Allerdings unterliegt die Umsetzung dieser Standards den menschlichen Designern, die diese in manchen Fällen durchaus unterschiedlich interpretieren, so daß Toolkits Interfacekonsistenz nicht garantieren können. Darüber hinaus erfordern Toolkits für komplexe Funktionalität Programmierkenntnisse, so daß Oberflächen mit ihrer Hilfe nicht direkt von Design-, Graphik- und Human-Factors Experten ohne solche Kenntnisse erstellt werden können [Wiecha et al. 91]. Auch können Oberflächen mit Toolkits beliebig „unsauber“ programmiert werden, das heißt die Implementation der Oberfläche und der zugrundeliegenden Applikation können stark miteinander verwoben sein, so daß die Oberfläche nur mit einer entsprechenden Anpassung der Applikation verändert werden kann und vice versa; auch können so diese beiden Bestandteile nicht durch voneinander unabhängige Entwickler mit jeweils eigenen Werkzeugen entworfen werden, wie es eigentlich wünschenswert wäre.

5.1.2 User Interface Management

Der Versuch der konsequenten Trennung von Anwendung und Interface ist ein Merkmal von User Interface Management Systemen (UIMS). Dieser Begriff umfaßt ein weites Spektrum an unterschiedlichen Systemen und Ansätzen [Myers 89], ein Beispiel für ein UIMS ist der Common Lisp Interface Manager (CLIM) [Möller 98]. Im engeren Sinn sind UIMS nach [Möller 98] „sophisticated software libraries with high-level abstractions for implementing user interfaces“ und somit Toolkits. Allerdings geht die Funktionalität von UIMS über die Bereitstellung einer Bibliothek hinaus, sie versuchen, weitergehende Funktionalität zu bieten, etwa Designhilfen und eine automatische Evaluation einer Oberfläche, die den Benutzer auf eventuelle Schwachpunkte hinweisen kann.

Wie erwähnt, ist der Versuch, Anwendung und Interface zu trennen, ein wichtiges Kennzeichen von UIMS. Konkret werden dazu verschiedene Layer eingeführt, von denen eines das sogenannte Interface Layer ist, das die eigentliche Darstellung und den Kontrollfluß innerhalb der Oberfläche und in Richtung des Application Layer regelt. Dabei können im letzteren eine Reihe von Funktionen definiert werden, die dann vom Interface Layer aufgerufen werden können. Allerdings muß ein Designer immer noch explizit die Namen von Funktionen spezifizieren, Menüs strukturieren und Fenster entwerfen, ihm werden nur die Grundbausteine zur Verfügung gestellt und zusätzlich besagte Hilfen angeboten, bzw. Konzepte forciert.

Meist bieten UIMS auch neben einem Interface-Builder eine Möglichkeit, ein Interface deklarativ in einer Interface Definition Language (IDL) zu spezifizieren. Ein Beispiel für eine Deklaration für einen Bildschirm eines Flugreservationssystems, auf dem man zwischen verschiedenen Optionen wählen kann [Wiecha et al. 91]:

```
:frame id=main
  :choice purpose=overview, message="Select an item to"
    :ci message="View flight for today", activate=check-today
    :ci message="View future flights", activate=check-by-date
    :ci message="Make a reservation", activate=reserve
  :echoice
:eframe
```

Allerdings geht dieses Beispiel bereits über ein normales UIMS hinaus, da nur der Zweck *overview* und nicht eine konkrete Komponente wie z.B. Pull-down-menu spezifiziert wird; ein solcher Ansatz wird als modell-basiert bezeichnet und soll im folgenden erläutert werden.

5.1.3 Modell-basierte Ansätze

Modell-basierte¹⁶ Ansätze sind spezielle UIMS (z.B. [Wiecha et al. 91]), die versuchen, auf einer höheren Abstraktionsebene anzusetzen; dazu werden umfangreiche Modelle der Mensch-Computer Interaktion entworfen. Der Designer spezifiziert dann eine Aufgabenstellung in Anlehnung an ein solches Modell und das System erzeugt ein

¹⁶ Die Einordnung in die Kategorien, bzw. die Trennung zwischen den unterschiedlichen Kategorien und auch deren Bezeichnung wird in der Literatur durchaus unterschiedlich gehandhabt; dieser Überblick orientiert sich in Bezug auf diese Aspekte an [Wiecha et al. 91].

entsprechendes Interface, ausgehend von dem abstrakten Modell und der konkreten Beschreibung. So könnte ein Modell zum Beispiel verschiedene Grundintentionen der Informationskodierung (Wichtige Meldung, Überblick, Zusatzinformation) definieren, so daß ein Designer eine Aufgabe nur in eine dieser Kategorien einordnen muß. Im Modell müssen dann entsprechende Richtlinien existieren, die festlegen, wie solche Informationen entsprechend der Intention am besten visualisiert werden.

Somit ist Konsistenz innerhalb der Oberfläche garantiert, da Entscheidungen gemäß klarer Regeln in Anlehnung an das zugrundeliegende Modell getroffen werden. So kann auch mangelndes Designwissen eines menschlichen Designers kompensiert werden. Das Problem ist nur, ein in sich schlüssiges Modell zu entwickeln, daß auch komplexen Ansprüchen gerecht wird. Im Fall eines einfachen Dialogs ist das kein Problem, nur lassen sich gegenwärtig komplexe Schnittstellen im Rahmen eines solchen Modells noch nicht umsetzen. Auch ist es problematisch, das Wissen zur Erstellung einer qualitativ hochwertigen Oberfläche zu abstrahieren und in einer geeigneten Form zu repräsentieren.

Allerdings versprechen sich viele Experten, z.B. [Maybury & Wahlster 98] von modellbasierten Ansätzen große Fortschritte im Bereich der Interface-Erzeugung.

5.2 Generierung von Oberflächen zur interaktiven Lösung von Zuordnungsproblemen

5.2.1 Einführung

Aufgabenstellung dieser Arbeit ist die automatische Generierung von interaktiven Oberflächen zur Lösung von Zuordnungsproblemen. Vergleicht man diese Zielsetzung mit den existierenden Arbeiten auf dem Gebiet der intelligenten Benutzerschnittstellen, so fällt auf, daß sie sehr hoch gesteckt ist. Nach [Maybury & Wahlster 98] ist kein System in der Lage, für reale Applikationen automatisch Oberflächen zu generieren, die mit von menschlichen Experten entworfenen Benutzerschnittstellen auch nur annähernd konkurrieren könnten, trotz der umfangreichen Anstrengungen, die auf diesem Gebiet unternommen wurden und werden. Bei allen im letzten Abschnitt vorgestellten Ansätzen muß ein Designer umfangreiche Spezifikationen vornehmen, wenn auch auf unterschiedlichen Abstraktionsebenen.

Die vorliegende Aufgabenstellung ist dennoch realistisch, da sie stark auf einen eng begrenzten Problemkreis fokussiert ist. Somit läßt sie sich auch nur schwer in die existierenden Ansätze einordnen, da sich diese primär allgemein mit Oberflächen, nicht mit Oberflächen nur für ein bestimmtes Problem beschäftigen. Ein solcher allgemeiner Ansatz ist jedoch für eine so komplizierte Oberfläche, wie sie zur interaktiven Lösung von Zuordnungsproblemen nötig ist, zumindest unter den Rahmenbedingungen einer Diplomarbeit wenig erfolgversprechend.

In diesem Abschnitt soll nun ein Konzept zur Lösung der Aufgabenstellung vorgestellt werden; dazu sollen erst die einzelnen Komponenten des Konzepts eingeführt werden und in einer Zusammenfassung gezeigt werden, wie sie sich zu einer Lösung zusammenfügen. Dabei gilt, daß diese Lösung ein eigenständiges System bildet, an das zur Erzeugung einer Oberfläche ein weiteres System, im allgemeinen ein Zuordnungswerkzeug (im konkreten Fall die Zuordnungsshell COKE), angekoppelt werden muß.

Die Wissensbasis

Die Wissensbasis enthält allgemeines Wissen über Zuordnungsprobleme und ihre Darstellung. Dieses Wissen kann nicht alle Aspekte aller Zuordnungsprobleme umfassen, deshalb muß, bzw. kann ein Experte oder auch ein Benutzer der generierten Oberfläche dieses Wissen problem- und auch benutzerspezifisch konfigurieren.

Bibliothek der Grundbausteine

Die Grundbausteine, die in Kapitel 3 vorgestellt wurden, wurden im Rahmen einer Bibliothek implementiert. Die Komponenten dieser Bibliothek wurden dabei als eine Hierarchie von abstrakten Visualisierungsklassen strukturiert. Dabei wurde besonders auf Möglichkeiten zur Erweiterung, bzw. Adaption geachtet, da die Grundbausteine nicht sämtliche Ansprüche aller denkbaren Zuordnungsprobleme abdecken können.

Trennung von Anwendung und Oberfläche – Das Interface

Das Präsentationstool, bzw. die Bausteine der Bibliothek, auf der einen Seite und das Zuordnungswerkzeug, für das die Oberfläche generiert werden soll, auf der anderen Seite

kommunizieren über ein exakt definiertes Interface. Dieses wurde so einfach wie möglich gehalten, so daß ein großes Spektrum an Zuordnungswerkzeugen an die Oberfläche angeschlossen werden kann, also Zuordnungswerkzeuge leicht den Anforderungen dieser Schnittstelle angepaßt werden können, bzw. diesen ohnehin weitgehend entsprechen.

Die Koordinationsstelle – Das Präsentationstool

Die Koordinationsstelle des Konzepts ist das Präsentationstool, das die Generierung einer Oberfläche, bzw. die Generierung einzelner Bestandteile der Oberfläche kontrolliert und die Daten von den einzelnen Komponenten anfordert, bzw. an diese sendet.

Zusammenfassung

Abbildung 31 zeigt, wie sich die einzelnen Komponenten in das Gesamtkonzept einfügen.

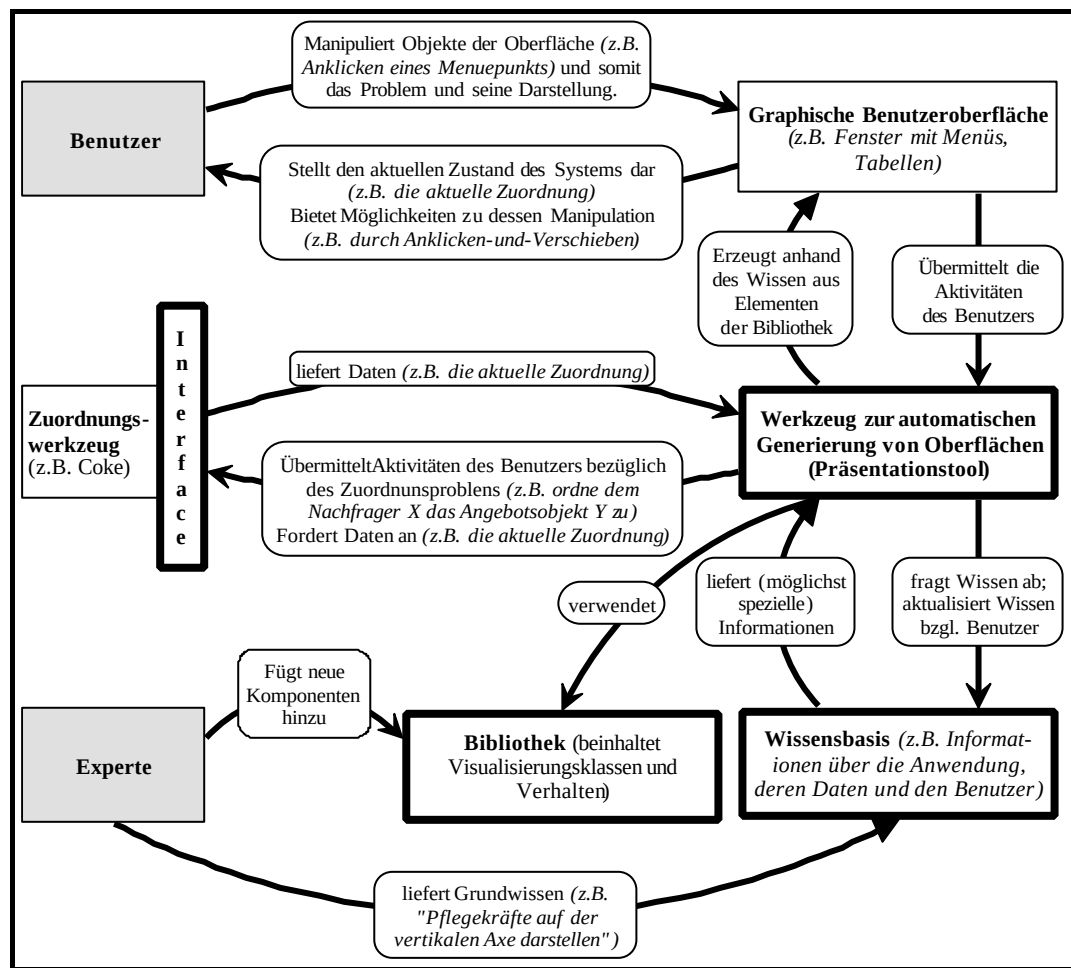


Abbildung 33: Gesamtkonzept zur Generierung von Oberflächen

Soll eine Oberfläche für ein Zuordnungsproblem generiert werden, so muß das zugrundeliegende Zuordnungswerkzeug ein Interface mit exakt definierter Funktionalität bereitstellen. Ein Experte muß einige wenige Informationen und kann beliebig viel über das darzustellende Zuordnungsproblem in die Wissensbasis eingeben; falls die vorhandenen Bausteine der Bibliothek die erwünschte Funktionalität nicht oder nur teilweise abdecken, muß er diese eventuell entsprechend seinen Anforderungen erweitern, bzw. anpassen.

Anhand des problemspezifischen, des benutzerspezifischen und des prinzipiellen Wissens über Zuordnungsprobleme und ihre Darstellung, das in der Wissensbasis enthalten ist, generiert das Präsentationstool dann aus den Bausteinen der Bibliothek eine Oberfläche. Die Daten, die es visualisieren soll, fordert das Präsentationstool über das Interface vom Zuordnungswerkzeug an.

Der Benutzer kann diese Oberfläche benutzen und dabei sowohl das Zuordnungsproblem als auch Visualisierungsaspekte manipulieren. Aktivitäten bezüglich des Zuordnungsproblems leitet das Präsentationstool an das Interface und somit an das Zuordnungswerkzeug weiter. Änderungswünsche bezüglich der Visualisierung setzt das Präsentationstool um und aktualisiert dabei gleichzeitig das Benutzerprofil in der Wissensbasis. Das Benutzerprofil wird dazu benutzt, bei zukünftigen Entscheidungen die Wünsche des Benutzers zu erschließen und ihm manuelle Anpassungen zu ersparen.

5.2.2 Wissensbasis

5.2.2.1 Grundstruktur

Die Wissensbasis enthält die gesamten Informationen, die zur Darstellung und Bearbeitung eines Zuordnungsproblems relevant sind. Die Forderung nach einer möglichst weitgehenden Individualisierung im Verbund mit dem Konzept einer lernfähigen, intelligenten Benutzeroberfläche erfordert die Möglichkeit einer weitgehenden Differenzierung nach einzelnen Benutzern oder Benutzergruppen, darüber hinaus natürlich auch nach einzelnen Zuordnungsproblemen. Wo immer möglich, sollen entweder vom Benutzer direkt eingestellte oder vom System aus seinem Wissen über den Benutzer erschlossene Werte benutzt werden. Die Wissensbasis sollte daher zwischen folgenden Wissensquellen differenzieren:

- ?? Allgemeines Grundwissen
- ?? Problemspezifisches Wissen für alle Benutzer
- ?? Benutzerspezifisches Wissen für alle Probleme
- ?? Benutzer- und problemspezifisches Wissen

Eine konkrete Anfrage besteht immer aus einem Schlüssel (z.B. Hintergrundfarbe) und einer Visualisierungsklasse (z.B. Zuordnungstabelle) oder der Instanz einer Visualisierungsklasse (z.B. „Keine 5 Nachtdienste in Folge“ als Instanz der Visualisierungsklasse „Komplexe Vermeidung“, einer Unterklasse der Visualisierungsklasse „Randbedingung“), die zusammen als Kontext bezeichnet werden, z.B.:

```
(query-kb FOR $background-color OF $assignment-grid  
REGARDING PROBLEM *nurse-scheduling*  
REGARDING USER *default-user*)
```

Für die Beantwortung einer Anfrage muß eine Hierarchie vereinbart werden, die regelt, welche Quellen bevorzugt benutzt werden sollen; schließlich können unterschiedliche Quellen auf dieselbe Anfrage verschiedene Antworten geben. Zusätzlich besteht noch die Möglichkeit, zu versuchen, aus vorhandenen Antworten auf ähnliche Anfragen auf eine Antwort auf diese konkrete Frage zu schließen. Sollte, wenn der Benutzer die Hintergrundfarbe eines Fensters auf eine bestimmte Farbe umstellt, diese Hintergrundfarbe dann auch für andere Fenster gewählt werden oder, wenn der Benutzer die Schriftgröße einstellt, diese neue Schriftgröße überall verwendet werden?

Der gewählte Kompromiß sieht so aus, daß der Benutzer bei der Einstellung von Präferenzen differenzieren kann, ob er diese Präferenz global oder nur im konkreten Fall, also lokal, berücksichtigt haben will. Dabei werden extrem allgemeine Einstellungen vermieden – es erscheint wenig sinnvoll, eine generelle Einstellung für Farbe vornehmen zu können, besser eine differenziertere Einstellung für Hintergrundfarbe, Textfarbe u.ä.

Besteht bei einer Anfrage keine globale Präferenz eines Benutzers, so wird die in Abbildung 34 dargestellte Reihenfolge benutzt, die beste Quelle festzulegen: Es wird immer versucht, die konkrete Anfrage zu beantworten; dabei wird zuerst problem- und benutzerspezifisch gesucht, dann nur problemspezifisch, dann nur benutzerspezifisch und schließlich allgemein. Bei der Anfrage, welche Hintergrundfarbe die Zuteilungstabelle der Dienstplanung für das Pflegepersonal für den Benutzer Pflegekraft A haben sollte, wird, falls es keine direkte Antwort gibt, nach der Hintergrundfarbe der Zuteilungstabelle der Dienstplanung für das Pflegepersonal gesucht, dann für die Hintergrundfarbe von

Zuteilungstabellen für den Benutzer A, dann schließlich allgemein für die Hintergrundfarbe von Zuteilungstabellen. Diese Reihenfolge lieferte bei praktischen Versuchen die besten Resultate.

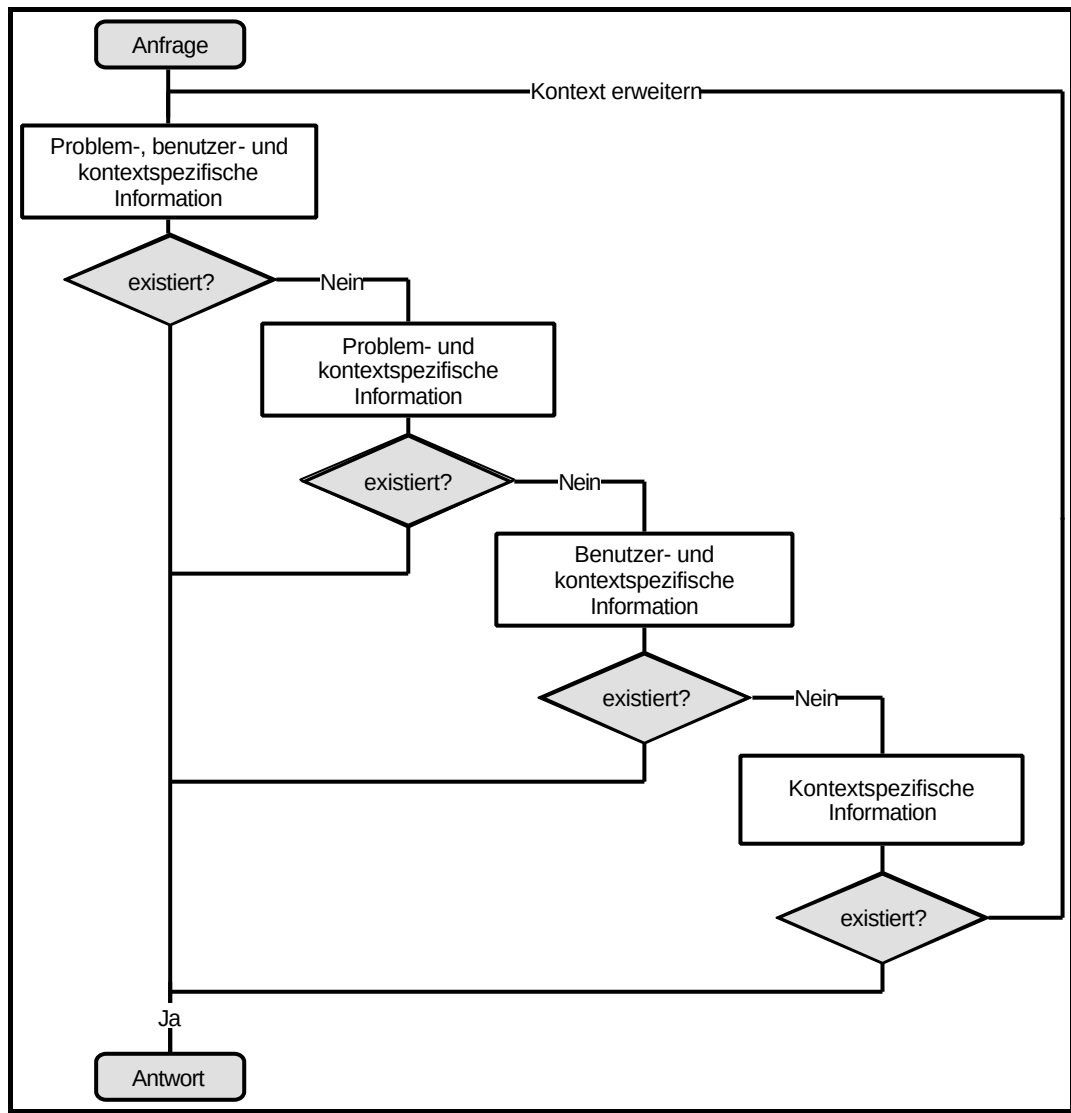


Abbildung 34: Beantwortung von Anfragen gemäß der Wissenshierarchie

Dabei besteht vor allem das allgemeine Wissen nicht aus statischen Werten, sondern wird dynamisch aus Regeln berechnet; der allgemeine Wert für die Hintergrundfarbe von Zuteilungstabellen ist keine bestimmte Farbe, sondern eine Regel, die diese Farbe aus anderen Faktoren abzuleiten versucht, etwa der gewählten Schriftfarbe. Der allgemeine Wert für die Größe eines Fensters wird zum Beispiel relativ zur aktuellen Bildschirmauflösung berechnet.

Wird auch im Grundwissen keine Antwort gefunden, so wird der Kontext erweitert. Eine Kontexterweiterung bedeutet, daß möglichst nah verwandte Visualisierungsklassen in die

Suche mit einbezogen werden, bzw. im Falle einer Instanz, daß die Visualisierungsklasse der Instanz miteinbezogen wird. Möglichst nah verwandt heißt hier konkret „möglichst früh im Vererbungsstammbaum der Visualisierungsklasse“. Deshalb ist es wichtig, nur dann allgemeine Antworten in die Wissensbasis einzutragen, wenn diese wirklich sinnvoll sind, da diese Antworten eine Suche in verwandten Visualisierungsklassen verhindern.

Im Beispielsfall würde die Prozedur mit der Frage nach der Hintergrundfarbe für Tabellen wiederholt werden, bis hin zur Frage nach der Hintergrundfarbe für die abstrakte Basisklasse, bei der eine Rekursion immer abbricht.

5.2.2.2 Erweiterung der Wissenshierarchie

Einen Sonderfall bilden Objekte, die selber Träger einer anderen Visualisierungsklasse sind und nur im Verbund mit einer solchen auftreten: Ein Beispiel dafür sind Zeilen oder Spalten einer Tabelle, die üblicherweise mit einem Datenobjekt oder einem Attribut assoziiert sind – eine Zeile oder Spalte an sich vermittelt keine Information, sondern erst im Verbund mit ihrem Inhalt, ihrer „Nutzlast“. Diese Situation muß auch beim Wissenserwerb berücksichtigt werden: Soll etwa die Spalte S, welche die Pflegekraft P darstellt, gelb dargestellt werden, so ist das eine Information, die logisch zur Pflegekraft A, technisch zur Spalte S gehört.

Deswegen muß beim Wissenserwerb für derartige Objekte zuerst geprüft werden, ob nicht wiederum für den Inhalt Werte spezifiziert wurden – wobei diese Anfrage wie beschrieben abläuft, schließlich ist der Inhalt wiederum eine Visualisierungsklasse, nur daß die gefundenen Werte für die Trägerklasse benutzt werden, nicht für den Inhalt. Konkret wird bei der Anfrage an die Wissensbasis bezüglich solcher spezieller Visualisierungsklassen- oder Objekte zuerst im Kontext des Inhalt gesucht, und nur, falls in diesen Kontext keine Antwort gefunden wird, zum Kontext der eigentlichen Klasse, bzw. des eigentlichen Objekts übergegangen.

5.2.2.3 Minimalkonfiguration eines Zuordnungsproblems

Ein gewisses Minimum an Information muß in der Wissensbasis vorhanden sein, um eine sinnvolle Darstellung und Lösung eines Zuordnungsproblems zu gewährleisten. Konkret muß für jede Oberfläche eines Zuordnungswerkzeugs die Struktur einer Zuteilung des modellierten Zuordnungsproblems bekannt sein.

Die Zuteilungsstruktur

Eine Zuteilungsstruktur ist ein närer Baum, dessen Knoten einen Funktionsnamen als Wert und einen Klassennamen als Schlüssel haben. Ein Schlüssel muß dabei eindeutig sein, keine zwei Knoten dürfen denselben Schlüssel haben. Wert und Schlüssel haben dabei folgende Bedeutung: Der Funktionsname eines Knotens spezifiziert die Funktion, die man aufrufen muß, um von einer Instanz der Klasse mit dem Klassennamen des Vaters des Knotens zu der Instanz des Klassennamens des Knotens zu kommen. Abbildung 35 zeigt die Zuteilungsstruktur der Dienstplanung für das Pflegepersonal eines Krankenhauses. Zur besseren Unterscheidung sind dabei die Schlüssel, also die Klassennamen, in die Knoten und die Werte, also die Funktionsnamen, auf die Äste gezeichnet. Um von einer Instanz der Klasse Zuteilung zu einer Instanz der Klasse

Arbeitskraft zu kommen, muß man also die Funktion „Nachfrageobjekt“ auf die Zuteilung anwenden.

Der konkrete Zweck der Zuteilungsstruktur wird in Zusammenhang mit der Zuteilungstabelle in 5.2.3 erläutert.

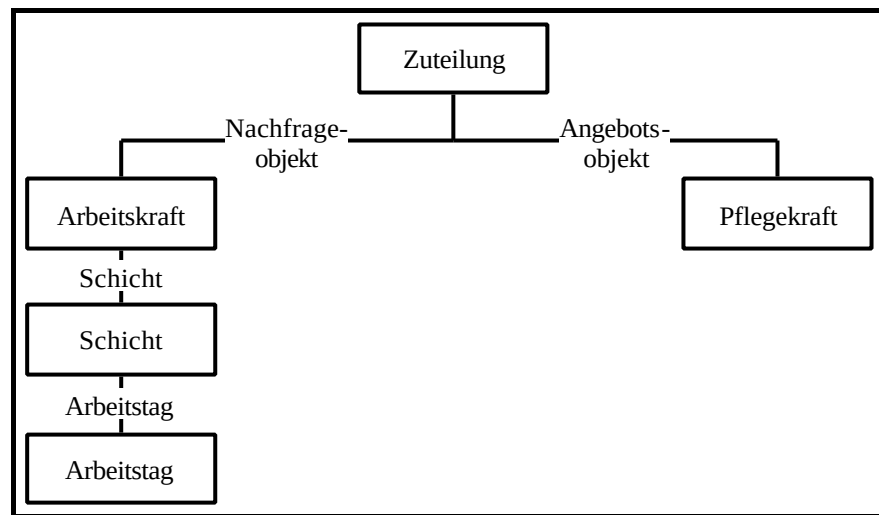


Abbildung 35: Zuteilungsstruktur der Dienstplanung für das Pflegepersonal eines Krankenhauses

5.2.2.4 Ausblick

Das vorgestellte Konzept beinhaltet noch einige Möglichkeiten, vorhandenes Wissen effektiver zu benutzen, bzw. intelligente Entscheidungen zu treffen, auf die im Rahmen dieser Arbeit nicht weiter eingegangen werden kann. So haben im derzeitigen Konzept Einstellungen eines normalen Benutzers nur für ihn selbst Gültigkeit. Eventuell ließe sich jedoch aus den Einstellungen vieler einzelner Benutzer Rückschlüsse ziehen, die allgemein gültig sind, entweder durch die Wissensbasis selbst (Stichwort „Machine learning“) oder durch einen menschlichen Experten, der diese Einstellungen analysiert.

5.2.3 Bibliothek

Die Bibliothek ist als eine Hierarchie von Visualisierungsklassen aufgebaut. Die Basis dieser Hierarchie stellt die abstrakte Klasse „Visualizable Object“ dar, auf der mittels abstrakter Methoden die Grundfunktionalität definiert wurde, die jede Visualisierungsklasse implementieren sollte. Damit ist gewährleistet, daß alle Elemente der Bibliothek dasselbe Grundverhalten haben und die Einarbeitungszeit in die Bibliothek reduziert wird. Weiter sind durch dieses objektorientierte Design und die entsprechende objektorientierte Implementierung eine leichte Erweiterbarkeit, bzw. Adaptierbarkeit einzelner Bausteine garantiert.

5.2.3.1 Die abstrakte Basisklasse „Visualizable Object“

Die abstrakte Basisklasse „Visualizable Object“ und damit jede Instanz einer Visualisierungsklasse hat folgende Eigenschaften:

- ?? User-Id: identifiziert den Benutzer eindeutig, z.B. Pflegekraft A
- ?? Problem-Id: identifiziert das Zuordnungsproblem eindeutig, z.B. Dienstplanung des Pflegepersonals der anästhesiologischen Klinik der Universität Würzburg
- ?? Instance-Id: identifiziert die Instanz der Visualisierungsklasse eindeutig
- ?? Visualization-Method: die Methode, die zur Darstellung des Objekts benutzt werden soll

Soll eine Instanz einer Visualisierungsklasse erzeugt werden, so muß zumindest eine User-Id angegeben werden; die Instance-Id wird automatisch angelegt, die weiteren Eigenschaften können, müssen aber nicht spezifiziert werden. Werden sie nicht angegeben, so initialisiert die Wissensbasis diese Eigenschaften (Siehe 5.2.3). Implementiert ist die Basisklasse wie folgt:

```
(defclass visualizable-object()
  ((problem-id :accessor problem-id
               :initarg :problem-id)
   (user-id :accessor user-id
            :initarg :user-id)
   (instance-id :accessor instance-id
                :initarg :instance-id)
   (visualization-method :accessor visualization-method
                         |:initarg :visualization-method)))
```

5.2.3.2 Die Visualisierungsmethode

Die zentrale Eigenschaft einer Visualisierungsklasse ist ihre Visualisierungsmethode. Diese wird mit einer Instanz einer Visualisierungsklasse, einer Darstellungskomponente des Bildschirms (z.B. ein Fenster), einem Bereich dieser Darstellungskomponente und eventuell noch weitere Informationen aufgerufen. Mit Aufruf dieser Funktion kann jede Visualisierungsklasse prinzipiell überall dargestellt werden, auch wenn das im Einzelfall nicht immer sinnvoll ist. So ist zum Beispiel eine Zuteilung als Visualisierungsklasse angelegt; soll eine konkrete Zuteilung dargestellt werden, so wird ihre Visualisierungsmethode aufgerufen. Welche konkrete Methode das ist, wird durch die Wissensbasis festgelegt und kann durchaus von der konkreten Instanz abhängen. Wie in

Kapitel 3.2 erläutert, kann es durchaus sinnvoll sein, verschiedene Restriktionen, d.h. verschiedene Instanzen einer einzelnen Visualisierungsklasse Constraint, unterschiedlich darzustellen. Die abstrakte Basismethode ist wie folgt definiert:

```
(defmethod visualize((visualizable-object visualizable-object)
                     (stream stream)
                     (box box)
                     &rest args)
  (declare (ignore visualizable-object stream box args)))
```

Welchen Inhalts die zusätzlichen Argumente sind, hängt von der speziellen Visualisierungsmethode ab; im allgemeinen müssen jedoch keine zusätzlichen Argumente übergeben werden.

5.2.3.3 Die Grundstruktur der Bibliothek

Wie bereits erläutert, ist die Bibliothek als eine Hierarchie von Visualisierungsklassen angelegt. Teilweise ist die Struktur dabei eine direkte Konsequenz aus den bisherigen Ergebnissen, so finden sich die verschiedenen Typen von Randbedingungen auch in der Bibliothek wieder. Im folgenden sollen nun die wichtigsten Visualisierungsklassen erläutert werden, von denen die wichtigsten in Abbildung 36 dargestellt sind.

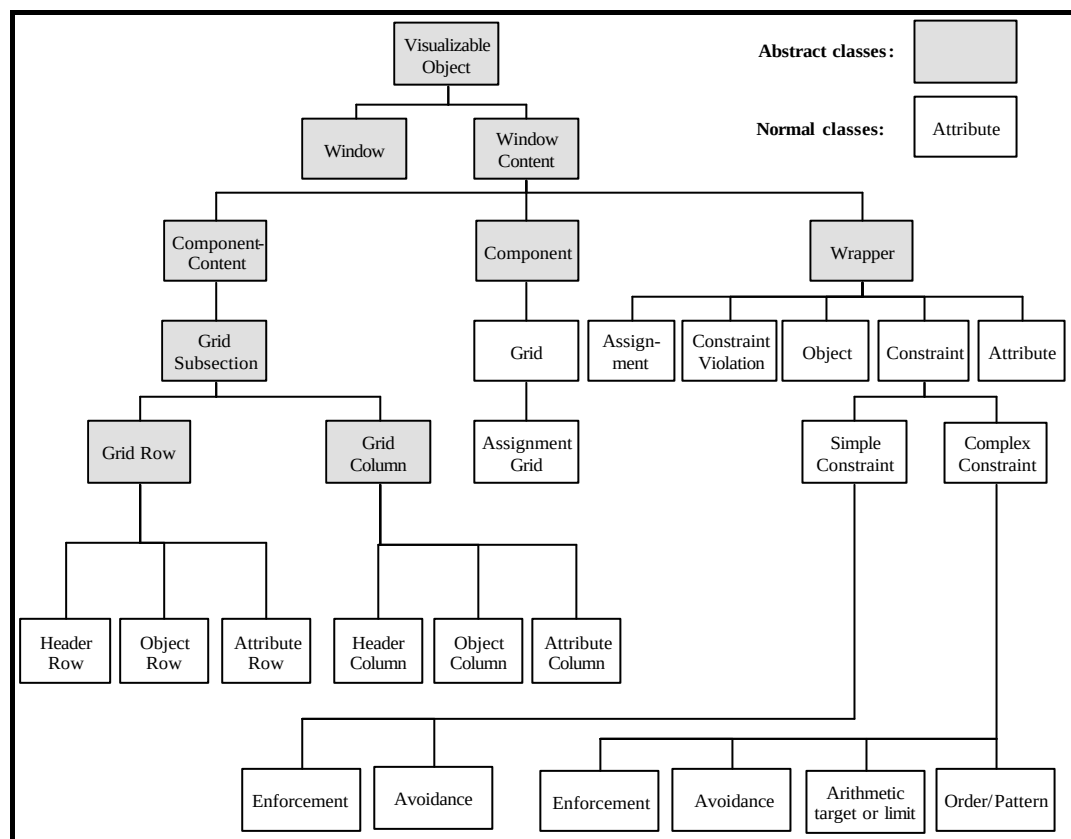


Abbildung 36: Ausschnitt aus Klassenhierarchie der Visualisierungsklassen

5.2.3.4 Hüllen-Klassen

Einen Sonderfall bilden die Klassen der Applikation – im Falle von Zuordnungsproblemen etwa die Klassen der Constraints oder der Nachfrageobjekte. Diese Objekte sollten wieder Instanzen von Visualisierungsklassen sein, allerdings sind sie vollkommen applikationsspezifisch. Dennoch müssen auch sie im Rahmen des Konzepts die erforderliche Funktionalität implementieren, also sich etwa visualisieren können, usw. Die Lösung für dieses Problem sind sogenannte Hüllenklassen (wrapper-classes), die intern ein Objekt einhüllen und die zusätzlichen Attribute und Methoden implementieren, also nach außen so wirken, als wären die eingehüllten Objekte tatsächlich Visualisierungsklassen, im allgemeinen Aufrufe von Methoden, die nicht Visualisierungsaspekte betreffen, an die eingehüllten Objekte weitergeben.

5.2.3.5 Die Visualisierungsklasse Zuordnungstabelle

Wichtigster Bestandteil der Bibliothek ist die Zuordnungstabelle; sie muß in der Lage sein, alle in Kapitel 3 beschriebene Grundtypen der Zuordnungsdarstellung zu visualisieren (die Kodierung und Visualisierung der Markierungsobjekte ist dagegen Aufgabe der Visualisierungsklasse Zuteilung). Die wichtigsten Bestandteile einer Zuordnungstabelle sind Sektionen. Sektionen bestehen wiederum aus Subsektionen, wobei Subsektion ein Sammelbegriff für Zeile und Spalte ist.

Sektion

Eine Sektion ist ein Abschnitt einer Tabelle, der Subsektionen enthält. Eine Tabelle kann beliebig viele Sektionen enthalten. Die in der Wissensbasis definierte Standardtabelle zur Darstellung eines Zuordnungsproblems enthält zum Beispiel 6 Sektionen, 3 horizontale und drei vertikale, wie in Abbildung 37 zu sehen ist.



Abbildung 37: Sektionen einer Tabelle

Subsektionen

Eine Subsektion ist eine Zeile oder Spalte und gehört zu einer Sektion. Eine Zuteilungstabelle hat 3 verschiedene Typen von Subsektionen:

?? Überschrift-Subsektionen werden mit einer Überschrift assoziiert.

?? Objekt-Subsektionen werden mit einem Objekt assoziiert.

?? Attribut-Subsektionen werden mit einem Attribut assoziiert.

Dabei haben Subsektionen an sich keine Bedeutung, nur der Schnittpunkt zweier Subsektionen, auch Zelle genannt, hat je nach Typ der sich schneidenden Subsektionen die in Tabelle 18 ausgeführte Bedeutung:

	Überschrift	Attribut	Objekt
Überschrift	-	Überschrift des Attributs	Überschrift des Objekts
Attribut	Überschrift des Attributs	-	Attribut des Objekts
Objekt	Überschrift des Objekts	Attribut des Objekts	Zuteilung

Tabelle 18: Bedeutung des Schnittpunkts zweier Subsektionen

5.2.3.6 Konfiguration einer Zuordnungsdarstellung

Will man eine Instanz der Klasse Zuordnungstabelle anlegen, so muß man spezifizieren, welche Objekte des Zuordnungsproblems auf der horizontalen, welche Objekte auf der vertikalen Achse dargestellt werden sollen. Bei der Dienstplanung für das Pflegepersonal eines Krankenhauses wären das alle Pflegekräfte auf der vertikalen, alle Arbeitstage auf der horizontalen Achse. Für jedes dieser Objekte kann man spezifizieren, in welcher Sektion der Zuordnungstabelle sie dargestellt sein sollen, die Standardeinstellung plaziert sie in die jeweilige Hauptsektion. Werden keine Objekte spezifiziert, so werden anhand der Zuteilungsstruktur die Klassen der Angebots- und Nachfrageobjekte bestimmt, über das Interface `all` ihre Instanzen angefordert und eine Funktionsdarstellung (siehe Kapitel 3) generiert. Ein Experte müßte für die Dienstplanung für das Pflegepersonal eines Krankenhauses also folgenden Eintrag in der Wissensbasis vornehmen, wobei die Funktionen „get-all-shifts“ und „get-all-workdays“ entsprechende Hilfsfunktionen sind, die über das Interface die entsprechenden Daten anfordern¹⁷.

```
(specify-in-kb FOR $get-objects-on-horizontal-axis-fn
  OF $assignment-grid
  REGARDING-PROBLEM *nurse-scheduling*
  REGARDING-USER *default-user*
  VALUE #'get-all-workdays)

(specify-in-kb FOR $get-objects-on-vertical-axis-fn
  OF $assignment-grid
  REGARDING-PROBLEM *nurse-scheduling*
  REGARDING-USER *default-user*
  VALUE #'get-all-nurses)
```

Die Spezifikation der Objekte geschieht im allgemeinen dynamisch durch Funktionen und nicht statisch durch die Angabe einzelner Objekte.

Zusätzlich kann man für jede der Objektklassen noch spezifizieren, welche ihrer Attribute dargestellt werden sollen, z.B.:

```
(specify-in-kb FOR $additional-attributes
  OF $nurse
  REGARDING-PROBLEM *nurse-scheduling*
  REGARDING-USER *default-user*
  VALUE '($name
    $working-hours-carry-over
    $hours-per-week
    $actual-versus-intended-working-hours
    $qualification-level))
```

Für jedes dieser Attribute kann man nun eine Visualisierungsmethode spezifizieren. Allerdings passen die in der Wissensbasis festgelegten Standardmethoden im allgemeinen, so werden Zahlen automatisch formatiert und dargestellt. Auch die Dateneingabe wird für die Standardtypen von der Wissensbasis abgedeckt. Spezifizieren muß man jedoch meistens die zugehörige Überschrift, da die entsprechende Methode der Wissensbasis im Beispielsfall die englischen Attributnamen verwenden würde. Auch die Sektion der Tabelle, in der das Attribut dargestellt werden soll, muß eventuell geändert werden, Attribute werden in der Standardeinstellung in der Anfangssektion dargestellt.

¹⁷ Die Pflegekräfte müßten eigentlich nicht spezifiziert werden, da sie als Angebotsobjekte auch in der Standardeinstellung gewählt werden würden.

5.2.3.7 Visualisierung von Zuteilungen

Die hier spezifizierten Daten reichen bereits aus, um eine Zuordnung darstellen zu können, vorausgesetzt, der Wissensbasis ist die Zuteilungsstruktur bekannt. Dazu wird eine Datenstruktur generiert, die für jeden Schnittpunkt zweier Objekt-Subsektionen, also allen Punkten der Zuordnungsdarstellung, an denen potentiell Zuteilungen dargestellt werden können, die zugehörigen Zuteilungen enthält. Sobald die Datenstruktur initialisiert ist, sollte man in konstanter Zeit durch die Angabe zweier Objekte die dazugehörigen Zuteilungen erhalten. In Pseudokode läßt sich eine solche Datenstruktur wie folgt generieren:

Fordere Liste aller Zuteilungen vom Interface an:

Für jede vorkommende KlasseH von Objekten auf der horizontalen Achse

Für jede vorkommende KlasseV von Objekten auf der vertikalen Achse

Für jede Zuteilung der Liste:

Extrahiere ObjektH der KlasseH von der Zuteilung

Extrahiere ObjektV der KlasseV von der Zuteilung

Merke Zuteilung unter Index (ObjektH, ObjektV)

Das Extrahieren eines Objekts einer beliebigen Klasse geschieht unter Einbeziehung der Zuteilungsstruktur. Im Beispielsfall wird nach Abbildung 35 zum Extrahieren eines Objekts der Klasse Pflegekraft die Methode „Nachfrageobjekt“ auf einer Zuteilung aufgerufen, für ein Objekt der Klasse Arbeitstag sukzessive die Methoden „Angebotsobjekt“, „Schicht“ und „Arbeitstag“.

Somit kann die Zuordnung effektiv visualisiert werden. Soll eine Zelle gezeichnet werden, so spezifiziert Tabelle 18 in Abhängigkeit von den sich schneidenden Subsektionen, was dargestellt werden soll. Soll eine Zelle dargestellt werden, die potentiell Zuteilungen enthält, so werden mittels obiger Datenstruktur in konstanter Zeit die mit dieser Zelle assoziierten Zuteilungen ermittelt. Falls mehrere Zuteilungen in einer Zelle dargestellt werden müssen, wird der Platz in der Zelle entsprechend aufgeteilt, in der Standardeinstellung in vertikale Abschnitte. Jeder der Zuteilungen wird dann der Auftrag gegeben, sich zu visualisieren. Im Beispielsfall muß dazu in der Wissensbasis für den Problemtyp „Dienstplanung für das Pflegepersonal“ und die Visualisierungs-klasse „Assignment“ eine Methode spezifiziert werden, die das Schichtsymbol des an der Zuteilung beteiligten Schichttyps in den übergebenen Bereich schreibt.

5.2.3.8 Visualisierung von Restriktionsverletzungen

Da die Visualisierung von Restriktionsverletzungen ein zentrales Thema dieser Arbeit ist, soll hier kurz darauf eingegangen werden. Auch Restriktionsverletzungen sind als Visualisierungsklassen angelegt, allerdings wie beschrieben als Hüllen-Klassen, da eine Restriktionsverletzung von einer Applikation erzeugt werden muß; die Überprüfung von Randbedingungen kann nicht Aufgabe der Oberfläche sein, da dies Oberfläche und Anwendung vermischen würde. Die eigentliche Visualisierung verläuft ähnlich wie die der Zuteilungen:

Wieder wird aus der Liste aller Restriktionsverletzungen eine Datenstruktur erzeugt, diesmal jedoch nach Zuteilungen indiziert. Bevor eine Zuteilung sich in ihrem Bereich visualisiert, überprüft sie in konstanter Zeit, ob sie an Restriktionsverletzungen beteiligt ist. Wenn ja, dann wird die entsprechende Restriktionsverletzung aufgefordert, sich zu visualisieren. Falls mehrere Verletzungen gleichzeitig dargestellt werden sollen, wird eine spezielle Methode zur Kollisionsauflösung aufgerufen. Diese versucht, den Konflikt entsprechend den in Kapitel 3.2 spezifizierten Mechanismen aufzulösen. Für diesen Fall ist in der Wissensbasis auch Wissen über die Kombinierbarkeit einzelner Visualisierungsmethoden vorhanden, bzw. konkrete Strategien, diese zu kombinieren.

Welche Methode zur Visualisierung gewählt wird, determiniert die Wissensbasis und ist im allgemeinen von der Instanz, bzw. Klasse der Regel abhängig. Für Restriktionsverletzungen sind spezielle Visualisierungsmethoden definiert, die zum Beispiel auch die Bewertung auszudrücken können. Die Visualisierungsmethode draw-background-color drückt zum Beispiel die Bewertung durch eine Variation des Farbtons der ursprünglichen Farbe aus, also helle Farbtöne für leichte, dunklere für schwere Restriktionsverletzungen.

In der Standarddarstellung werden alle Restriktionsverletzungen mittels einer roten Hintergrundfarbe visualisiert. Soll zum Beispiel eine Verletzung der Regel „Ist-Stunden möglichst nah an Soll-Stunden“ in Abweichung von der Standardvisualisierung durch die Symbole *** dargestellt werden, so müßte das wie folgt spezifiziert werden:

```
(specify-in-kb FOR $visualization-method  
  OF $c_check-actual-versus-intended-working-hours  
  REGARDING-PROBLEM *nurse-scheduling*  
  REGARDING-USER *default-user*  
  VALUE draw-symbol  
  ADDITIONAL-VALUES '((SYMBOL ***)  
    (HORIZONTAL-JUSTIFICATION :left)))
```

Dies ist also ein Beispiel für eine instanzspezifische Visualisierung einer Visualisierungsklasse. Die Eigenschaft „visualization-method“ einer Verletzung dieser Regel von der Wissensbasis mit dem Wert draw-symbol belegt. Soll diese Verletzung nun in einer Zelle der Tabelle visualisiert werden, so wird diese Methode wie beschrieben mit der Verletzung als Objekt, der Tabelle als Darstellungskomponente, der Zelle als Bereich dieser Darstellungskomponente und den zusätzlichen Werten aufgerufen. Die Methode draw-symbol akzeptiert eine Reihe von zusätzlichen Werten, unter anderem die spezifizierten; wäre kein zusätzliches Symbol übergeben worden, so hätte sie versucht, die Restriktionsverletzung zu verbalisieren.

Sollen Restriktionsverletzungen wie in Kapitel 3 erläutert, zusätzlich oder nur über das entsprechende Attribut visualisiert werden, so kann auch das in der Wissensbasis spezifiziert werden. Im Beispielsfall soll eine Verletzung der Regel „Ist-Stunden

möglichst nah an Soll-Stunden“ nur über das Attribut visualisiert werden (zusätzlich muß noch spezifiziert werden, um welches Attribut es sich handelt).

```
(specify-in-kb FOR $attributes-only-p  
  OF $c_check-actual-versus-intended-working-hours  
  REGARDING-PROBLEM *nurse-scheduling*  
  REGARDING-USER *default-user*  
  VALUE t)
```

Je mehr Wissen über eine Randbedingung vorhanden ist, desto besser ist die resultierende Darstellung. In der Standardeinstellung werden alle Regeln der Applikation als Visualisierungsklasse „Constraint“ behandelt. Wird jedoch genauer spezifiziert, um welchen Regeltyp es sich handelt, so kann die Oberfläche, wie in Kapitel 3 beschrieben, Verletzungen dieser Regel spezifischer und damit besser visualisieren.

5.2.4 Schnittstelle

Die Schnittstelle definiert eine Reihe von Methoden, die ein Zuordnungswerkzeug implementieren muß, um an das Präsentationstool angeschlossen werden zu können. Diese Methoden sind in zwei Gruppen unterteilt, zum einen den Methoden, die zur Darstellung eines Zuordnungsproblems benötigt werden, zum anderen Methoden, die der interaktiven Planung, bzw. der Unterstützung der interaktiven Planung dienen. Für manche weitergehende Funktionalität müssen eventuell noch zusätzliche Schnittstellenfunktionen definiert werden. Sämtliche Grundfunktionen sind aber mit den hier angegebenen Methoden abgedeckt.

Schnittstellenfunktion	Zusätzliche Parameter	Rückgabewert	Bemerkung
Methoden zur Darstellung von Zuordnungsproblemen			
GetAllAssignments	-	Alle Zuteilungen	Wird für die Datenstruktur in 5.2.3.7 benötigt
GetAllConstraints	-	Alle Regeln	Liefert alle Regeln, so daß der Benutzer einstellen kann, wie sie visualisiert werden sollen
GetAllConstraintViolations		Alle Regelverstöße	Wird für die Datenstruktur in 5.2.3.8 benötigt
GetAllInstancesOfClass	Klassenname	Alle Objekte einer Klasse	Liefert alle Instanzen einer Klasse
GetClassOfObject	Objekt	Klassenname	Liefert den Klassennamen eines Objekts
GetAttributeValue	Objekt, Attributname	Wert	Liefert den Wert eines Attributs
Methoden zur interaktiven Planung			
MakeAssignment	Angebotsobjekt, Nachfrageobjekt	Boolean	Erzeugt neue Zuteilung
RevokeAssignment	Zuteilung	Boolean	Nimmt Zuteilung zurück
CreateObjectOfClass	Klassenname	Neues Objekt einer Klasse	Erzeugt neues Objekt
DeleteObject	Objekt	Boolean	Löscht ein Objekt
SetAttributeValue	Objekt, Attributname, Wert	Boolean	Setzt den Wert eines Attributs
ProposeDemandObjects	Angebotsobjekt	Nachfrageobjekte	Schlägt Nachfrageobjekte vor
ProposeOfferObjects	Nachfrageobjekt	Angebotsobjekte	Schlägt Angebotsobjekte vor
AssignDemandObjects	Angebotsobjekt	Zuteilung	Sucht passendes Nachfrageobjekt und erzeugt neue Zuteilung

AssignOfferObjects	Nachfrageobjekt	Zuteilung	Sucht passendes Angebotsobjekt und erzeugt neue Zuteilung
EvaluateAssignment	Zuteilung	Regelverstöße	Liefert die Regelverstöße, an denen eine Zuteilung beteiligt ist
ProposeAlternatives	Zuteilung	Zuteilungen	Zeigt Alternativen für ein Zuordnung auf
FixAssignment	Zuteilung	Boolean	Legt eine Zuteilung fest
UnfixAssignment	Zuteilung	Boolean	Löst die Festlegung einer Zuteilung auf

Tabelle 19: Definition der Schnittstelle

In der Bibliothek ist eine abstrakte (virtuelle) Definition einer Schnittstelle enthalten; ein konkretes Zuordnungswerkzeug muß also eine eigene Interface-Klasse von dieser abstrakten Basisklasse ableiten und die abstrakten Methoden dieser Basisklasse implementieren. Innerhalb des Präsentationstools wird dann ausschließlich über dieses Interface mit dem Zuordnungswerkzeug kommuniziert, die Oberfläche ist prinzipiell vollkommen unabhängig und funktioniert mit jedem Zuordnungswerkzeug, das die Schnittstelle unterstützt. Somit ist gewährleistet, daß Oberfläche und Zuordnungswerkzeug getrennt sind, so daß sich Änderungen an einem der beiden Bestandteile nicht auf den jeweils anderen auswirken.

5.2.5 Präsentationstool

Das Präsentationstool hat nur wenige Aufgaben; die primäre ist die Erzeugung von Visualisierungsklassen und die Belegung der Eigenschaften der Visualisierungsobjekte. Sämtliche Instanzen von Visualisierungsklassen werden über eine Standardfunktion definiert, die als Parameter den Namen der Visualisierungsklasse, die Problem-Id und die Benutzer-Id erwartet.

```
(defmethod make-visualization-class(visualization-class-name
                                   (problem-id *pt-default-problem-id*)
                                   (user-id *pt-default-user-id*))
  ...)
```

Anhand dieser Informationen kann das Objekt dann anhand der Einträge in der Wissensbasis erzeugt und initialisiert werden.

5.2.6 Fazit

In diesem Abschnitt soll das vorliegende Konzept anhand der prinzipiellen Anforderungen an Software evaluiert werden. Allerdings soll primär die Qualität der erzeugten Oberflächen evaluiert werden, nicht die Implementierung des vorgestellten Konzepts. Kriterien wie Benutzbarkeit beziehen sich also auf die Oberfläche, nicht auf die Frage, wie einfach die Konfiguration eines Zuordnungsproblems in der Wissensbasis ist.

Kriterium	Urteil	Grund
Adaptierbarkeit	Ja	Verwendung von OOD und OOP
Adäquate Funktionalität	?	Konnte nicht umfassend evaluiert werden, die Ergebnisse in Kapitel 6 sind jedoch sehr positiv
Benutzbarkeit	Ja (potentiell schlechter)	Mangelnde Designexpertise der Entwickler wird kompensiert, allerdings ist eine speziell von Hand entwickelte Oberfläche in der Regel besser
Durchschaubarkeit	Ja	Trennung von Anwendung und Interface, übersichtliche Klassenbibliothek und somit eine klare Struktur
Leistung	Ja (potentiell schlechter)	Konnte nicht umfassend evaluiert werden, die Ergebnisse in Kapitel 6 sind jedoch sehr positiv
On-Budget	Ja	Besser, da übersichtlicher
On-Time	Ja	Design-, Implementierungs- und Testphase sind stark verkürzt
Portabilität	Ja	Ja, da nur das Konzept portiert werden muß
Robustheit	Nein	Nicht berücksichtigt
Schutz vor Informations-verlust	Nein	Nicht berücksichtigt
Sicherheit, Datenschutz	Nein	Nicht berücksichtigt
Verlässlichkeit	Ja	Es werden keine neuen Komponenten programmiert, sondern bewährte Komponenten konfiguriert; somit ist das Fehlerpotential geringer
Wiederverwendbarkeit	Ja	Verwendung von OOD und OOP

Tabelle 20: Evaluation des Konzepts anhand der Anforderungen an Software

Insgesamt entspricht das vorgestellte Konzept fast allen Anforderungen; die Vorteile einer solchen Generierung sind evident; allerdings wäre für eine sinnvolle Beurteilung des vorliegenden Konzepts eine Generierung von Oberflächen für verschiedene Zuordnungsprobleme und eine Evaluation sowohl des Arbeitsaufwands für diese Erzeugung als auch der erzeugten Oberflächen notwendig. Eine solche Untersuchung ist allerdings mit großen Zeitaufwand verbunden und im Rahmen dieser Diplomarbeit nicht praktikabel.

6 Vorstellung und Evaluation einer erzeugten Oberfläche

In diesem Kapitel soll eine für die Dienstplanung des Pflegepersonals der anästhesiologischen Klinik der Universität Würzburg generierte Oberfläche vorgestellt werden. Anschließend soll ein Versuch einer Pflegekraft, mit dieser Oberfläche einen Dienstplan zu erstellen, evaluiert werden.

6.1 Beschreibung der Oberfläche

Für die Dienstplanung des Pflegepersonals der anästhesiologischen Klinik der Universität Würzburg wurde mit einem Prototyp der im 5. Kapitel beschriebenen Lösung eine Oberfläche implementiert. Gleichzeitig wurde die Dienstplanung in der Zuordnungsschell Coke [Poock 95] als Zuordnungsproblem modelliert, so daß auch Möglichkeiten zur automatischen Berechnung und Optimierung eines Dienstplans zur Verfügung standen. Sowohl der Prototyp der Lösung als auch die erzeugte Oberfläche basieren auf Allegro Common Lisp für Microsoft Windows, somit ist die Oberfläche also nur unter Microsoft Windows lauffähig. Mindestausstattung ist ein Intel Pentium Prozessor mit 200 Mhz und 64 MB Hauptspeicher. Im diesem Abschnitt sollen nun kurz die wichtigsten Aspekte der Oberfläche vorgestellt werden, wobei eine gewisse Vertrautheit mit dem Standardverhalten von Applikationen unter Microsoft Windows vorausgesetzt wird. Abbildung 38 gibt einen Gesamtüberblick über die wichtigsten Komponenten der Oberfläche.

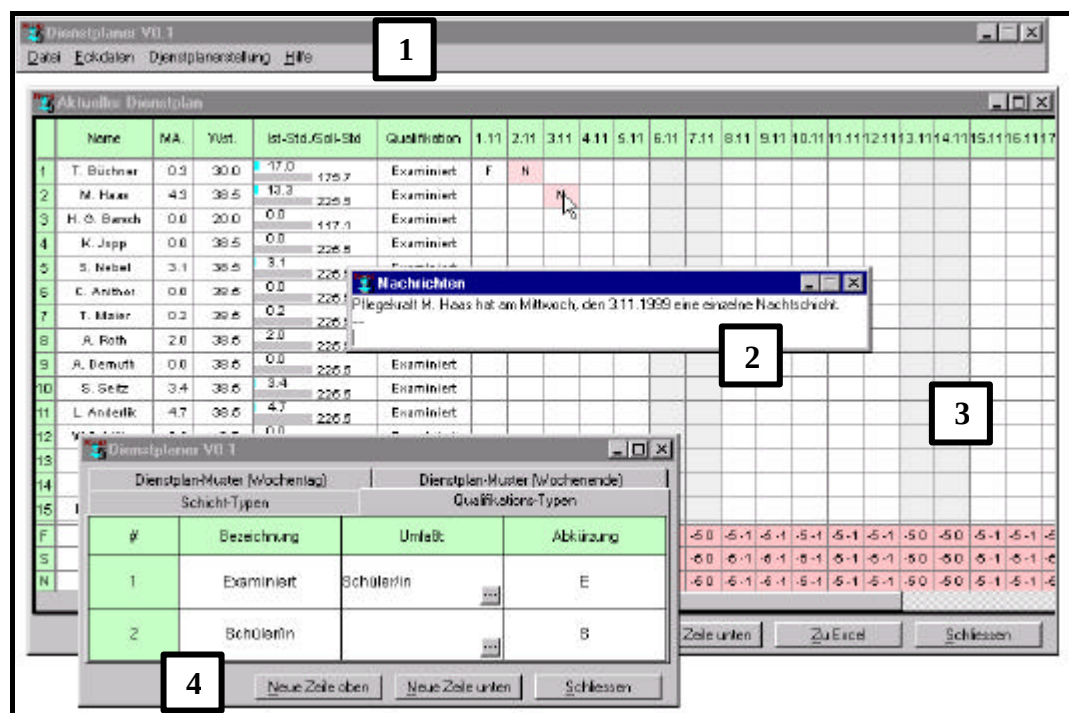


Abbildung 38: Gesamtüberblick über die wichtigsten Komponenten der Oberfläche

Zur Erstellung eines Dienstplans wird ein Benutzer, solange er nicht die Arbeit an einem existierenden Plan fortsetzt, üblicherweise über das Hauptmenü (1) die Dialoge zur Festlegung der Eckdaten des Dienstplans aufrufen, also zum Beispiel den Planungszeitraum oder die Qualifikationsstufen (4) einstellen. Sobald er diese Rahmendaten festgelegt hat, kann er die Darstellung des Dienstplans (3) dazu benutzen, den Plan zu erstellen. Dabei werden Meldungen an den Benutzer immer auch explizit im Nachrichtenfenster (2) verbalisiert.

6.1.1 Nachrichtenfenster

Tabelle 3 fordert, sämtliche Aktionen des Benutzers zu kommentieren und dabei Gründe für einen Mißerfolg darzulegen, bzw. die erfolgreiche Durchführung zu bestätigen. Für diesen Zweck besitzt die Oberfläche ein Nachrichtenfenster, das ständig aktuelle Meldungen enthält und zusätzlich dazu dient, dem Benutzer Bedienungshinweise zu geben.

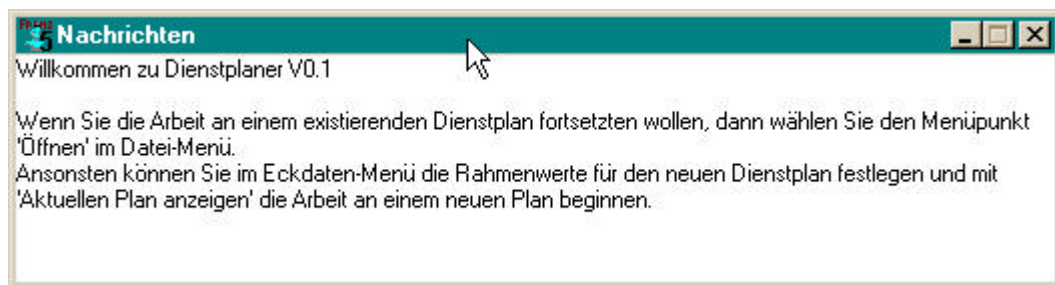


Abbildung 39: Nachrichtenfenster

6.1.2 Hauptmenü

Zentraler Punkt der Oberfläche ist das Hauptmenü, von dem aus sämtliche Optionen aufgerufen werden können.

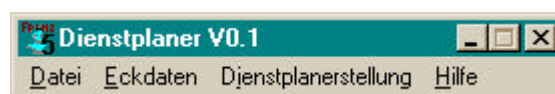


Abbildung 40: Hauptmenü der Oberfläche

Diese Optionen sind in vier Untermenüs mit folgendem Inhalt unterteilt:

- ?? Datei: Optionen zur Verwaltung von Dienstplänen
- ?? Eckdaten: Optionen zur Festlegung der Rahmenwerte des aktuellen Dienstplans
- ?? Dienstplanerstellung: Optionen zur Durchführung der eigentlichen Planung
- ?? Hilfe: Informationen über das Programm (werden im Rahmen dieser Arbeit nicht beschrieben)

6.1.2.1 Verwaltung von Dienstplänen

Die Optionen zur Verwaltung von Dienstplänen entsprechen dem Standardmenü „Datei“ einer Microsoft Windows Applikation, um einem Benutzer das gewohnte Bild und die vertrauten Möglichkeiten bieten zu können. Da alle Optionen dieses Menüs auch in ihrem Verhalten diesem Standard entsprechen, sollen sie hier nicht weiter besprochen werden.

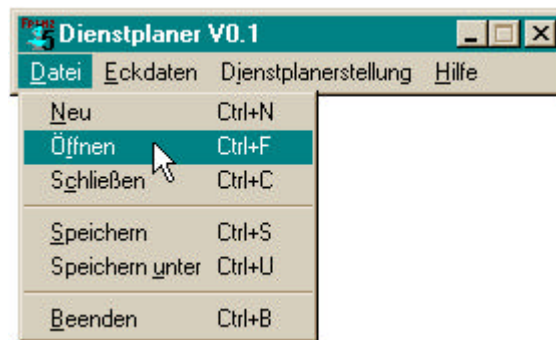


Abbildung 41: Optionen zur Verwaltung von Dienstplänen

6.1.2.2 Festlegung der Eckdaten

Zur Erstellung eines Dienstplans kann eine Reihe von Eckdaten für den aktuellen Planungszeitraum festgelegt werden.

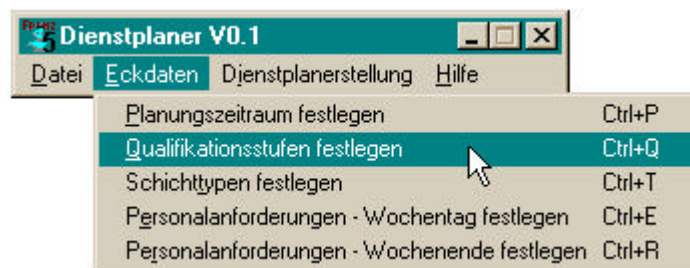


Abbildung 42: Optionen zur Festlegung der Eckdaten des Plans

Planungszeitraum festlegen

Der Planer kann den Anfangstag des Planungszeitraum festlegen, der ansonsten auf das aktuelle Datum gesetzt wird, und angeben, wie viele Tage der Planungszeitraum umfassen soll (normalerweise 42 Tage, also 6 Wochen). In Abbildung 43 wird der Planungszeitraum also auf den Anfangstag Dienstag, den 26.10.1999 mit einer Planungsdauer von 30 Tagen festgesetzt.

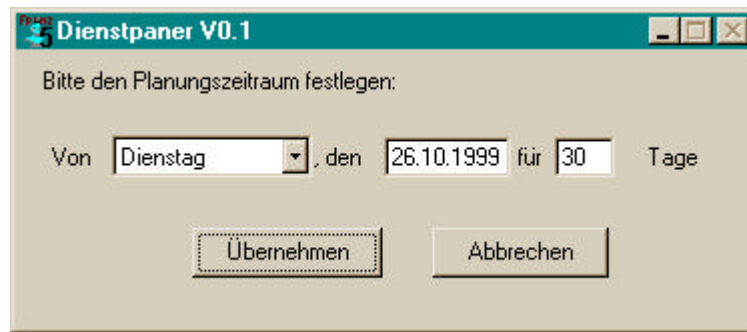


Abbildung 43: Festlegung des Planungszeitraums

Qualifikationsstufen festlegen

Ein Benutzer kann beliebig viele eigene Qualifikationsstufen oder -typen definieren. Für jeden dieser Schichttypen kann ein Planer dann eine Reihe von anderen Stufen spezifizieren, die dieser Schichttyp umfaßt. In Abbildung 44 umfaßt zum Beispiel die Qualifikation „Schichtleiter(in)“ die Qualifikation „Examiniert“. Somit verfügt jede Pflegekraft, die als „Schichtleiter(in)“ qualifiziert ist, automatisch auch über die Qualifikation „Examiniert“. Eine Qualifikationsstufe kann beliebig viele andere Stufen miteinschließen. Die Abkürzung einer Qualifikationsstufe kann dazu benutzt werden, den Personalbedarf einer Schicht im Dienstplan darzustellen, E2 S1 heißt dann, zwei Pflegekräfte mit der Qualifikation „Examiniert“ und eine mit der Qualifikation „Schüler(in)“. Meistens wird die Abkürzung für die Qualifikationsstufe jedoch aus Platzgründen weggelassen, also der Personalbedarf zum Beispiel nur mit 2-1 angegeben, da ein Planer aus Erfahrung bereits weiß, welche Stufe mit der jeweiligen Zahl gemeint ist. Schließlich ändern sich die Qualifikationsstufen einer Station im allgemeinen nur äußerst selten.

Dienstplan-Muster (Wochentag)		Dienstplan-Muster (Wochenende)	
Schicht-Typen		Qualifikations-Typen	
#	Bezeichnung	Umfaßt:	Abkürzung
1	Schichtleiter(in)	Examiniert	SL
2	Examiniert	Schüler(in)	E
3	Schüler(in)		S
4	Praktikant(in)		

Neue Zeile oben Neue Zeile unten Schliessen

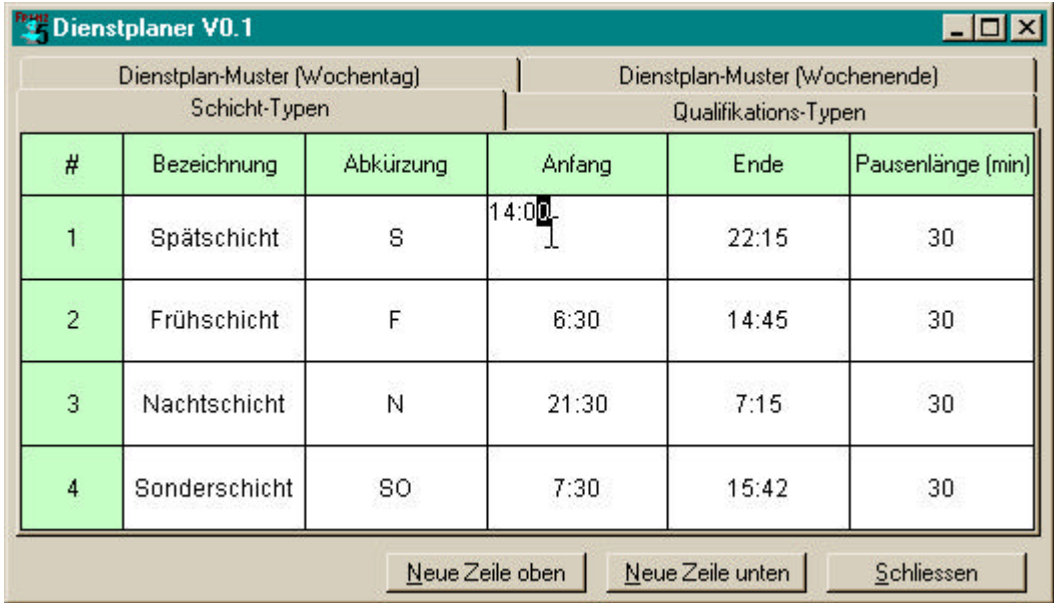
Füge Examiniert hinzu
Füge Praktikant(in) hinzu
Füge Schichtleiter(in) hinzu

Abbildung 44: Festlegung der Qualifikationsstufen

Schichttypen festlegen

Der Planer kann beliebig viele eigene Schichttypen definieren. Dabei werden Schichttypen so spezifiziert, wie es im Pflegebereich üblich ist, das heißt über (Schicht-) Anfang, (Schicht-) Ende und über die Dauer der Pause. Endet der Schichttyp zu einem früheren Zeitpunkt als er anfängt, so wird die Schicht als Nachtschicht aufgefaßt und das Schichtende auf den nächsten Tag bezogen. In Abbildung 45 bezieht sich das Ende der Nachtschicht auf den Morgen des folgenden Tages. Um Verwechslungen auszuschließen, werden nur Zeitangaben von 0:00 bis 23:59 Uhr benutzt. Die Abkürzung eines Schichttyps wird dazu benutzt, sie auf dem Dienstplan darzustellen.

Bei der Pause wird davon ausgegangen, daß sie weder direkt am Anfang noch direkt am Ende einer Schicht liegt, was für Regeln in Bezug auf den Mindestabstand zweier Schichten relevant ist.



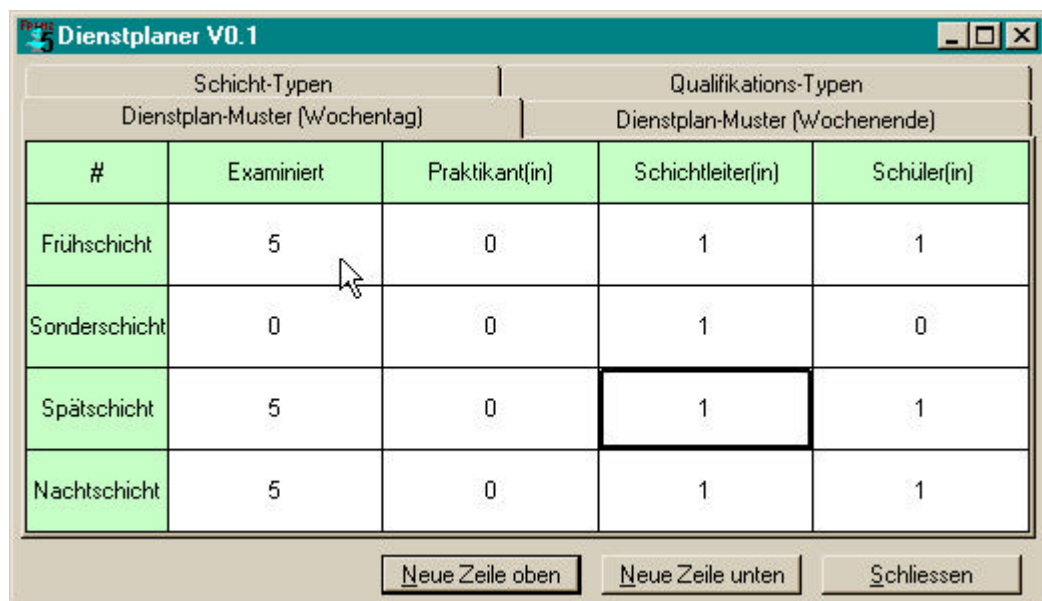
Dienstplan-Muster (Wochentag)			Dienstplan-Muster (Wochenende)		
Schicht-Typen			Qualifikations-Typen		
#	Bezeichnung	Abkürzung	Anfang	Ende	Pausenlänge (min)
1	Spätschicht	S	14:00	22:15	30
2	Frühschicht	F	6:30	14:45	30
3	Nachtschicht	N	21:30	7:15	30
4	Sonderschicht	SO	7:30	15:42	30

Neue Zeile oben Neue Zeile unten Schliessen

Abbildung 45: Festlegung der Schichttypen

Personalanforderungen festlegen

Ein Planer kann die Muster- oder Standard-Personalanforderungen differenziert nach Schichttyp und Qualifikationsstufe festlegen (Abbildung 46). Zusätzlich wird noch zwischen einem Muster für einen normalen Wochentag und dem Wochenende unterschieden. Die erstellten Muster definieren dann den Personalbedarf eines jeden Tags des Planungszeitraums; für jeden einzelnen Tag kann jedoch direkt in der Darstellung des Dienstplans ein spezieller Personalbedarf spezifiziert werden, der dann das Standard-Muster überschreibt.



Schicht-Typen		Qualifikations-Typen		
Dienstplan-Muster (Wochentag)		Dienstplan-Muster (Wochenende)		
#	Examiniert	Praktikant(in)	Schichtleiter(in)	Schüler(in)
Frühschicht	5	0	1	1
Sonderschicht	0	0	1	0
Spätschicht	5	0	1	1
Nachtschicht	5	0	1	1

Buttons: Neue Zeile oben, Neue Zeile unten, Schliessen

Abbildung 46: Festlegung der Standard-Personalanforderungen

6.1.2.3 Erstellung von Dienstplänen

Eigentlicher Zweck der Oberfläche ist die Erstellung von Dienstplänen. Dazu gibt es im Menü „Dienstplanerstellung“ eine Reihe von Optionen.

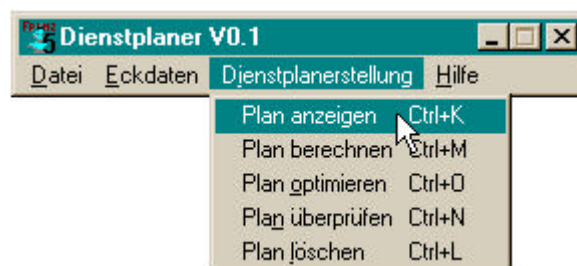


Abbildung 47: Optionen zur Erstellung von Dienstplänen

Plan anzeigen

Der Aufruf dieser Option führt zur Anzeige des aktuellen Dienstplans in einem neuen Planungsfenster. Wurde ein bestehender Plan geladen, so wird anhand der Eckdaten ein neuer Plan erzeugt und dargestellt. Abbildung 48 zeigt ein Beispiel für die Darstellung eines neuen, unausgefüllten Plans, Abbildung 49 ein Beispiel eines (nahezu) fertigen Plans. Die Darstellung eines Dienstplans verfügt über einen Großteil der in Tabelle 2 aufgeführten Funktionalität zur Individualisierung der Darstellung – jede Zeile oder Spalte lässt sich in ihrer Größe anpassen, die Tabelle kann nach einer beliebigen Spalte sortiert werden, die Reihenfolge der Zeilen und Spalten kann beliebig vertauscht werden usw.

Plan errechnen

Beim Aufruf dieser Option wird vom Programm ein Dienstplan errechnet, den der Benutzer dann noch seinen Wünschen anpassen kann. Die Berechnung eines Plans für 42 Tage mit 40 Pflegekräften dauert allerdings mehrere Minuten; die Restdauer wird dabei durch eine Verlaufsanzeige dargestellt.

Plan optimieren

Diese Funktion versucht, den bestehenden Plan zu optimieren; intern wird versucht, das Gesamtgewicht der Restriktionsverletzungen zu optimieren.

Plan überprüfen

Diese Funktion überprüft, welche Randbedingungen durch die aktuelle Zuordnung verletzt sind. Da diese Überprüfung unter normalen Umständen ständig in Echtzeit erfolgt, wird diese Funktion im allgemeinen nicht gebraucht, sie ist nur für sehr langsame Rechner gedacht, die eine Echtzeitberechnung nicht durchführen können.

Plan löschen

Der Aufruf dieser Option löscht die Zuordnung, d.h. alle Zuteilungen aller Pflegekräfte zu Schichten werden zurückgenommen.

Aktueller Dienstplan																			
	Name	MA.	Wst.	Ist-Std./Soll-Std	Qualifikation	28.10	29.10	30.10	31.10	1.11	2.11	3.11	4.11	5.11	6.11	7.11	8.11	9.11	10.11
1	T. Büchner	0.3	30.0	0.3 128.6	Examiniert														
2	M. Haas	4.3	38.5	4.3 165.0	Examiniert														
3	H. G. Barsch	0.0	20.0	0.0 85.7	Examiniert														
4	K. Jopp	0.0	38.5	0.0 165.0	Examiniert														
5	S. Nebel	3.1	38.5	3.1 165.0	Examiniert														
6	C. Anithor	0.0	38.5	0.0 165.0	Examiniert														
7	T. Maier	0.2	38.5	0.2 165.0	Examiniert														
8	A. Roth	2.0	38.5	2.0 165.0	Examiniert														
9	A. Demuth	0.0	38.5	0.0 165.0	Examiniert														
10	S. Seitz	3.4	38.5	3.4 165.0	Examiniert														
11	L. Anderlik	4.7	38.5	4.7 165.0	Examiniert														
12	W. Schöller	0.0	9.5	0.0 40.7	Examiniert														
13	M. Nolte	0.0	38.5	0.0 165.0	Examiniert														
14	J. Schöll	3.2	38.5	3.2 165.0	Examiniert														
15	B. Bittkow	2.9	30.0	2.9 128.6	Examiniert														
16	A. Weber	1.3	38.5	1.3 165.0	Examiniert														
17	K. Hirsch	1.3	38.5	1.3 165.0	Examiniert														
18	P. Paul	2.9	30.0	2.9 128.6	Examiniert														
F						-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1	-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1
S						-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1	-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1
N						-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1	-5-1	-5-1	-5-0	-5-0	-5-1	-5-1	-5-1

Abbildung 48: Darstellung eines leeren Dienstplans

Aktueller Dienstplan																								
	Name	MA.	Wst.	Ist-Std./Soll-Std	Qualifikation	28.10	29.10	30.10	31.10	1.11	2.11	3.11	4.11	5.11	6.11	7.11	8.11	9.11	10.11	11.11	12.11	13.11	14.11	15.11
24	H. Baier	4.1	38.5	133.7 137.5	Examiniert	N	N							S	S	S	N		S		S	S	S	N
25	S. Albert	0.0	38.5	131.7 137.5	Examiniert	S		S	S		S	S		S			S		S	S	S	S	S	S
26	R. Dessieu	2.4	38.5	141.9 137.5	Examiniert	S	S	S	S		S	S		S			S			S	S	S	S	S
27	H. Büttner	1.1	38.5	140.6 137.5	Examiniert	S	S	S	S		S	S	S	S				S		S	S	S	S	
28	A. Blank	0.0	9.5	31.0 33.9	Examiniert			S																
29	S. Barsch	3.6	38.5	135.4 137.5	Examiniert	S	S		S	S	S	S	S	S										
30	M. Kunad	3.6	30.0	113.3 107.1	Examiniert	S	S			S	S	S	S			S	N			S		F	F	
31	L. Göller	4.3	38.5	136.1 137.5	Examiniert		S	S	S	S	S	S	S	S	S	S	S							S
32	U. Leuchs	0.0	38.5	133.0 137.5	Examiniert	F	F			S		F	S		F	F	S	S	S		F	F	F	S
33	G. Werner	1.0	38.5	132.7 137.5	Examiniert	F	F	F	F	S		F	F		F	F	S	S	S		F			S
34	B. Schlund	0.0	38.5	87.7 137.5	Examiniert	F		N			F	F	F	N		F					F			
35	K. Wolff	1.2	38.5	140.7 137.5	Examiniert		F	F		F	F	F	F	F		F	F	F	F	F	F			F
36	S. Henn	3.3	38.5	144.1 137.5	Examiniert			F		F	F		F	F	F	F	F	F	F	F				F
37	I. Klement	0.0	38.5	140.7 137.5	Examiniert		F	F	F	F	F		F	F			F	F	F	F		N		F
38	M. Wehnert	1.3	38.5	61.8 137.5	Schüler/in	S				N	N	N	N	N			S							
39	J. Lassonczyk	2.1	38.5	118.3 137.5	Schüler/in		S			S	S	S	S	S				S	S	S	S			S
40	D. Kaewel	0.0	38.5	140.7 137.5	Examiniert	F	F	F	F	F	F	N		F			F	F	F	F		F	F	F
41	B. Ritter	1.9	38.5	133.6 137.5	Schüler/in	F	F			F	F	F	F	F			F	F	F	F	F			F
F						00	00	00	-10	00	10	00	00	00	-10	00	00	00	00	00	00	00	00	00
S						00	00	10	10	00	10	10	00	10	10	00	00	-10	-10	00	00	00	00	00
N						00	00	00	00	00	-10	00	00	00	00	00	00	00	00	00	00	00	00	00

Abbildung 49: Darstellung eins fertigen Dienstplans

6.1.3 Unterstützung der interaktiven Planung

Eigentlicher Zweck der Oberfläche ist es, einen Dienstplan zu erstellen, wobei in dieser Arbeit nur die interaktive Erstellung behandelt wird. Dies unterstützt die Oberfläche mit folgender Funktionalität:

Verwaltung der Objekte der Zuordnung

Pflegekräfte können neu in den Dienstplan eingetragen, bzw. aus dem Dienstplan gelöscht werden, indem zu der Tabelle neue Zeilen hinzugefügt werden, bzw. bestehende Zeilen gelöscht werden. Sämtliche Felder des Plans sind editierbar, bis auf die Ist/Soll-Stunden, die sich wiederum aus anderen Eintragungen berechnen, somit können auch die Attribute der Pflegekräfte verändert werden.

Plantage werden durch die Verwaltung des Planungszeitraums erzeugt, bzw. gelöscht.

Verwaltung der Zuteilungen

Für die eigentliche Planung werden sowohl die Tastatur als auch Zeigegeräte wie die Maus unterstützt, obwohl letztere angesichts der großen Datenmengen unpraktisch sind. Um ein Zuteilung durchzuführen, setzt der Benutzer mit den Pfeiltasten das Tastaturfokus auf eine Zelle, bzw. klickt eine Zelle der Tabelle mit einem Zeigegerät an, und gibt ein Schichtkürzel ein.

Nach Drücken der Eingabetaste, bzw. Auswahl einer anderen Zelle überprüft die Oberfläche in Echtzeit, ob durch die veränderte Zuteilung Randbedingungen verletzt wurden und zeigt diese dann eventuell durch eine rote Hintergrundfarbe der Zelle an. Dabei gibt der Farbton über die Schwere der Verletzung Auskunft, eine leichte Verletzung wird hellrot, eine schwere entsprechend dunkelrot angezeigt. Auch die Anzeige der Ist-Stundenzahl der Pflegekraft und der Besetzung der Schicht wird aktualisiert.

Zusätzlich zu der internen Visualisierung von Regelverstößen innerhalb der Zuordnungsdarstellung findet auch eine externe Visualisierung im Nachrichtenfenster statt (Abbildung 50). Dabei werden immer die Restriktionen angezeigt, die mit der Stelle der Zuordnung assoziiert sind, die sich gegenwärtig unter dem Mauszeiger befindet.

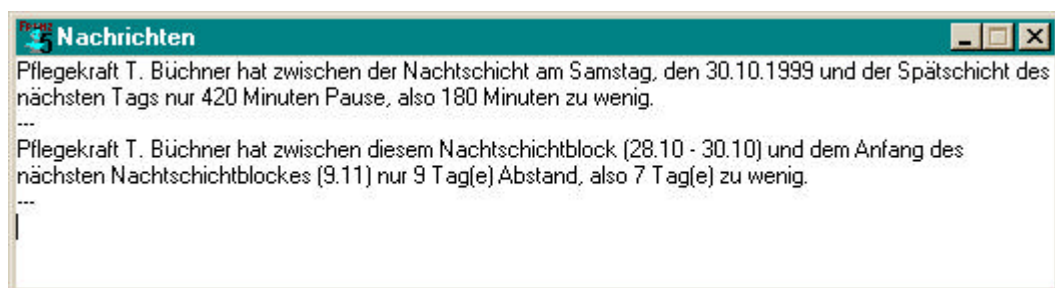


Abbildung 50: Externe Visualisierung einer Restriktionsverletzung

Zuteilungen können mittels Drag&Drop verschoben werden; der Benutzer kann ein Schichtsymbol anklicken und verschieben, die resultierenden Änderungen werden sofort angezeigt.

Folgende weitere Funktionalität, die für die Erstellung eines Dienstplans notwendig ist, wird unterstützt:

- ?? Markierung von Urlaubstagen (werden grün dargestellt).
- ?? Markierung von Tagen, an denen die Pflegekraft nicht Dienst haben sollten (werden grau dargestellt).
- ?? Spezifizierung des Personalbedarfs einzelner Tage, falls diese vom Muster abweichen.

6.2 Evaluation

Die im letzten Abschnitt beschriebene Oberfläche wurde vom Pflegepersonal der anästhesiologischen Klinik der Universität Würzburg getestet. Dabei wurde einer Pflegekraft, die nur über geringe Computererfahrung verfügt, eine kurze Einführung (ca. 3 Minuten) gegeben. Anschließend sollte sie versuchen, selbständig einen Dienstplan zu erstellen. Allerdings war es nicht das Ziel, einen vollständigen Dienstplan zu erstellen, sondern die prinzipielle Brauchbarkeit zu evaluieren und auszuloten, ob der nächste anstehende Dienstplan bereits mit der Oberfläche erstellt werden kann oder ob die Qualität der Oberfläche dazu (noch) nicht ausreicht.

Die automatische Planung und Optimierung wurde bei der Evaluation nicht benutzt, da sie nur ein Nebenaspekt dieser Arbeit ist. Insgesamt wurden bezüglich der Hauptkriterien folgende Erfahrungen gemacht:

Geringe Einarbeitungszeit

Da die Oberfläche der gewohnten Umgebung eines Planers, also dem für die Planung von Hand verwendeten Formular, nachempfunden ist, betrug die Eingewöhnungszeit erwartungsgemäß nur wenige Minuten. Auch wurde diese Zeit vor allem zum Erlernen der Standardfunktionalität einer Benutzeroberfläche benötigt, also zum Beispiel zum Verständnis des Prinzips eines Pull-down-Menüs, des Anklickens mit der Maus und ähnlicher Aspekte. Alle wesentlichen Funktionen, wie die Festlegung der Schichttypen, der Qualifikationsstufen und der Muster des Personalbedarfs und die eigentliche Planung wurden auf Anhieb richtig eingesetzt. Das Ziel der schnellen Einarbeitung erfüllt die Oberfläche somit.

Hohes Bedienungstempo

Ein hohes Bedienungstempo ist essentiell für die interaktive Erstellung eines Dienstplans, da für einen typischen Plan circa 1000 Zellen der Tabelle auszufüllen sind. Zwar wurden die Berechnungs- und Darstellungsaufgaben, die mit Veränderungen des Dienstplans verbunden sind, so optimiert, daß sie durchgängig weniger als 1 Sekunde dauern und somit einem hohen Bedienungstempo nicht im Wege stehen. Jedoch benötigt die Zuteilung einer Pflegekraft zu einer Schicht in der Regel 3 Tastendrucke, was sich bei der

großen Zahl der nötigen Eingaben als zuviel erwies. Die Planung mit der Maus ist viel zu aufwendig, allerdings liegt das in der Natur dieses Eingabegeräts. Deswegen müssen hier Mechanismen für eine schnellere Dateneingabe gefunden werden, mit dem Ziel, eine Eingabe mit nur einem Tastendruck durchführen zu können. Das Verschieben einzelner Zuteilungen mittels Drag&Drop ist sehr effektiv, wird aber nur selten benötigt. Das Kriterium des hohen Bedienungstempos erfüllt die Oberfläche derzeit also nur bedingt, hier besteht noch weiterer Optimierungsbedarf; es liegen noch keine genauen Zeiten vor, die eine Erstellung eines Dienstplans von Hand (ca. 10h-15h) mit der Erstellung per Computer vergleichen, da bei der Evaluation nur ein Teil eines Planes erstellt wurde.

Geringe Fehlerrate

Die Fehlerrate war relativ niedrig und die auftretenden Fehler waren zum größten Teil auf die fehlende Einarbeitung in die Oberfläche, bzw. generell Computerprogramme zurückzuführen. Primär traten also nebensächliche Fehler wie „Daneben-Klicken“ auf. Problematisch war lediglich die mangelnde Toleranz der Oberfläche bei fast richtigen Eingaben, so wurde „14.30“ als Uhrzeit nicht akzeptiert (nur „14:30“), genauso Schichtkürzel wie „ S“ (nur „S“). Insgesamt ist die Fehlerrate erfreulich gering und sollte sich mittels geringer Verbesserungen noch weiter senken lassen.

Subjektive Zufriedenheit des Benutzers

Die Benutzerin war mit dem Programm sehr zufrieden und freut sich schon darauf, das Programm weiter einzusetzen, also den nächsten Dienstplan unter realen Bedingungen mit der Oberfläche zu erstellen. Der Testlauf verlief problemlos, besonderes Gefallen fand die ständige Aktualisierung der Anzeige der Ist-Stunden und der Personalbedarfsanzeige der Schichten, da diese Berechnungen bei der Planung von Hand besonders mühsam sind. Auch die kontinuierliche Überprüfung der Regeln wurde als sehr hilfreich empfunden, da aufgrund der großen Datenmengen sehr leicht Fehler unterlaufen, ebenso die Sortierfunktionen, die es ermöglicht, die gesamte Besetzung einer Schicht nebeneinander darzustellen. Primärer Kritikpunkt war das noch zu umständliche Eingabeverfahren (siehe „Hohes Bedienungstempo“).

Mittel- bis langfristige Beibehaltung des Wissens

Leider konnte dieses Kriterium nicht überprüft werden, da nicht genügend Zeit zur Verfügung stand, einen zweiten Testlauf mit einigen Wochen Abstand durchzuführen. Allerdings spielt dieses Kriterium in Anbetracht der ohnehin kurzen Einarbeitungszeit von nur wenigen Minuten auch keine große Rolle.

Fazit

Insgesamt war das Resultat der Evaluation sehr erfreulich, lediglich kleinere Verbesserung insbesondere in Bezug auf die Dateneingabe sind noch nötig, dann sollte einem Alltagseinsatz nichts mehr im Wege stehen. Konkret wurde die Frage nach der Brauchbarkeit insgesamt sehr positiv beantwortet und die Entscheidung getroffen, den nächsten anstehenden Plan tatsächlich mit der Oberfläche zu planen. Allerdings wurde

die Oberfläche bisher nur in einer Klinik wenige Stunden eingesetzt, eine umfassende Evaluation über einen längeren Zeitraum und in mehreren Kliniken steht also noch aus, so daß hier kein allgemeingültiges Urteil gefällt werden kann. Mit Sicherheit kann jedoch jetzt schon gesagt werden, daß die hier vorgestellte Oberfläche einen deutlichen Fortschritt gegenüber der mühsamen Planung von Hand darstellt.

7 Zusammenfassung und Ausblick

7.1 Ergebnisse

Im Rahmen dieser Arbeit wurde ein Konzept entwickelt, Oberflächen zur interaktiven Lösung von Zuordnungsproblemen automatisch zu generieren. Dazu wurde in Kapitel 2 eine genaue Definition eines Zuordnungsproblems erarbeitet, wobei insbesondere ein System zur Typisierung von Randbedingungen (S.15) entworfen wurden, um eine differenziertere Diskussion dieses komplexen Begriffs zu ermöglichen.

Ausgehend von dieser abstrakten Definition wurden die Ansprüche an eine Oberfläche zur interaktiven Lösung von Zuordnungsproblemen formuliert. Dabei wurde konsequent auf bestehende Prinzipien auf dem Gebiet des Interface Designs geachtet.

Die wichtigsten Resultate dieser Untersuchung waren

- ?? die Definition einiger weniger Grundtypen der Zuordnungsdarstellung, mit denen sich zumindest die untersuchten Zuordnungsprobleme visualisieren lassen (S.23).
- ?? eine Reihe von prinzipiellen Möglichkeiten, Verletzungen von Randbedingungen je nach Typ effektiv zu visualisieren; ausgehend davon wurde eine sinnvolle Standardvisualisierung (S. 45) erarbeitet und ein Konzept entworfen, wie Designer und Endbenutzer die Darstellung im Rahmen der prinzipiellen Möglichkeiten einfach an ihre Bedürfnisse anpassen können (S.49).
- ?? Eine Aufstellung sinnvoller Interaktionsmöglichkeiten (S.52) und entsprechender Interaktionsformen, über die diese Interaktionsmöglichkeiten aufgerufen werden können.

Im 4. Kapitel wurden reale Zuordnungsprobleme untersucht. Die Hauptergebnisse stellen sich wie folgt dar:

- ?? Die Darstellungen der realen Zuordnungsprobleme lassen sich vollständig in die Grundtypen der Zuordnungsdarstellung einordnen.
- ?? Die Randbedingungen dieser Probleme lassen sich fast alle einem der in Kapitel 2 definierten Typen von Regeln zuordnen (einzige Ausnahme eine Regel der Dienstplanung für das ärztliche Personal). Die resultierende Visualisierung erscheint angemessen.

Damit wurde gezeigt, daß die in Kapitel 3 abstrakt definierten Ansprüche die in der Praxis auftretenden Aufgabenstellungen abdecken, natürlich mit der Einschränkung, daß nur 3 Zuordnungsprobleme explizit untersucht wurden und auch keine ausführliche Evaluierung stattfinden konnte.

Im folgenden Kapitel 5 wurde ein Konzept entworfen, Oberflächen zu generieren, welche die geforderten Ansprüche erfüllt. Dabei wurde darauf geachtet, moderne Erkenntnisse auf dem Gebiet der intelligenten Benutzerschnittstellen umzusetzen, wie etwa die Forderung nach der Trennung von Anwendung und Benutzerschnittstelle oder einer möglichst abstrakten Wissensspezifizierung – die Oberfläche versucht, möglichst viele Detailfragen automatisch zu beantworten, der Benutzer muß nur sich, wenn überhaupt, nur um wesentliche, high-level Aspekte kümmern.

Mit einer Implementierung des vorgestellten Konzepts wurde dann eine Oberfläche für die Dienstplanung der Pflegepersonals der anästhesiologischen Klinik der Universität Würzburg generiert. Diese Oberfläche wurde dann in der Praxis erprobt und evaluiert.

7.2 Bewertung

Das in dieser Arbeit erstellte Konzept zur Generierung von interaktiven Oberflächen zur Lösung von Zuordnungsproblemen wurde erfolgreich zur Erstellung einer Oberfläche eingesetzt. Dabei war der Arbeitsaufwand zur eigentlichen Generierung der Oberfläche, also der Bereich, der nicht durch die automatische Generierung abgedeckt wurde, nur sehr gering, konkret im Bereich von wenigen Stunden¹⁸.

Die erzeugte Oberfläche weist eine erstaunlich hohe Qualität auf, was durch die Tatsache belegt wird, daß sie bis auf einige kleinere Mängel sehr gut in der Praxis eingesetzt werden kann. So weit dies nachprüfbar ist, wurden die Ziele dieser Arbeit mehr als erreicht. Zu einem fundierten Urteil wäre jedoch eine Generierung von Oberflächen von verschiedenen Zuordnungsproblemen und eine Evaluation sowohl des Arbeitsaufwands für diese Erzeugung als auch der erzeugten Oberflächen notwendig. Eine solche Untersuchung ist allerdings mit großen Zeitaufwand verbunden und im Rahmen dieser Diplomarbeit nicht praktikabel. Ohnehin mußten bei manchen Themen Einschränkungen gemacht werden; so wäre eine ausführlichere Beschreibung des Konzepts zur Generierung von Oberflächen wäre wünschenswert gewesen. Dies ließ sich aber im gegebenen Rahmen nicht realisieren, schließlich mußten noch Probleme gelöst werden, die keinen Eingang in die vorliegende schriftlichen Fassung dieser Arbeit finden konnten, zum Beispiel die Implementation der Dienstplanung für das Pflegepersonal eines Krankenhauses als Zuordnungsproblem im Rahmen der Zuordnungsshell Coke, eine Aufgabe, die sich als sehr zeitaufwendig erwies.

7.3 Ausblick

Der nächste Schritt wird eine vollständige Implementation des vorgestellten Konzepts sein, um dann Oberflächen für weitere Zuordnungsprobleme zu generieren, zum einen für die anderen der im Rahmen dieser Arbeit beschriebenen Oberflächen, zum anderen für weitere, noch nicht untersuchte Zuordnungsprobleme.

In Hinblick auf die bereits generierte Oberfläche für die Dienstplanung des Pflegepersonals steht die Umsetzung der in Kapitel 6 als Ergebnis der Evaluation formulierten Verbesserungen an. Nach der Integration dieser Verbesserungen in die Oberfläche sollte der Einsatz der Oberfläche im Klinikalltag langfristig untersucht werden. Auch die in dieser Arbeit nur am Rande behandelten Aspekte der automatischen Planung und Optimierung von Dienstplänen sollte untersucht werden, da er den Nutzen der Oberfläche noch weiter erhöhen könnte.

Zu prüfen wäre auch, ob sich das vorgestellte Konzept zur Generierung von Oberflächen auf verwandte Probleme erweitern läßt oder auf weitere Anforderungen, die bei interaktiven Lösung von Zuordnungsproblemen auftreten, wie die Verwaltung der Angebots- und Nachfragemengen.

¹⁸ Diese Aussage bezieht sich auf die Schwerpunkte dieser Arbeit, also die Darstellung der Zuordnung und der Visualisierung von Randbedingungen, nicht auf den zusätzlichen Aufwand zur Verwaltung der Objekte der Zuordnung, also etwa der Spezifikation der Schichttypen.

Anhang A: Abbildungsverzeichnis

Abbildung 1: Darstellung einer Mehrfachzuordnung, wobei die Abbildung von N auf A_1 eine 1-n Zuordnung und die Abbildung von N auf A_p eine 1-1 Zuordnung ist.....	11
Abbildung 2: Überblick über die Typen von Randbedingungen.....	18
Abbildung 3: Liste von Zuteilungen.....	26
Abbildung 4: Komprimierte Listendarstellung	26
Abbildung 5: Prinzip der Funktionsdarstellung (nach [Busch 97]).....	27
Abbildung 6: Funktionsdarstellung mit Hilfslinien (nach [Busch 97]).....	27
Abbildung 7: Darstellung als Tabelle (nach [Busch 97]).....	28
Abbildung 8: Tabelle mit injektiver Gruppierung	28
Abbildung 9: Tabelle mit nicht-injektiver Gruppierung	29
Abbildung 10: Darstellung einer Mehrfachzuordnung	30
Abbildung 11: Kombination von Mehrfach- und Gruppendarstellung	30
Abbildung 12: Symbole als Kodierungsform	33
Abbildung 13: Icons als Kodierungsform.....	34
Abbildung 14: Farbe und Schraffierungen als Kodierungsform	35
Abbildung 15: Schriftform als Kodierungsform	35
Abbildung 16: Linienart als Kodierungsform	36
Abbildung 17: Darstellung von Attributen im Rahmen der Zuordnungsdarstellung.....	40
Abbildung 18: Visualisierung von Attributen bei der Gruppendarstellung von Zuordnungen	40
Abbildung 19: Mögliche Stellen zur internen Visualisierung von Regelverstößen	41
Abbildung 20: Mögliche Visualisierungen des Typs Vermeidung.....	44
Abbildung 21: Mögliche Visualisierungen des Typs Durchsetzung	44
Abbildung 22: Mögliche Visualisierungen des Typs komplexe Vermeidung	45
Abbildung 23: Mögliche Visualisierungen des Typs Arithmetischer Ziel- oder Grenzwert	46
Abbildung 24: Mögliche Visualisierung des Typs Reihenfolge	47
Abbildung 25: Ausführliche Visualisierung einer ausgewählten Restriktionsverletzung (mit Schraffierungen als Ersatz für Blinken)	49
Abbildung 26: Externe Standarddarstellung.....	50
Abbildung 27: Dialog zur Spezifikation der Darstellung von Regelverstößen.....	54
Abbildung 28: Zwei simultane Verstöße gegen dieselbe Regel.....	55
Abbildung 29: Selektionsorientierte Präsentation der Interaktionsmöglichkeiten	59
Abbildung 30: Visualisierung der Existenz von Randbedingungen	62
Abbildung 31: Visualisierung von Veränderungen der Zuordnung	63
Abbildung 32: Traditionelle Darstellung der Schulstundenplanung durch eine Plantafel (nach [Busch 97]).....	80
Abbildung 33: Gesamtkonzept zur Generierung von Oberflächen.....	96
Abbildung 34: Beantwortung von Anfragen gemäß der Wissenshierarchie	99
Abbildung 35: Zuteilungsstruktur der Dienstplanung für das Pflegepersonal eines Krankenhauses.....	101
Abbildung 36: Ausschnitt aus Klassenhierarchie der Visualisierungsklassen.....	103
Abbildung 37: Sektionen einer Tabelle	104
Abbildung 38: Gesamtüberblick über die wichtigsten Komponenten der Oberfläche ...	114
Abbildung 39: Nachrichtenfenster.....	115
Abbildung 40: Hauptmenü der Oberfläche.....	115
Abbildung 41: Optionen zur Verwaltung von Dienstplänen	116
Abbildung 42: Optionen zur Festlegung der Eckdaten des Plans.....	116

Abbildung 43: Festlegung des Planungszeitraums	117
Abbildung 44: Festlegung der Qualifikationsstufen.....	118
Abbildung 45: Festlegung der Schichttypen.....	119
Abbildung 46: Festlegung der Standard-Personalanforderungen.....	120
Abbildung 47: Optionen zur Erstellung von Dienstplanen.....	120
Abbildung 48: Darstellung eines leeren Dienstplans	122
Abbildung 49: Darstellung eines fertigen Dienstplans	124
Abbildung 50: Externe Visualisierung einer Restriktionsverletzung.....	126

Anhang B: Quellen

- [Apple 86] Apple Computer, Inc.: HUMAN INTERFACE GUIDELINES: THE APPLE DESKTOP INTERFACE, Apple Programmer's and Developer's Association, Renton (WA, USA), 1986
- [Busch 97] **Busch C.:** SPEZIFIKATION EINES KOOPERATIVEN WERKZEUGES ZUR LÖSUNG KOMPLEXER ZUORDNUNGSPROBLEME, Diplomarbeit, Universität Würzburg, Lehrstuhl für künstliche Intelligenz und angewandte Informatik, 1997
- [Bronstein 97] **Bronstein I.:** TASCHENBUCH DER MATHEMATIK, 2. Auflage, VERLAG HARRI DEUTSCH, Thun und Frankfurt am Main, 1997
- [gp-UNTIS 96] GP-UNTIS: DAS STUNDENPLANPROGRAMM FÜR DEN COMPUTER, Handbuch zur Version 12.1, Gruber & Petters Software, Stockau, 1996
- [Herczeg 94] **Herczeg M.:** SOFTWARE ERGONOMIE: GRUNDLAGEN DER MENSCH-COMPUTER-KOMMUNIKATION, Addison-Wesley, 1994
- [Jacobson 92] **Jacobson I.:** OBJECT-ORIENTED SOFTWARE ENGINEERING, Addison-Wesley, 1992
- [Hestermann 97] **Hestermann C.:** WIZARD: A SYSTEM FOR PREDICATIVE; REACTIVE AND INTERACTIVE MULTI-RESSOURCE SCHEDULING, Forschungsbericht, Universität Würzburg, Lehrstuhl für künstliche Intelligenz und angewandte Informatik, 1997
- [Kurbel 93] **Kurbel K.:** PRODUKTIONSPLANUNG UND -STEUERUNG: METHODISCHE GRUNDLAGEN VON PPS-SYSTEMEN UND ERWEITERUNGEN, Handbuch der Informatik, Band 13.2; Oldenburg Verlag, München, 1993

- [Maybury & Wahlster 98] **Maybury M., Wahlster W.:** READINGS IN INTELLIGENT USER INTERFACES, Morgan Kaufmann Publishers, San Francisco (CA, USA), 1998
- [Möller 98] **Möller R.:** USER INTERFACE MANAGEMENT SYSTEMS: THE CLIM PERSPECTIVE, <http://kogs-www.informatik.uni-hamburg.de/~moeller/uims-clim/clim-intro.html>, 1998
- [Myers 98] **Myers B. A.:** USER INTERFACE TOOLS: INTRODUCTION AND SURVEY, S. 15-23, IEEE Software, January 1989,
- [Poeck 95] **Poeck K.:** KONFIGURIERBARE PROBLEMLÖSUNGSMETHODEN AM BEISPIEL DER PROBLEMKLASSEN ZUORDNUNG UND DIAGNOSTIK, Dissertation, Universität Würzburg, Infix (Dissertationen zur künstlichen Intelligenz, Bd. 86), Sankt Augustin, 1995
- [Puppe 90] **Puppe F.:** PROBLEMLÖSUNGSMETHODEN IN EXPERTENSYSTEMEN, Springer-Verlag, Berlin, Heidelberg, 1990
- [Shneiderman 98] **Shneiderman B.:** DESIGNING THE USER INTERFACE, 3^d Edition, Addison-Wesley, 1998
- [Sommerville 96] **Sommerville I.:** SOFTWARE ENGINEERING, 5th Edition, Addison-Wesley, 1996
- [Stary 96] **Stary C.:** INTERAKTIVE SYSTEM, 2. Auflage, Vieweg, Braunschweig 1996
- [Vanderdonckt 95] **Vanderdonckt J.:** AUTOMATIC GENERATION OF A USER INTERFACE FOR HIGHLY INTERACTIVE BUSINESS-ORIENTED APPLICATIONS, in Maybury M. and Wahlster W. (Hrsg.): Readings in Intelligent User Interfaces, S.516-518; Morgan Kaufmann Publishers, San Francisco (CA, USA), 1998
- [Wiecha et al. 91] **Wiecha C, Bennet W., Boies S., Gould J. and Greene S.:** IST: A TOOL FOR RAPIDLY DEVELOPING INTERACTIVE APPLICATIONS, in Maybury M. and Wahlster W. (Hrsg.): Readings in Intelligent User Interfaces, S.481-497; Morgan Kaufmann Publishers, San Francisco (CA, USA), 1998

Anhang C: Source-Code

Der Source-Code befindet sich auf der beiliegenden Diskette, bzw. CD.

Anhang D: Danksagung

Ich danke allen, die mir bei der Erstellung dieser Arbeit geholfen habe, insbesondere meinem Betreuer Ciske.