

528 | Dezember 1992

## SCHRIFTENREIHE SCHIFFBAU

Harald Jensen

**Überprüfung der Anwendbarkeit und  
Einhaltung technischer Regelwerke  
sowie Dimensionierung an Hand von  
Regelwerken mit Hilfe von Rechnern**

**TUHH**

*Technische Universität Hamburg-Harburg*

**Überprüfung der Anwendbarkeit und Einhaltung technischer Regelwerke sowie Dimensionierung an Hand von Regelwerken mit Hilfe von Rechnern**

Harald Jensen, Hamburg, Technische Universität Hamburg-Harburg, 1992

ISBN: 3-89220-528-0

© Technische Universität Hamburg-Harburg  
Schriftenreihe Schiffbau  
Schwarzenbergstraße 95c  
D-21073 Hamburg

<http://www.tuhh.de/vss>

Überprüfung der Anwendbarkeit  
und Einhaltung technischer Regelwerke  
sowie Dimensionierung an Hand von  
Regelwerken  
mit Hilfe von Rechnern

Dissertation  
zur Erlangung des Grades  
Doktor-Ingenieur  
der Universität Hamburg

vorgelegt von  
Harald Jensen  
aus Hamburg

Hamburg  
September 1992

Tag der Disputation: 17.12.92

Gutachter: Prof. Dr.-Ing. H. Keil  
Prof. Dr.-Ing. H. Söding

INSTITUT FÜR SCHIFFBAU DER UNIVERSITÄT HAMBURG

Bericht Nr. 528

Überprüfung der Anwendbarkeit und  
Einhaltung technischer Regelwerke  
sowie Dimensionierung an Hand von  
Regelwerken mit Hilfe von Rechnern

Harald Jensen

Dezember 1992

# Inhaltsverzeichnis

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Einleitung</b>   | <b>5</b>  |
| 1.1      | Stand der Technik, <i>Inhalt</i> und <i>Relevanz</i> von Regelwerken . . . . .    | 6         |
| 1.2      | Stand der Technik, <i>Prüfen</i> und <i>Dimensionieren</i> von Objekten . . . . . | 7         |
| 1.3      | Die Aufgabenstellung . . . . .  | 8         |
| <b>2</b> | <b>Formalisierung von Regelwerken</b>   | <b>10</b> |
| 2.1      | Elemente und Berechnung des Ergebnisses eines Regelwerkes . . . . .               | 10        |
| 2.1.1    | Beispiel . . . . .  | 13        |
| 2.2      | Berechnung der Ergebnisse logischer Aussagen . . . . .                            | 14        |
| 2.2.1    | Bestimmung der Wahrscheinlichkeit einer Aussage . . . . .                         | 15        |
| 2.2.2    | Verknüpfungen von Aussagen . . . . .  | 17        |
| 2.3      | Datentypen und Datenmodelle . . . . .   | 18        |
| 2.4      | Zusammenfassung . . . . .   | 19        |
| <b>3</b> | <b>Lösungsansatz zur Interpretation von Regelwerken</b>                           | <b>21</b> |
| 3.1      | Aufbereiten der Regelwerke . . . . .  | 21        |
| 3.1.1    | Regelsyntax . . . . .   | 22        |
| 3.2      | <i>Prüfen</i> von Objekten . . . . .  | 26        |
| 3.2.1    | Ausgabemodi . . . . .   | 26        |
| 3.3      | <i>Relevanz</i> von Regeln . . . . .  | 28        |
| 3.3.1    | Ausgabemodi . . . . .   | 28        |
| 3.4      | <i>Inhalt</i> von Regelwerken . . . . .   | 29        |
| 3.5      | <i>Dimensionieren</i> von Daten an Hand von Regeln . . . . .                      | 29        |
| 3.5.1    | Verwendete Optimierungsmethoden . . . . .   | 31        |
| 3.6      | Unbekannte und unsichere Daten . . . . .  | 32        |
| 3.7      | Berechnungen in Regelwerken . . . . .   | 33        |

|          |  |           |
|----------|--|-----------|
| <b>4</b> | <b>Softwareengineering und Realisierung</b>            | <b>34</b> |
| 4.1      | Systemanalyse und Realisierungskonzept . . . . .       | 34        |
| 4.2      | Compiler, Aufbereiten des Regelcodes . . . . .         | 38        |
| 4.2.1    | Analyserroutinen . . . . .                             | 39        |
| 4.2.2    | Regel-Datenbank . . . . .                              | 39        |
| 4.3      | Regelmodul, Anwendung von Regeln auf Objekte . . . . . | 40        |
| 4.3.1    | Verwaltung der Objektdaten . . . . .                   | 41        |
| 4.4      | Postprozessor, Darstellung der Ergebnisse . . . . .    | 42        |
| 4.5      | Benutzerschnittstelle RSHELL . . . . .                 | 42        |
| 4.6      | Programmier-Sprache, Programmier-Umgebung . . . . .    | 43        |
| <b>5</b> | <b>Beispielanwendungen und Tests</b>                   | <b>46</b> |
| 5.1      | Beispiel-Vorschrift . . . . .                          | 46        |
| 5.1.1    | Übersetzen in den Regelcode . . . . .                  | 46        |
| 5.1.2    | Ergebnisse . . . . .                                   | 46        |
| 5.1.3    | Test der Integrationsmethoden . . . . .                | 47        |
| 5.1.4    | Test der Dimensionierungsmöglichkeiten . . . . .       | 49        |
| 5.2      | IMO Resolution A167 . . . . .                          | 51        |
| 5.3      | SOLAS 83 II-2 . . . . .                                | 52        |
| 5.4      | Ressourcenverbrauch . . . . .                          | 52        |
| <b>6</b> | <b>Zusammenfassung</b>                                 | <b>55</b> |
| <b>7</b> | <b>Begriffe, Abkürzungen und Symbole</b>               | <b>57</b> |
| 7.1      | Verzeichnis der Begriffe und Abkürzungen . . . . .     | 57        |
| 7.2      | Symbolverzeichnis . . . . .                            | 58        |
| <b>8</b> | <b>Literatur</b>                                       | <b>59</b> |

# Anhang

|          |  |           |
|----------|--|-----------|
| <b>A</b> | <b>Beschreibung existierender Programme</b>                                    | <b>63</b> |
| A.1      | Stabilität . . . . .   | 63        |
| A.2      | Festigkeit . . . . .   | 64        |
| A.3      | andere Bereiche . . . . .  | 65        |
| <b>B</b> | <b>EBNF – Erweiterte Backus-Naur Form</b>                                      | <b>66</b> |
| <b>C</b> | <b>Beispiele</b>   | <b>68</b> |
| C.1      | Beispiel Vorschrift . . . . .  | 68        |
| C.1.1    | Regeltext . . . . .  | 68        |
| C.1.2    | Regelcode . . . . .  | 69        |
| C.1.3    | Datenmodell . . . . .  | 71        |
| C.1.4    | Daten . . . . .  | 72        |
| C.1.5    | Prüfen . . . . .   | 73        |
| C.1.5.1  | Prüfen, tabellarische Ausgabe, kurz . . . . .                                  | 73        |
| C.1.5.2  | Prüfen, tabellarische Ausgabe, Überblick . . . . .                             | 74        |
| C.1.5.3  | Prüfen, tabellarische Ausgabe, vollständig . . . . .                           | 77        |
| C.1.5.4  | Prüfen einer aufgerufenen Regel, chronologische Ausgabe, vollständig . . . . . | 81        |
| C.1.6    | Ausgabe Dimensionieren, tabellarisch, kurz . . . . .                           | 83        |
| C.1.7    | Ausgabe Inhalt, chronologische Ausgabe . . . . .                               | 84        |
| C.1.8    | Ausgabe Relevanz . . . . .   | 87        |
| C.1.8.1  | Relevanz, tabellarische Ausgabe, Überblick . . . . .                           | 87        |
| C.1.8.2  | Relevanz, tabellarische Ausgabe, vollständig . . . . .                         | 89        |
| C.2      | IMO Resolution A167 . . . . .  | 93        |
| C.2.1    | Regelcode . . . . .  | 93        |

|          |   |            |
|----------|---|------------|
| C.2.2    | Datenmodell . . . . .                                 | 100        |
| C.2.3    | Daten . . . . .                                       | 108        |
| C.2.4    | Prüfen, tabellarische Ausgabe, kurz . . . . .         | 110        |
| C.2.5    | Relevanz, tabellarische Ausgabe, Überblick . . . . .  | 112        |
| C.3      | SOLAS 83 II-2 . . . . .                               | 119        |
| C.3.1    | Regelcode . . . . .                                   | 119        |
| C.3.2    | Datenmodell . . . . .                                 | 137        |
| C.3.3    | Daten . . . . .                                       | 143        |
| C.3.4    | Prüfen, tabellarische Ausgabe, kurz . . . . .         | 151        |
| C.3.5    | Dimensionieren, tabellarische Ausgabe, kurz . . . . . | 160        |
| <b>D</b> | <b>Programmsteuerung</b>                              | <b>161</b> |
| D.1      | Compiler . . . . .                                    | 161        |
| D.2      | Bearbeiten . . . . .                                  | 162        |
| D.3      | Ausgabe . . . . .                                     | 164        |
| D.4      | RSHELL . . . . .                                      | 166        |

# 1 Einleitung

In immer größerem Umfang muß der Ingenieur bei der Entwurfsarbeit und auch bei der Detailkonstruktion die verschiedensten technischen Regelwerke berücksichtigen und einhalten. Der Umfang und die Komplexität dieser Regelwerke wächst ständig.

Mit technischen Regelwerken sind im Bereich des Schiffbaus die internationalen Vorschriften mit Anhängen und Codes der IMO<sup>1</sup>, die Vorschriften der Klassifikationsgesellschaften, die nationalen Vorschriften wie die der deutschen SBG oder der amerikanischen USCG und natürlich die Spezifikationen der Reeder gemeint. Hinzu kommen allgemeine Vorschriften und Normen wie DIN und ISO.

Während der Bearbeitung eines Projektes muß der Ingenieur in den einzelnen Phasen verschiedene Tätigkeiten mit Regelwerken durchführen. Als erstes gilt es festzustellen, welche Regeln eines Regelwerkes auf ein Objekt angewendet werden müssen (*Relevanz*). Danach muß geprüft werden, ob die schon festgelegten Daten und Details des Entwurfs bzw. der Konstruktion die Forderungen der Regelwerke erfüllen (*Prüfen*). Sofern die Forderungen nicht erfüllt werden, muß die Konstruktion bzw. müssen die Daten so festgelegt werden, daß die einzelnen Regeln erfüllt werden (*Dimensionieren*). Daneben muß der Ingenieur sich ganz allgemein über den Inhalt von Vorschriften informieren (*Inhalt*).

Das Arbeiten mit den o.g. Regelwerken erfordert i.A. viel Erfahrung, da die Strukturen einiger Regelwerke sehr komplex sind. Auch sind die Regelwerke durch Ergänzungen einem ständigen Wandel unterworfen. So ist es oft schwierig herauszufinden, welche Version eines Regelwerkes für ein älteres Schiff relevant ist. Eine andere Problematik ist, daß viele Details eines Entwurfs oder einer Konstruktion mehreren Regelwerken genügen müssen (siehe auch [IMO91b]<sup>2</sup>, Seite 3).

Von den eben genannten Aufgaben: *Inhalt*, *Relevanz*, *Prüfen* und *Dimensionieren*, sind die letzten drei am Beginn der Entwurfsphase problematisch, wenn viele Größen und Konstruktionsweisen noch unbekannt bzw. nicht endgültig festgelegt sind.

In immer größerem Ausmaß stehen Rechner zur Verfügung, und die technischen Daten von Entwürfen und Konstruktionen werden mit Hilfe von Rechnern erstellt und in ihnen gespeichert (CAD, CAM, CIM usw.). Konsequenterweise sollten die oben genannten Aufgaben und die dafür erforderlichen Berechnungen auch mit Hilfe von Rechnern durchgeführt werden. Man kann dies als eine weitere der sogenannten CA-Techniken, dem *Computer Aided Regulationhandling*, CAR, einführen.

---

<sup>1</sup>Abkürzungen siehe 7.1

<sup>2</sup>Literaturangaben sind in Kapitel 8 zusammengefaßt

## 1.1 Stand der Technik, *Inhalt* und *Relevanz* von Regelwerken

Um sich über den Inhalt von Regelwerken, allgemein von Texten zu informieren, gibt es durch die Verwendung von Rechnern eine ganze Reihe von Hilfsmitteln. Genannt seien hier Indizierung, Hypertextfunktionalitäten und Volltextsuche. Geeignete Programme können im Bereich der Sprachwissenschaftler oder auch der Jurisprudenz gefunden werden.

Bisher war neben dem Verwalten der sehr großen Datenmengen vor allem das Erfassen der Texte und das Einbinden von Bildern und Diagrammen problematisch. Immer schnellere und günstigere Hardware mit Scannern<sup>3</sup>, OCR-Software und Datenbanksystemen ermöglichen es heute, diese Aufgaben durchzuführen.

ABS bietet als erste Klassifikationsgesellschaft ihre Vorschriften auf dem elektronischen Medium CD-ROM an ([ABS91]). Das mitgelieferte Program ermöglicht Volltextsuche nach mehreren Begriffen sowie über sogenannte *Hotlinks* Verzweigungen zu Verweisen und Graphiken. Es ermöglicht die mit *Inhalt* beschriebene Aufgabe zu lösen.

Seit 1987 gibt es von DNV das System VEGA [DNV91]. Das System deckt sowohl die Vorschriften der IMO als auch die zugehörigen Interpretationen und nationalen Forderungen ab. VEGA ist ein Datenbanksystem, in dem alle Informationen, Texte und Graphiken einzelner Regeln, in Abhängigkeit von Schiffstyp, Ladungsart, Fahrtgebiet, Länge zwischen den Loten, Vermessung und Zuladung gespeichert sind. Es läßt sich damit z.B. eine Liste aller Regeln, die ein bestimmtes Schiff erfüllen muß, erstellen (*Relevanz*).

Innerhalb der IMO wurde ebenfalls ein Datenbanksystem geplant, das helfen soll, alle für ein Schiff geltende Regeln zu finden. Die IMO hat eine Machbarkeitsstudie [IMO91b] durchführen lassen, die zu dem Schluß kommt, daß es sinnvoll und wirtschaftlich ist, so ein Datenbanksystem zu entwickeln und zu pflegen. Anfang 1992 wurde das von DNV zwischenzeitlich für PC's umgeschriebene Programm VEGA als offizielles IMO-Datenbank-System für Vorschriften vorgestellt [DNV92]. Das Programm wird weiter von DNV gepflegt. Die *IMO/Vega* genannte Datenbank wird von der IMO gepflegt. Z.Zt. umfaßt der Datenbestand ca. 150000 DIN A4 Seiten Vorschriften. Das System wird auf einer CD-ROM ausgeliefert.

Außerdem will die IMO dazu übergehen, ihre Veröffentlichungen auch auf elektronischen Medien zu publizieren. Geplant ist für 1992, den IMDG Code und 1993 SOLAS oder MARPOL oder beides auf CD-ROM herauszugeben [IMO91a].

---

<sup>3</sup>siehe 7.1, Verzeichnis der Begriffe und Abkürzungen

## 1.2 Stand der Technik, Prüfen und Dimensionieren von Objekten

Um zu prüfen, ob ein Objekt, z.B. ein Schiff, einem Regelwerk genügt, reicht es nicht aus, den reinen Text der Vorschrift mit dem Rechner verwalten zu können. Der Text muß interpretiert werden und in logische, dem Regelwerk entsprechende Abfolgen von meist mathematisch formulierbaren Fragen oder Vergleichen zerlegt werden. Anschließend müssen diese Fragen an hand der aktuellen Daten des Objekts beantwortet werden. Hierzu muß eine Verbindung zwischen dem Regelwerk und den Daten des Objekts hergestellt werden.

Es existieren eine ganze Reihe von individuellen Lösungen für einzelne, meist eng begrenzte Teilbereiche von Regelwerken. In allen Fällen wurden Programm-Routinen, die die Prüfung und in Einzelfällen auch die Dimensionierung durchführen, in Programme integriert, deren Aufgabe im wesentlichen darin besteht, die Daten, die zur Prüfung benötigt werden, zu berechnen. Diese Programme stammen durchweg aus den Bereichen Stabilität und Festigkeit. Eine Auswahl ist in Tabelle 1.1 aufgeführt und in Anhang A beschrieben.

Daneben gibt es eine Fülle von kleinen und großen Programmen und Programm-Systemen, die Zahlenwerte für die Prüfung bereitstellen.

Eine andere Aufgabe bearbeiten Programme wie CHEMSHIP II und S2PROD16, die unter Verwendung einer Datenbank die Liste der zur Beförderung zugelassenen Produkte entsprechend verschiedener IMO-Codes für ein Schiff erstellen.

Im Auftrag des BVM ist im Hause des GL eine Studie über die Möglichkeiten des Einsatzes von Expertensystemen zur Interpretation von Regelwerken erstellt worden [Sans90]. Nach einigen grundsätzlichen Überlegungen, ob so etwas überhaupt sinnvoll ist und welche Werkzeug hierfür verwendet werden könnten, wurde mit einer Expertensystem-Shell ein Prototyp für ein Expertensystem 'Brandschutz' erstellt. Implementiert wurden ca. 9% der Regeln des Kapitels II-2, Feuersicherheit, aus Safety of Life at Sea (SOLAS).

Der Prototyp kann dem Benutzer aufzeigen, welche Regeln für ein gegebenes Objekt nicht erfüllt werden. Zusätzlich werden Hinweise auf die zu berücksichtigenden Abschnitte der SchSV gegeben. In der Zusammenfassung wird das System als intelligentes Inhaltsverzeichnis charakterisiert.

Die Studie kommt zu dem Ergebnis, daß Computerunterstützung zur Interpretation von Regelwerken und damit Entlastungen einzelner menschlicher Experten möglich sind, daß sie benötigt werden, und empfiehlt, in dieser Richtung weiter zu arbeiten. Die Autoren weisen allerdings auch darauf hin, daß die Erstellung solcher Expertensysteme sehr zeit- und kostenintensiv sind. Speziell die Vorbereitung und Strukturierung der Wissensbasis erfordere einen erheblichen Aufwand.

An der Dimensionierung von Stahlstrukturen an Hand von Bauvorschriften wird an der Technischen Universität Hamburg-Harburg gearbeitet [Top92]. Bisher wurde die

| Programm    | Organisation | Vorschrift   |
|-------------|--------------|--|
| SFOLDS      | BMT          | fast alle IMO Vorschriften zu Intakt- und Leckstabilität<br>Längsfestigkeit                |
| AutoHYDRO   | Coastdesign  | fast alle IMO Vorschriften zu Intakt- und Leckstabilität<br>Längsfestigkeit                |
| AutoCLASS   | Coastdesign  | Klassezeichnungen und Dimensionierung einzelner Bauteile nach DNV Vorschriften             |
| Pilot       | DNV          | IMO A167 (Intaktstabilität)<br>Längsfestigkeit<br>Örtliche Festigkeit nach DNV-Rules       |
| GLRULES     | GL           | der ganze Bereich der GL-Vorschriften für Stahlschiffbau                                   |
| S2PROD16    | GL           | Erstellen der Produktenliste nach IMO Chemical-Code  |
| SEASAFE 1   | LR           | fast alle IMO Vorschriften zu Intakt- und Leckstabilität                                   |
| LR.PASS     | LR           | der ganze Bereich der LR-Rules für Stahlschiffbau  |
| CHEMSHIP II | LR           | Erstellen der Produktenliste nach IMO Gas- und Chemical-Code                               |
| NAPA        | NAPA OY      | alle IMO und viele nationale Vorschriften zu Intakt- und Leckstabilität<br>Längsfestigkeit |

Tabelle 1.1: existierende Programme zum Überprüfen

Bereitstellung der Informationen für die Dimensionierung mit Hilfe eines Produktmodells behandelt.

### 1.3 Die Aufgabenstellung

Die in 1.1 und 1.2 genannten Programme sind Lösungen für ganz spezielle Teilaspekte oder Teilbereiche der in 1 definierten Aufgaben. Ein allgemeiner Ansatz ist mir, auch nach Anfrage bei verschiedenen Klassen und Behörden einiger großer Schifffahrtsnationen sowie bei einigen Softwarehäusern, nicht bekannt geworden.

Der überwiegende Teil der in der Einleitung genannten Regelwerke wird von den aufgeführten Programmen nicht abgedeckt. Unsichere und unbekannt Daten in der Entwurfsphase werden von keinem dieser Programme berücksichtigt.

Mit dem Schwerpunkt auf *Prüfen* wird nach einem allgemeinen Ansatz gesucht, mit dem die in 1 definierten vier Aufgaben (*Inhalt, Relevanz, Prüfen, Dimensionieren*) für verschiedenste technische Regelwerke bearbeitet werden können. Als wesentliche Forderungen soll

1. die Umsetzung der geschriebenen Regelwerke in die Wissensbasis so einfach wie möglich gemacht werden,
2. die Ergebnisse so dargestellt werden, daß die aktuelle Fragestellung schnell beantwortet werden kann, z.B. ob ein Objekt die Forderungen eines Regelwerkes erfüllt.

Im folgenden Kapitel werden die benötigten Grundlagen dargestellt und anschließend ein Lösungsansatz und die Implementierung beschrieben. Der Lösungsansatz wird dann an hand einer einfachen Beispiel-Vorschrift und Teilen internationaler Regelwerke diskutiert.

## 2 Formalisierung von Regelwerken

Dieses Kapitel behandelt grundsätzliche Definitionen und Methoden, mit denen man die in der Einleitung gestellten Aufgaben lösen kann.

Im Abschnitt 2.1 werden die Elemente definiert, in die man die in der Einleitung genannten Regelwerke zerlegen kann und wie deren Ergebnisse definiert und berechnet werden können. Der Abschnitt 2.2 definiert eine realwertige Logik, die es ermöglicht auch mit unsicheren und unbekanntem Daten zu arbeiten. Der Abschnitt 2.3 definiert schließlich die Behandlung verschiedener Datentypen.

### 2.1 Elemente und Berechnung des Ergebnisses eines Regelwerkes

Das einfachste denkbare Regelwerk besteht aus einer einzelnen Regel, die erfüllt sein muß. Diese besteht im einfachsten Fall nur aus einer Bedingung. Im technischen Bereich ist diese Bedingung häufig eine Forderung von Kleinst- bzw. Größtwerten für eine bestimmte Größe. Die Bedingung ist also ein Vergleich des Ist-Wertes des aktuellen Objektes und des Soll-Wertes der Vorschrift. Allgemeiner: eine Bedingung  $B(O)$  ist ein logischer Ausdruck, der wahr ist, wenn die Bedingung für das aktuelle Objekt  $O$  erfüllt ist. In diesem einfachen Fall ist, wenn die Bedingung erfüllt ist, d.h. der logische Ausdruck wahr ist, auch die Regel  $R(O)$  für das Objekt erfüllt.

$$R(O) = B(O)^4$$

Man kann so auch eine Regel als logische Aussage betrachten, die wahr ist, wenn die Regel erfüllt ist.

Häufig gilt eine Regel nicht für alle denkbaren Objekte, sondern für eine bestimmte Gruppe. Die Zugehörigkeit des Objekts zu dieser Gruppe ist die Voraussetzung  $V(O)$ , die auch wieder als logische Aussage verstanden werden kann. Man kann jetzt zusätzlich folgende Aussage machen: Eine Regel ist erfüllt, wenn die Voraussetzung nicht erfüllt ist. Mit anderen Worten, eine Regel ist erfüllt, wenn sie nicht anzuwenden ist. Damit ergibt sich das Erfülltsein der Regel zu:

$$R(O) = !V(O) \parallel B(O)^5$$

Dies kann wie folgt umgeformt werden:

$$R(O) = (V(O) \& B(O)) \parallel !V(O) = K(O) \parallel V_d(O)$$

<sup>4</sup>Siehe auch Kapitel 7.2, Symbolverzeichnis

<sup>5</sup>Die Verwendung von unterschiedlichen Oder-Verknüpfungen für stochastisch unabhängige und sich ausschliessende Aussagen geschieht im Hinblick auf die in Kapitel 2.2 definierte Logik.

mit dem Kriterium

$$K(O) = V(O) \& B(O)$$

und mit der Ansonst-Voraussetzung  $V_d$ :

$$V_d(O) = !V(O)$$

Oft werden unterschiedliche Bedingungen  $B_1(O), B_2(O), \dots, B_n(O)$ , die für verschiedene Voraussetzungen  $V_1(O), V_2(O), \dots, V_n(O)$  gelten sollen, in einer Regel zusammengefaßt. Das heißt, eine Regel besteht aus einer Liste  $L(O)$  von Kriterien  $K_1(O), K_2(O), \dots, K_n(O)$ . Wenn man jetzt annimmt, daß ein Objekt nur einer der in den Voraussetzungen definierten Gruppen angehören kann, und so sind Vorschriften i.a. aufgebaut, heißt das, daß sich die Voraussetzungen gegenseitig ausschließen. Daraus ergibt sich das Erfülltsein einer Regel zu:

$$R(O) = L(O) = K_1(O) \parallel K_2(O) \parallel \dots \parallel K_n(O) \parallel V_d(O)$$

mit

$$V_d(O) = !(V_1(O) \parallel V_2(O) \parallel \dots \parallel V_n(O))$$

Eine Regel, die keine Kriterien enthält, wird als erfüllt betrachtet. Um diesen Fall einzuschließen, wird  $V_d$  endgültig wie folgt definiert:

$$V_d(O) = !(FALSCH \parallel V_1(O) \parallel V_2(O) \parallel \dots \parallel V_n(O))$$

In manchen Regeln soll für ein Objekt nachgewiesen werden, daß eine Liste von Kriterien für alle von mehreren Teilbereichen des gleichen Typs eines Objekts, den Teilobjekten  $T_1, T_2, \dots, T_n$ , erfüllt ist, bzw. es soll nachgewiesen werden, daß die Regel von mindestens einem der Teilobjekte erfüllt wird. Das Erfülltsein der Regel läßt sich dann wie folgt berechnen :

$$R(O) = WAHR \& L(T_1) \& L(T_2) \& \dots \& L(T_n)$$

wenn die Regel für alle Teilobjekte erfüllt sein soll bzw. :

$$R(O) = FALSCH | L(T_1) | L(T_2) | \dots | L(T_n)$$

wenn mindestens ein Teilobjekt die Regel erfüllen soll. Die Ergebnisse  $L(T_i)$  werden mit ODER für voneinander unabhängige Aussagen verknüpft.

Diese Konstruktionen werden im folgenden als *für alle – Schleife* bzw. *einervon – Schleife* bezeichnet. Die Konstanten *WAHR* und *FALSCH* werden eingeführt für den Fall, daß die Anzahl der Teilobjekte 0 ist.

Wenn eine *einervon – Schleife* vorliegt, ist es nötig, die Ansonst-Voraussetzung  $V_d$  der Kriterienliste immer gleich *FALSCH* zu setzen, da sonst Teilobjekte, für die keine Voraussetzung wahr ist, die Regel erfüllen.

Voraussetzungen, Bedingungen, Kriterien und Listen werden hier ebenso wie Regeln als logische Aussagen behandelt. Eine Bedingung oder Voraussetzung kann damit nicht nur aus Vergleichen, sondern z.B. auch wieder aus einer Regel oder der logischen Verknüpfung mehrerer Regeln und Vergleiche bestehen.

Die in der Einleitung aufgeführten Regelwerke, die hier bearbeitet werden sollen, sind alle hierarchisch aufgebaut. So betrachtet kann ein Regelwerk als Regel behandelt werden, deren Voraussetzungen und Bedingungen Regeln sind, die aus den untergeordneten Teilen der Vorschrift bestehen. Ob ein Objekt einem Regelwerk entspricht, ergibt sich dann als das Erfülltsein der Regel, die dem Regelwerk entspricht.

Teile von Regelwerken lassen sich prüfen, indem mit der entsprechenden untergeordneten Regel gearbeitet wird.

Bei der Frage, ob ein Regelwerk auf ein bestimmtes Objekt anzuwenden ist (*Relevanz*), ergibt sich das Ergebnis ähnlich wie bei *Prüfen*. Bei der Berechnung der Ergebnisse der Bedingungen wird immer der Wert WAHR gesetzt, d.h. es werden nur die Voraussetzungen berücksichtigt. Die Kriterien einer Liste werden mit ODER für voneinander abhängige Werte verknüpft. Bei Schleifen wird das Gesamtergebnis aus den Teilergebnissen mit ODER für stochastisch unabhängige Aussagen bestimmt.

$$R(O) = L(T_1) | L(T_2) | \dots | L(T_n)$$

$$L(T_i) = V_1(T_i) || V_2(T_i) || \dots || V_n(T_i)$$

Um sich alle Regeln eines Regelwerkes ausgeben zu lassen (*Inhalt*), werden zusätzlich die Ergebnisse aller Voraussetzungen auf WAHR gesetzt. Mit diesem Ansatz können die Aufgaben *Inhalt*, *Relevanz* und *Prüfen* nach der gleichen schematischen Vorgehensweise bearbeitet werden.

Das beschriebene System mit Regeln, Kriterien usw. erscheint evtl. unübersichtlich oder sogar überflüssig. Im Grunde benötigt man nur Vergleiche und die logischen Verknüpfungen. Ziel dieses Vorgehens ist jedoch, die Elemente, die in einer gedruckten Vorschrift vorkommen, direkt zu modellieren, so daß sich das Umsetzen der Vorschrift auf den Rechner einfach und nahe an dem Originaltext durchführen läßt.

Ein Element wird deshalb noch hinzugefügt, die zweidimensionale Tabelle  $E(O)$ , als spezielle Form einer Liste von Kriterien. Die Tabelle besteht aus einem zweidimensionalen Feld von Bedingungen. Die Voraussetzungen ergeben sich aus den mit UND verknüpften Vergleichen, ob die beiden aus den Daten des Objekts bestimmten Indizes mit den Indizes der Bedingung übereinstimmen. Das Ergebnis einer Tabelle läßt sich dann wie bei einer Liste von Kriterien bestimmen. Nicht definierte Bedingungen erhalten den Wert WAHR.

Neben dem Vergleich wird als grundlegende logische Aussage noch der *Bereich* definiert. Bei der Berechnung des Ergebnisses eines Bereichs wird die stochastische Abhängigkeit der darin enthaltenen beiden Vergleiche berücksichtigt.

In manchen Vorschriften gibt es Regeln mit hinweisendem Charakter, allgemeinen Voraussetzungen oder Fragestellungen, so daß das Erfülltsein nicht anhand technischer Daten bestimmt werden kann. Ob eine solche Regel erfüllt ist, muß bei der beschriebenen Methodik nach wie vor der Benutzer entscheiden und das Ergebnis direkt angeben.

### 2.1.1 Beispiel

Als einfaches Beispiel sei die folgende, fiktive Stabilitätsvorschrift verwendet:

§1 : Um die Sicherheit von Schiffen zu gewährleisten müssen die in den Paragraphen 2 und 3 geforderten Grenzwerte eingehalten werden.

§2 : Für alle Ladefälle außer 'Leeres Schiff' muss gelten,

```
wenn Lpp >= 100m    a >= 0.2 fuer B*Lpp >= 2000
                   a >= 0.4 fuer B*Lpp <= 1000
                   fuer Zwischenwerte ist linear zu
                   interpolieren
```

```
mit a = GM * Umfang **2 ,
        Umfang im Bogenmass
```

```
andernfalls        GM >= 0.5m und Umfang >= 50 Grad
                   oder GM * Umfang >= 30 m*Grad
```

§3 : Für den Ladefall 'Leeres Schiff' muss gelten

```
GM >= 0.2m und Umfang >= 20 Grad
oder GM * Umfang >= 5 m*Grad
```

Das Ergebnis der Vorschrift entspricht dem Ergebnis der Regel §1, wenn von §1 aus in die Regeln §2 und §3 verzweigt wird.

Die Regel §2 besteht aus einer *fuer alle* Schleife über alle Ladefälle mit der Voraussetzung, daß der Ladefall nicht vom Typ *leeres schiff* ist.

Als Bedingung kann man eine Regel aufrufen, die die eigentlichen Grenzwerte enthält. Dies bietet die Möglichkeit, die Grenzwertabfragen auch auf einen einzelnen Ladefall anwenden zu können. Diese Regel besteht aus zwei Kriterien mit den Voraussetzungen  $Lpp < 100m$  und der Ansonstvoraussetzung. Die Bedingung des ersten Kriteriums gliedert sich wieder in eine Liste von Voraussetzungen und Bedingungen. Lediglich die Bedingung für  $1000 \leq B \cdot Lpp \leq 2000$  eine Formel für die lineare Interpolation zu verwenden, erfordert eine Bearbeitung, die über ein rein mechanisches Umsetzen hinausgeht.

Die Regel §3 ist ähnlich der Regel §2 aufgebaut.

## 2.2 Berechnung der Ergebnisse logischer Aussagen

Im folgenden wird für logische Aussagen das Ergebnis, die Berechnung des Ergebnisses sowie die Verknüpfung mehrerer Aussagen definiert.

Vorschriften sollten schon im frühen Entwurfsstadium berücksichtigt werden. Da zu diesem Zeitpunkt Daten z.T. noch unsicher oder noch gar nicht bekannt sind, sollen hier sowohl die Behandlung unbekannter und unsicherer Daten als auch die daraus resultierenden unsicheren Aussagen berücksichtigt werden. Das heißt Daten bzw. Werte müssen als :

- exakt bestimmte Werte
- unsichere, geschätzte Werte
- unbekannte Werte

gekennzeichnet werden können.

Für Aussagen ergeben sich damit als mögliche Ergebnisse :

- wahr
- nicht sicher wahr und nicht sicher falsch ( unsichere, geschätzte oder unbekannte Werte )
- falsch

Um diese Ergebnisse beschreiben zu können, wird eine real-wertige Logik eingeführt. Das Ergebnis  $P(A)$  einer Aussage  $A$  wird als die Wahrscheinlichkeit, daß die Aussage wahr ist, definiert. Dies entspricht den Ansätzen von Keynes [Krapp73] bzw. Carnap [Carnap59], die Wahrscheinlichkeit als Bestätigungsgrad definieren. Dabei hat das Ergebnis  $P(A)$  einer Aussage  $A$  folgende Bedeutung:

$$\begin{aligned} P(A) = 0 & & : \text{ falsch.} \\ 0 < P(A) < 1.0 & & : \text{ mit der Wahrscheinlichkeit } P(A) \text{ wahr} \\ P(A) = 1.0 & & : \text{ wahr} \end{aligned}$$

Für Daten gibt es zwei Möglichkeiten 'Unsicherheit' zu markieren:

1. dem Wert wird eine Wahrscheinlichkeit  $W$  zugeordnet, die angibt, ob der Wert zutrifft:
  - $W = 0$  : unbekannter Wert
  - $0 < W < 1.0$  : unsicherer, geschätzter Wert
  - $W = 1.0$  : exakt bekannter Wert

2. dem Wert wird ein Unsicherheitsintervall zugeordnet, in dem der exakte Wert liegt. Die Grenzen des Intervalls können bei unbekanntem Wert sehr große Zahlen, z.B.  $-10^{20}$  und  $10^{20}$ , sein. Bei exakt bekannten Werten wird beiden Grenzen der exakte Wert zugewiesen.

Da diese Arbeit für den Bereich der Technik entstanden ist, wird das Konzept des Unsicherheitsintervalls für Werte verwendet. Bereichsangaben für unsichere Werte sind in der Technik üblich, während sich Wahrscheinlichkeiten für Werte nur schwer angeben lassen. In anderen Bereichen, z.B. der Medizin, ist es dagegen üblich mit Wahrscheinlichkeiten zu arbeiten, z.B. bei MYCIN, einem Medizinischen Diagnose-Expertensystem [Appel85].

### 2.2.1 Bestimmung der Wahrscheinlichkeit einer Aussage

Es sei  $A$  eine Aussage, die von  $n$  unsicheren Werten  $X_1, X_2, \dots, X_n$  abhängt. Die Werte  $X_i$  liegen jeweils in dem Intervall  $X_i \in [X_{min_i}, X_{max_i}]$ . Die Wahrscheinlichkeit  $P(A)$ , daß eine Aussage wahr ist, läßt sich wie folgt berechnen:

$$P(A) = \frac{p}{p_{ges}}$$

mit

$$p = \int_{X_{min_n}}^{X_{max_n}} \dots \int_{X_{min_2}}^{X_{max_2}} \int_{X_{min_1}}^{X_{max_1}} a(A(X_1, X_2, \dots, X_n)) dX_1 dX_2 \dots dX_n,$$

$$p_{ges} = \int_{X_{min_n}}^{X_{max_n}} \dots \int_{X_{min_2}}^{X_{max_2}} \int_{X_{min_1}}^{X_{max_1}} dX_1 dX_2 \dots dX_n = \prod_{i=1}^n (X_{max_i} - X_{min_i})$$

und

$$a(A(X_1, X_2, \dots, X_n)) = 1, \quad \text{wenn die Aussage } A \text{ mit den aktuellen } X_i \text{ wahr ist}$$

$$a(A(X_1, X_2, \dots, X_n)) = 0, \quad \text{wenn die Aussage } A \text{ mit den aktuellen } X_i \text{ falsch ist}$$

Vorausgesetzt wird hierbei, daß die Werte  $X_i$  tatsächlich in dem Intervall zwischen  $X_{min_i}$  und  $X_{max_i}$  liegen und daß die Häufigkeitsdichte der Werte innerhalb der Intervalle konstant ist.

Beispiele, welche Integrale bei Aussagen  $A$ , die von ein bzw. zwei unsicheren Werten abhängig sind, zu berechnen sind, werden in den Abbildungen 2.1 und 2.2 dargestellt. In Abbildung 2.1 ergibt sich  $P$  zu  $1/3$ . In Abbildung 2.2 ergibt sich  $P$  als Quotient aus der schraffierten Fläche und der Fläche des Rechtecks, das die schraffierte Fläche umschließt.

Auch wenn die Funktionen  $a(A)$  analytisch gegeben sind, kann eine analytische Integration sehr aufwendig oder sogar unmöglich sein. Daher soll bzw. muß die Integration für  $p$  numerisch durchgeführt werden. Der einfachste Ansatz ist die Verwendung

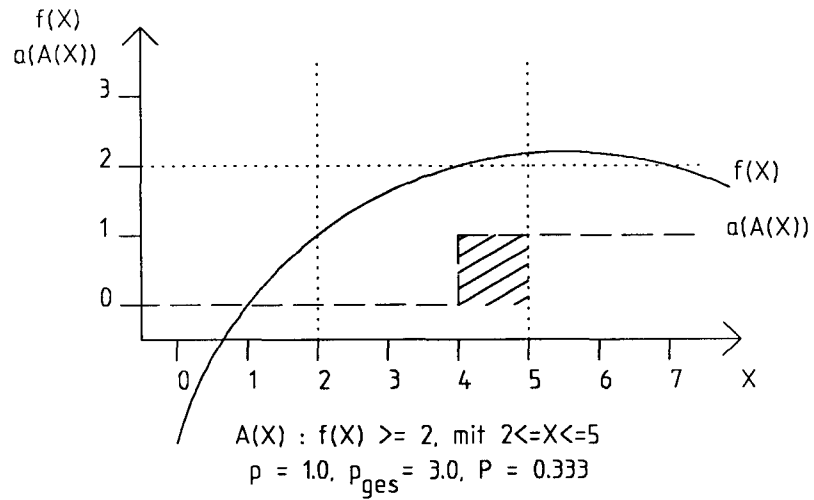


Abbildung 2.1: Zu berechnendes Integral für eine Aussage, die von einem unsicheren Wert abhängt

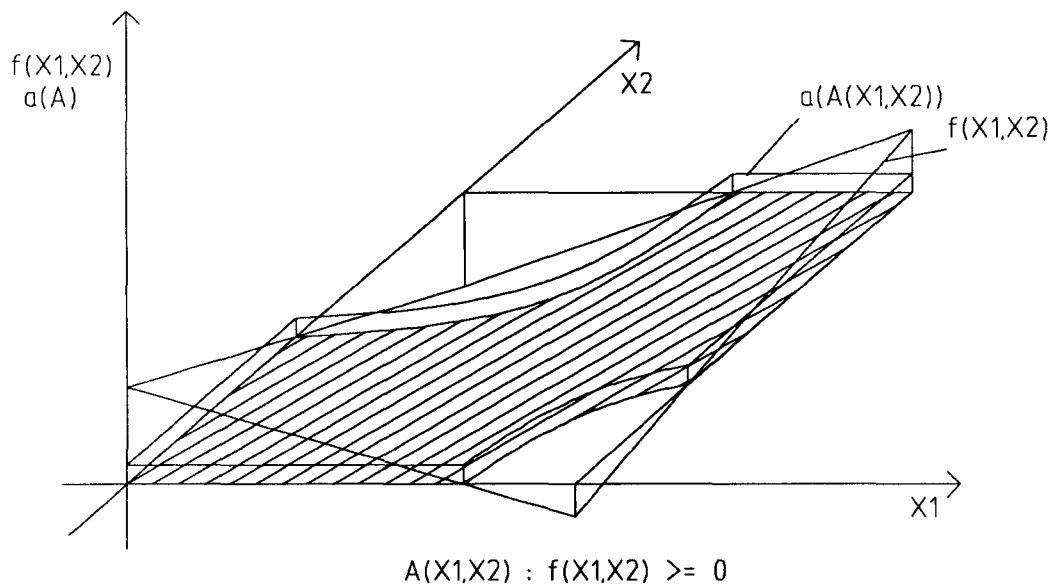


Abbildung 2.2: Zu berechnendes Integral für eine Aussage, die von zwei unsicheren Werten abhängt

der Trapezregel mit z.B.  $m = 10$  äquidistanten Stützstellen im Wertebereich jedes unsicheren Wertes.

$$p = \sum_{j_1=0}^m \dots \sum_{j_2}^m \sum_{j_n}^m a(A(X_{1,j_1}, X_{2,j_2}, \dots, X_{n,j_n}))g$$

mit

$$X_{i,j} = X_{min_i} + \frac{j_i}{m} (X_{max_i} - X_{min_i})$$

und

$$g = \prod_{i=1}^n g_i, \quad g_i = 0.5 \text{ f\"ur } j_i = 0 \text{ oder } j_i = m \\ \text{sonst } g_i = 1$$

Praktisch f"uhrt eine genaue Integration f"ur  $p$  bei gr"o"seren Anzahlen  $n$  von unsicheren Werten zu sehr vielen Wertekombinationen  $m^n$  und damit zu nicht mehr akzeptablen Rechenzeiten. Da f"ur die Projekt-Phase, f"ur die das Arbeiten mit unsicheren Werten konzipiert wurde, die absolute Genauigkeit nicht wesentlich ist, kann die Berechnung von  $p$  durch n"aherungsweise Integration nach der Monte-Carlo-Methode, [Hen78], durchgef"uhrt werden. D.h. f"ur eine vorgegebene Anzahl  $m$ , z.B.  $m = 50$ , von zuf"alligen Wertekombinationen  $(X_1, X_2, \dots, X_n)_j$  wird der Ausdruck  $a(A((X_1, X_2, \dots, X_n)_j))$  berechnet und daraus  $P(A)$  abgesch"atzt:

$$P(A) = \frac{1}{m} \sum_{j=1}^m a(A((X_1, X_2, \dots, X_n)_j))$$

### 2.2.2 Verkn"upfungen von Aussagen

Nach der in 2.2 vorgenommenen Definition ist das Ergebnis einer Aussage  $A$  die Wahrscheinlichkeit, da"b die Aussage wahr ist. Mit

|                |   |
|----------------|---|
| $A_i$          | ist eine beliebige Aussage,   |
| $P(A)$         | ist die Wahrscheinlichkeit, da"b die Aussage $A$ WAHR ist,              |
| $P(!A)$        | ist die Wahrscheinlichkeit, da"b die Aussage $A$ nicht WAHR ist,        |
| $P(A_1 + A_2)$ | ist die Wahrscheinlichkeit, da"b entweder $A_1$ oder $A_2$ WAHR ist und |
| $P(A_1 A_2)$   | ist die Wahrscheinlichkeit, da"b sowohl $A_1$ als auch $A_2$ WAHR sind  |

ergeben sich f"ur die Verkn"upfung von Aussagen aus der Wahrscheinlichkeitstheorie, siehe z.B. [Krapp73], folgende Rechenregeln:

$$P(!A) = 1 - P(A)$$

$$P(A_1 + A_2) = P(A_1) + P(A_2) - P(A_1 A_2)$$

Für stochastisch unabhängige Aussagen  $A_1, A_2$  ist

$$P(A_1 A_2) = P(A_1 \& A_2) = P(A_1)P(A_2)$$

und damit

$$\begin{aligned} P(A_1 + A_2) = P(A_1 | A_2) &= P(A_1) + P(A_2) - P(A_1)P(A_2) \\ &= 1 - ((1 - P(A_1))(1 - P(A_2))) \end{aligned}$$

Für zwei sich einander ausschließende Aussagen  $A_1, A_2$  ist

$$P(A_1 A_2) = 0$$

und damit

$$P(A_1 + A_2) = P(A_1 \| A_2) = P(A_1) + P(A_2)$$

Zur Unterscheidung, welche Rechenregel bei einer Oder-Verknüpfung anzuwenden ist, wird für stochastisch unabhängige Aussagen das Zeichen  $|$  und für einander ausschließende Aussagen das Zeichen  $\|$  definiert. Für die Und-Verknüpfung wird das Zeichen  $\&$  definiert.

Das Ergebnis von  $P(A_1 A_2)$  und  $P(A_1 + A_2)$  für stochastisch abhängige und sich nicht ausschließende Aussagen  $A_1, A_2$  läßt sich nicht auf die Wahrscheinlichkeiten der beiden Einzel-Aussagen zurückführen. Daher wird dieser Fall hier nicht berücksichtigt.

## 2.3 Datentypen und Datenmodelle

Die Daten der Objekte, auf die technische Regelwerke sich beziehen, können im allgemeinen folgenden drei Typen zugeordnet werden:

- reelle Zahlen
- ganze Zahlen
- Aufzählung (enumerated), ein Wert aus einer begrenzten Menge von möglichen Werten, meist Begriffen. Z.B. stammt die Klasse eines Schiffes aus der Menge (GL, ABS, DNV, BV, LR, ...).

Reelle Zahlen werden entsprechend dem in 2.2 angesprochenen Konzept als Zahlenpaare, bestehend aus Unter- und Obergrenze, behandelt.

Ganze Zahlen werden wie reelle Zahlen auch als Zahlenpaare behandelt.

In Aufzählungen können die einzelnen Elemente der Menge der möglichen Werte in sinnfälliger Weise sortiert werden. Jedem möglichen Wert kann dann, bei 0 beginnend, eine laufende Nummer zugewiesen werden<sup>6</sup>. Aufzählungen können dann wie

<sup>6</sup>Analogie zu dem Datentyp `enumerate` in C

ganze Zahlen behandelt werden. Unbekannte Werte sollten als Untergrenzen 0 und als Obergrenzen die Mächtigkeit der Menge der möglichen Werte minus 1 erhalten.

Um bei der formalen Beschreibung eines Regelwerkes Objekte, Teilbereiche von Objekten und Daten von Objekten ansprechen zu können, benötigt man eine eindeutig definierte Struktur der Daten, ein sogenanntes Daten- oder Produktmodell. Das Daten- bzw. Produktmodell und die Syntax der Bezeichnungen einzelner Werte darin hängen von der verwendeten Datenverwaltung oder Datenbank ab. In dieser Arbeit wird eine hierarchische Datenverwaltung mit entsprechend aufgebauten Objekt- und Datennamen verwendet, siehe [Jen90].

Die Definition eines bestimmten Typs von Objekten bzw. Werten in einem Daten- bzw. Produktmodell wird im folgenden Objekttyp bzw. Wertetyp genannt. Man sollte zur einfacheren Beschreibung von Werten in Regeln die Möglichkeit haben, einen Objekttyp oder Wertetyp relativ zu einem bereits beschriebenen Objekttyp anzugeben (relativer Objekttyp bzw. relativer Wertetyp).

Zu der Beispiel-Vorschrift aus 2.1.1 gehört das Datenmodell in C.1.3. Der Wert für die Länge zwischen den Loten kann hier mit `ship=Name.lpp` angesprochen werden. Der Wertetyp wäre dann `ship.lpp`.

## 2.4 Zusammenfassung

Vorschriften, einzelne Regeln, Voraussetzungen und Bedingungen in Regeln werden als logische Aussagen aufgefaßt. Das Ergebnis, ob eine Vorschrift erfüllt ist, berechnet sich mit logischen Verknüpfungen aus den Ergebnissen der Voraussetzungen und Bedingungen.

Es können unsichere und unbekannte Daten verwendet werden. Bei unsicheren Daten wird ein Intervall angegeben, in dem der Wert liegt.

Aus der Verwendung von unsicheren Daten folgt, daß Regeln usw. auch nicht sichere Ergebnisse haben können. Als Ergebnis, ob eine logische Aussage *WAHR* ist, wird die Wahrscheinlichkeit, mit der sie erfüllt ist, eingeführt. Die Verknüpfung von logischen Aussagen erfolgt damit nach den Rechenregeln der Wahrscheinlichkeitstheorie.

In den vorherigen Abschnitten sind folgende Annahmen gemacht worden:

- Die Voraussetzungen einer Liste von Kriterien in einer Regel schließen sich gegenseitig aus.
- Die Ergebnisse Regelauswertung der Teilbereiche eines Objekts sind voneinander unabhängig.
- Die Daten liegen innerhalb der angegebenen Intervalle, und die Häufigkeitsdichte in jedem Intervall ist konstant.

Die ersten beiden Annahmen sind in den in der Einleitung angesprochenen Vorschriften und ihren einzelnen Regeln durchweg erfüllt. Ob die letzte Annahme zutrifft,

| P (Vor.) | P (Bed.) | P (Kriterium) | P (AnsonstV.) | P (Regel)    |
|----------|----------|---------------|---------------|--------------|
| $x$      | $y$      | $xy$          | $1 - x$       | $1 - x + xy$ |
| 1.00     | $y$      | $y$           | 0.0           | $y$          |
| $x$      | 1.00     | $x$           | $1 - x$       | 1.00         |
| 0.0      | $y$      | 0.0           | 1.00          | 1.00         |
| $x$      | 0.0      | 0.0           | $1 - x$       | $1 - x$      |
| 0.50     | 0.50     | 0.25          | 0.50          | 0.75         |
| 1.00     | 1.00     | 1.00          | 0.0           | 1.00         |

Tabelle 2.1: Berechnung von  $P$  für eine Regel mit einem Kriterium

hängt von dem einzelnen Wert ab und davon, wie das Intervall festgelegt wurde. D.h. der Benutzer kann beeinflussen, ob oder wie stark diese Annahme verletzt wird.

Zur Verdeutlichung der Zusammenhänge der in den Kapiteln 2.1 und 2.2 vorgeschlagenen Berechnungen für die Wahrscheinlichkeit, daß eine Regel erfüllt ist, zeigt Tabelle 2.1 einige Beispiele für eine Regel mit einem Kriterium. Aus den gegebenen Wahrscheinlichkeiten der Voraussetzung und der Bedingung des Kriteriums (Spalten 1 und 2) ergibt sich die Wahrscheinlichkeit, mit der das Kriterium erfüllt ist (Spalte 3). Die Wahrscheinlichkeit für die Ansonst-Voraussetzung in Spalte 4 ergibt sich aus der Voraussetzung. Aus Kriterium und Ansonst-Voraussetzung wird dann die Wahrscheinlichkeit, daß die Regel erfüllt ist (Spalte 5) bestimmt. Die erste Zeile beschreibt das Ergebnis in der allgemeinsten Form.

### 3 Lösungsansatz zur Interpretation von Regelwerken

Der wesentliche Aspekt bei der Lösung der Aufgabe, Regelwerke mit dem Rechner zu bearbeiten, ist die Kommunikation mit dem Rechner. Diese Kommunikation gliedert sich in zwei Bereiche:

- Aufbereiten des Inhalts der Regelwerke für den Rechner und
- Anwenden der Regelwerke auf Objekte mit dem Rechner.

Der erste Punkt berührt dabei im wesentlichen, wie Regelwerke in eine Grammatik, die der Rechner verarbeiten kann, übersetzt werden. Der zweite Punkt beinhaltet die Steuerung und die Verständlichkeit der Ausgabe zu den einzelnen Aufgaben (*Inhalt, Relevanz, Prüfen, Dimensionieren*).

#### 3.1 Aufbereiten der Regelwerke

Die Regelwerke müssen, um sie mit dem Rechner bearbeiten zu können, von dem gedruckten Vorschriftentext nach einer einheitlichen Grammatik in eine für den Rechner verständliche Form übersetzt werden. Im folgenden wird der gedruckte Vorschriftentext als *Regeltext* und das Ergebnis der Übersetzung als *Regelcode* bezeichnet.

Für das Übersetzen müssen die Vorschriften in Regeln, wie sie in 2.1 definiert wurden, zerlegt werden. Die einzelnen Regeln können dann mit der im folgenden definierten Grammatik als Regelcode beschrieben werden.

Ziel der in 2.1 beschriebenen Zerlegung und der Grammatik ist eine möglichst weitgehende Ähnlichkeit zwischen dem gedrucktem Text, dem Regeltext, und der für den Rechner aufbereiteten Regeln, dem Regelcode. Nur dann ist die Übersetzung hinreichend einfach und damit sicher gegen Fehler.

Die Beispiele in diesem Abschnitt beziehen sich auf die fiktive Vorschrift aus 2.1.1.

Regeln bestehen aus dem Regelkopf und dem Regelrumpf. Der Regelkopf enthält den Namen der Regel und die Angabe, auf welchen Objekttyp sich die Regel bezieht. Die hier betrachteten Regelwerke können in einzelne hierarchische Regeln, meist analog zu den Abschnitten, Paragraphen, etc. aufgeteilt werden. Namen der Regeln werden aus dem Namen der übergeordneten Regel und dem lokalen Namen verbunden mit einem Punkt gebildet.

Für den Paragraphen 1 aus dem Beispiel Seite 13 könnte man als Namen `Beispiel.Paragraph1` verwenden. Der Objekttyp wäre in diesem Fall einfach `ship`, da die Regel auf alle zu einem Schiff gehörenden Ladefälle anzuwenden ist.

Ein Regelrumpf besteht aus den in 2.1 definierten Elementen, *füralle–* bzw. *einervon–Schleife* und einer Liste von Kriterien. Zusätzlich werden *rechnen–*Elemente

eingeführt, um innerhalb und außerhalb von Schleifen spezielle, für die Regel erforderliche Berechnungen durchführen zu können. Die Ergebnisse können als *lokal*, d.h. nur innerhalb dieser Regel bekannte Werte, oder als *global*, d.h. auch in folgenden Regelaufrufen bekannte Werte, deklariert werden. Berechnungen innerhalb der Schleifen werden für jedes Teilobjekt erneut durchgeführt.

Für Regeln, deren Ergebnisse nicht an Hand der Daten ermittelt werden können, kann der Regelrumpf aus dem Schlüsselwort **frage** und einem Fragetext bestehen. Das Regelergebnis wird dann aus der Datenbank entnommen oder während der Bearbeitung erfragt.

Um innerhalb der Voraussetzungen oder Bedingungen weitere Regeln aufrufen zu können, werden folgende Möglichkeiten angeboten:

1. Eine Regel wird unter Angabe ihres Namens und optional des Objektes aufgerufen (*Regelaufruf*). Der Name wird relativ zum Namen der aufrufenden Regel angegeben (*relativer Name*). Das Objekt wird ebenfalls relativ zum aktuellen Objekt angegeben (*relatives Objekt*). Zum Beispiel erfolgt der Aufruf der Paragraphen 2 und 3 aus Paragraph 1 heraus auf diesem Weg.
2. Für Verzweigungen innerhalb einer Regel können Regelrümpfe direkt als logische Ausdrücke angegeben werden (*innere Regel*). Diese Konstruktion könnte z.B. innerhalb der Regel für die Grenzwerte des Paragraphen 2 verwendet werden.

Für die Dokumentation der Ergebnisse können den verschiedenen Elementen Kommentare zugeordnet werden. Diese Kommentare sollten, soweit sinnvoll, dem Originaltext der Vorschrift entsprechen.

Im folgenden wird auf dieser Basis mit Hilfe einer erweiterten Backus-Naur-Form, EBNF (siehe Anhang B, Seite 66), eine kontextfreie Syntax der Regelgrammatik definiert.

Die Bedeutung der einzelnen nicht-terminalen Symbole ergibt sich im allgemeinen aus der Syntax oder wurde in diesem Abschnitt bzw. in 2.1 beschrieben.

Die vollständige Übersetzung der Beispielvorschrift mit dieser Syntax ist im Anhang unter C.1.2, Seite 69 wiedergegeben.

### 3.1.1 Regelsyntax

Die Syntax lehnt sich an den folgenden 'Satz' an:

```

REGELNAME für OBJEKTTYP muß
für alle bzw. einen von RELATIVEROBJEKTTYP gelten :
Wenn VORAUSSETZUNG 1 wahr ist
dann muß BEDINGUNG 1 wahr sein
oder
```

wenn VORAUSSETZUNG 2 wahr ist  
dann muß BEDINGUNG 2 wahr sein

...

Syntax :

|                     |   |  |
|---------------------|---|--|
| <i>Regelwerk</i>    | = | [ <i>Regelwerk</i> ] <i>Regel</i> .  |
| <i>Regel</i>        | = | <i>Regelkopf</i> { <i>Blank</i> }<br>“(” { <i>Blank</i> } ( <i>Regelrumpf</i>   <i>Frage</i> ) { <i>Blank</i> } “)” { <i>Blank</i> }.  |
| <i>Regelkopf</i>    | = | { <i>Blank</i> } “regel” { <i>Blank</i> } <i>Name</i> { <i>Blank</i> } “fuer” { <i>Blank</i> } <i>Objekttyp</i> [ <i>Kommentar</i> ].  |
| <i>Name</i>         | = | [ <i>Name</i> “. ”] <i>Teilname</i> .  |
| <i>Teilname</i>     | = | / Buchstabe / {/ Buchstabe /   / Ziffer /   “_”}.  |
| <i>Blank</i>        | = | / Leerzeichen /<br>  / Zeilenvorschub /<br>  / Seitenvorschub /<br>  / Tabulator /.  |
| <i>Kommentar</i>    | = | { <i>Blank</i> } “\” { <i>Blank</i> } {/alles außer “\”/} { <i>Blank</i> } “\” { <i>Blank</i> }.   |
| <i>Objekttyp</i>    | = | /Syntax entsprechend der verwendeten Datenbank /.  |
| <i>Frage</i>        | = | “frage” [ <i>Kommentar</i> <sup>7</sup> ]  |
| <i>Regelrumpf</i>   | = | ( [ <i>Rechnen</i> ] [ <i>Schleife</i> [ <i>Rechnen</i> ] ] [ <i>Kritliste</i> ] ).  |
| <i>Schleife</i>     | = | (“füralle”   “einervon”) { <i>Blank</i> } <i>relObjekttyp</i> [ <i>Kommentar</i> ].  |
| <i>relObjekttyp</i> | = | / Syntax entsprechend der verwendeten Datenbank /.   |
| <i>Kritliste</i>    | = | [ <i>Kritliste</i> ] <i>Kriterium</i><br>  “dann” { <i>Blank</i> }<br>“(” { <i>Blank</i> } <i>Bedingung</i> { <i>Blank</i> } “)” { <i>Blank</i> } [ <i>Kommentar</i> ] )<br>  <i>Tabelle</i> . |

---

<sup>7</sup>Der Text des Kommentars wird beim interaktiven Erfragen des Regelergebnisses verwendet.

|                                 |   |   |
|---------------------------------|---|---|
| <i>Kriterium</i>                | = | ( “wenn” {Blank}<br>“(” {Blank} Voraussetzung {Blank} “)” {Blank} [Kommentar]<br>“dann” {Blank}<br>“(” {Blank} Bedingung {Blank} “)” {Blank} [Kommentar] ).   |
| <i>Voraussetzung</i>            | = | <i>logAusdruck</i>   “default”.   |
| <i>Bedingung</i>                | = | <i>logAusdruck</i> .  |
| <i>logAusdruck</i>              | = | <i>undVerknüpf</i> [ {Blank} ( “ ”   “  ” ) {Blank} <i>logAusdruck</i> ].   |
| <i>undVerknüpf</i>              | = | <i>Aussage</i> [ {Blank} “&” {Blank} <i>undVerknüpf</i> ].  |
| <i>Aussage</i>                  | = | {Blank} “(” {Blank} <i>logAusdruck</i> {Blank} “)” {Blank}<br>  {Blank} “!” {Blank} <i>Aussage</i><br>  <i>Bereich</i><br>  <i>Vergleich</i><br>  <i>Regelaufruf</i><br>  <i>innereRegel</i><br>  “wahr”<br>  “falsch”. |
| <i>Bereich</i>                  | = | <i>arithAusdruck</i> {Blank} ( “<”   “<=” ) {Blank} <i>arithAusdruck</i><br>{Blank} ( “<”   “<=” ) {Blank} <i>arithAusdruck</i> [Kommentar].  |
| <i>Vergleich</i>                | = | <i>arithAusdruck</i><br>{Blank} ( “==”   “!=”   “>”   “>=”   “<=”   “<” )<br>{Blank} <i>arithAusdruck</i> [Kommentar].  |
| <i>Tabelle</i>                  | = | tabelle” {Blank} “(” {Blank} <i>TabellenDef</i> {Blank} “,” {Blank}<br><i>TabellenDef</i> {Blank} “;” {Blank} <i>TabBedListe</i> {Blank} “)”.   |
| <i>TabellenDef</i> <sup>8</sup> | = | / ganze Zahl / {Blank} “,” {Blank} <i>arithAusdruck</i> .   |
| <i>TabBedListe</i>              | = | [ <i>TabBedListe</i> ] <i>TabBed</i> .  |
| <i>TabBed</i>                   | = | “(” {Blank} / ganze Zahl / {Blank} “,” {Blank} / ganze Zahl /<br>{Blank} “)” <sup>9</sup> {Blank} <i>Bedingung</i> {Blank} “;” [Kommentar].   |

---

<sup>8</sup>Dimension der Tabelle und arithmetischer Ausdruck, mit dem der Index berechnet werden soll.

*arithAusdruck* = / arithmetischer Ausdruck in C-Syntax [Ker83] mit *relWerttyp*, *Variablenname* und *EnumKonst* als 'Variable Identifier'. Funktionen der Standard Bibliotheken können benutzt werden. Andere Funktionen müssen in das Programm integriert werden. /.

*relWerttyp* = / Syntax entsprechend der verwendeten Datenbank /.

*Variablenname* = / Buchstabe / {/ Buchstabe / | / Ziffer /}.

*EnumKonst*<sup>10</sup> = {*Blank*} ““” {*Blank*} / Buchstabe / {/ Buchstabe / | / Ziffer /} {*Blank*} ““” {*Blank*}.

*Regelaufruf* = {*Blank*} “->” {*Blank*} *relativerName*  
 [{*Blank*} “(” {*Blank*} *relObjekt* {*Blank*} “)” {*Blank*}].

*relativerName* = (({”^“} [“.”]) | “.”) *Name*<sup>11</sup>.

*relObjekt* = / Syntax entsprechend der verwendeten Datenbank /.

*innereRegel* = *Regelrumpf*.

*Rechnen* = {*Blank*} “rechnen” {*Blank*}  
 “(” {*Blank*} *Rechenrumpf* {*Blank*} “)” {*Blank*}  
 [*Kommentar*].

*Rechenrumpf* = [*Deklliste*] [*Zuweisliste*].

*Deklliste* = [*Deklliste*] *Deklaration*.

*Deklaration* = (“intlokal” | “intglobal” | “reallokal” | “realglobal”)<sup>12</sup>  
 {*Blank*} *Variablenname*  
 [{*Blank*} “,” {*Blank*} *Variablenname*] {*Blank*} “;” {*Blank*}.

*Zuweisliste* = [*Zuweisliste*] *Zuweisung*.

<sup>9</sup>Indizes der Bedingungen, beginnend bei 0.

<sup>10</sup>*EnumKonst* müssen im zugehörigen Datenmodell zulässige Werte der Aufzählung sein.

<sup>11</sup>Vorangestellte  $\wedge$  bei *relativerName* bewirken, daß der Name der aufrufenden Regel, an den der relative Name angehängt werden soll, für je ein Zeichen  $\wedge$  um einen Teilnamen von hinten aufgekürzt wird.

<sup>12</sup>“int...” für ganze Zahlen, “real...” für reelle Zahlen,

“...lokal” für Werte, die nur innerhalb der aktuellen Regel existieren,

“...global” für Werte, in der aktuellen und allen daraus aufgerufenen Regeln gelten.

*Zuweisung* = (*relWerttyp* | *Variablenname*) {*Blank*} “=” {*Blank*} *arithAusdruck* {*Blank*} “;” {*Blank*}.

Die Syntax von *Objekttyp*, *relObjekttyp*, *relObjekt* und *relWerttyp* ist von der verwendeten Datenbank oder Datenverwaltung abhängig. Bei der in den Beispielen benutzten Datenverwaltung JOSEPH [Jen90] gilt die folgende Syntax:

*Objekttyp* = *Name*.  
*relObjekttyp* = (({“^”} [“.”]) | “.”)<sup>13</sup>[*dbName*].  
*relObjekt* = *relObjekttyp*.  
*relWerttyp* = *relObjekttyp*.  
*dbName* = [*dbName* “.”] *dbTeilname*.  
*dbTeilname* = *Teilname* [“=” *Teilname*].

Durch diese Definition von *dbTeilname* muß bei *Zuweisung* mindestens ein *Blank* rechts oder links vom Gleichheitszeichen vorhanden sein.

## 3.2 Prüfen von Objekten

Wenn geprüft werden soll, ob eine bestimmte Regel für ein gegebenes Objekt erfüllt oder nicht erfüllt ist, wird mit den in 2.1 und 2.2 beschriebenen Rechenvorschriften die Wahrscheinlichkeit berechnet, mit der die Regel erfüllt ist.

Hierzu werden, wenn eine Schleife vorhanden ist, zuerst alle der Datenverwaltung bekannten Teilobjekte des angegebenen Typs ermittelt und dann für jedes Teilobjekt das Ergebnis der Kriterienliste berechnet. Die Ergebnisse der Voraussetzungen werden immer alle berechnet. Die Ergebnisse der Bedingungen werden nur berechnet, wenn das Ergebnis der zugehörigen Voraussetzung nicht 0 ist.

Die Ergebnisse der Voraussetzungen und Bedingungen werden unter Berücksichtigung der in ihnen vorkommenden Regelaufrufe, inneren Regeln, Vergleiche etc. berechnet. Diese Vorgehensweise wird im Sprachgebrauch der künstlichen Intelligenz als *forward chaining* bezeichnet. Die Elemente in *logAusdruck* und *undVerknüpf* werden von links nach rechts berechnet.

Die Berechnung der Ergebnisse von *Vergleichen*, *Bereichen* und *Tabellen* bei unsicheren Werten erfolgt für jeden *Vergleich* etc. getrennt.

### 3.2.1 Ausgabemodi

Für die Darstellung der Ergebnisse werden zwei Möglichkeiten vorgeschlagen:

---

<sup>13</sup>Siehe Fußnote zu *relativerName*

1. chronologische Ausgabe,
2. tabellarische Ausgabe.

Die chronologische Ausgabe gibt alle Details der Berechnung wieder. Sie erfolgt in der Reihenfolge der Bearbeitung. Die Ausgabe von Regelaufrufen, inneren Regeln und Vergleichen erfolgt geschachtelt innerhalb der Ausgabe der aufrufenden Bedingung oder Voraussetzung.

Das Ergebnis wird in Anlehnung an folgenden Satz ausgegeben:

```

REGELNAME für das Objekt OBJEKTNAME :
Mit dem Ergebnis der Voraussetzung 1 V = x.xxx
und dem Ergebnis der Bedingung 1 B = x.xxx
ergibt das Kriterium 1 K = x.xxx
und mit dem Ergebnis der Voraussetzung 2 V = x.xxx
und dem Ergebnis der Bedingung 2 B = x.xxx
ergibt das Kriterium 2 K = x.xxx
...
und damit ergibt sich das Ergebnis der Regel
REGELNAME zu P = x.xxx .

```

Zusätzlich können alle in der Regel angegebenen Kommentare und die Ergebnisse von Berechnungen ausgegeben werden (Beispiel siehe Anhang C: C.1.5.4, Seite 81).

Für eine bessere Übersicht wird eine tabellarische Ausgabe entsprechend Abbildung 3.1 vorgeschlagen, in der einige Zwischenergebnisse nicht mit aufgeführt werden (Beispiele siehe Anhang C: C.1.5.1 bis C.1.5.3, ab Seite 73).

Um kürzere und damit besser kontrollierbare Ausgaben zu erhalten, können nicht benötigte Informationen unterdrückt werden, wie:

- rekursive Regelaufrufe
- Kriterien, deren Voraussetzung  $V = 0$  ist
- alle Kriterien
- innere Regeln
- Teilbereiche, auf die die Regel nicht angewendet werden muß
- Regelaufrufe und innere Regeln von Regeln, die sicher ( $P = 1.0$ ) erfüllt sind
- Kommentare

In den meisten Fällen wird eine erste, kurze tabellarische Übersicht genügen, die über das Erfülltsein der Regel informiert und wenn die Regel nicht erfüllt ist, die Regeln angibt, die nicht erfüllt sind. Wenn diese Information nicht ausreicht, kann das Ergebnis mit detaillierteren Informationen erneut angezeigt werden.

In C.1.5.1 wird die geringst sinnvolle Menge an Information gezeigt. Es werden nur Regeln, die nicht erfüllt sind und alle Teilbereiche, auf die die Regel angewendet werden muß, ausgegeben. C.1.5.2 zeigt zusätzlich die Kriterien und inneren Regeln, während

```
+-----+
| Prüfen |
+-----+
```

Wahrscheinlichkeit, dass aus  
VORSCHRIFTNAME  
folgende Regel erfuehlt ist :

Regel REGELNAME  
fuer OBJEKTNAME

| Ebene | Regel               | angewendet auf        | er- gibt |
|-------|---------------------|-----------------------|----------|
| 0     | REGELNAME           | OBJEKTNAME            | x.xxx    |
|       | O. Kriterium        |                       | x.xxx    |
|       | Vorr.               |                       | x.xxx    |
|       | Bed.                |                       | x.xxx    |
| 1     | REGELNAME.REGELNAME | OBJEKTNAME.OBJEKTNAME | x.xxx    |
|       | -fuer alle Schleife | REOBJEKTYP            |          |

Abbildung 3.1: Tabellarische Ausgabe

in C.1.5.3 dann alle Informationen in tabellarischer Form ausgegeben werden. C.1.5.4 zeigt das Ergebnis der nicht erfüllten Regel desselben Beispiels als chronologischer Ausgabe mit der vollständigen Information.

Zusätzlich können alle Daten, die in eine Regel-Bearbeitung eingehen, aufgelistet werden.

### 3.3 Relevanz von Regeln

Um zu prüfen, ob eine Regel auf ein gegebenes Objekt anzuwenden ist, wird das Ergebnis der Regel wie in 3.2 beschrieben berechnet, wobei die in 2.1 für *Relevanz* beschriebenen Ergänzungen berücksichtigt werden. Das Ergebnis für jede einzelne Regel ist die Wahrscheinlichkeit, mit der sie auf das gegebene Objekt angewendet werden muß.

#### 3.3.1 Ausgabemodi

Die Ausgabe erfolgt nach den selben Schemata wie in 3.2.1 beschrieben.

Die chronologische Ausgabe erfolgt in Anlehnung an den Satz:

```
REGELNAME für das Objekt OBJEKTNAME:  
mit der Wahrscheinlichkeit P = x.xxx  
der Voraussetzung 1 VORAUSSETZUNG1  
muß die Bedingung 1 BEDINGUNG1 erfüllt werden  
und mit der Wahrscheinlichkeit P = x.xxx  
der Voraussetzung 2 VORAUSSETZUNG2  
muß die Bedingung 2 BEDINGUNG2 erfüllt werden  
...
```

Auch die tabellarische Ausgabe erfolgt wie in 3.2.1 beschrieben (Beispiele siehe Anhang C: C.1.8.1 und C.1.8.2).

### 3.4 Inhalt von Regelwerken

In diesem Fall, der nicht von einem konkreten Objekt abhängt, wird zusätzlich zu den bei *Relevanz* gemachten Ergänzungen, auch allen Voraussetzungen als Ergebnis der Wert *WAHR* zugewiesen (siehe 2.1).

Da dieser Bearbeitungsmodus meist nur für die Prüfung der Übersetzung des Regeltextes in den Regelcode genutzt wird, ist nur die chronologische Ausgabe sinnvoll (Beispiel siehe Anhang C: C.1.7).

### 3.5 Dimensionieren von Daten an Hand von Regeln

Dimensionieren von Daten an Hand von Regeln heißt, daß Daten, die unbekannt oder unsicher sind, so bestimmt werden, daß die Regeln erfüllt werden. Wenn unsichere Daten verwendet werden, heißt das, daß die Wahrscheinlichkeit, mit der die Regeln erfüllt werden, möglichst groß sein sollte. Ziel ist es, daß die Wahrscheinlichkeit 1 wird.

Üblicherweise werden beim Entwurf die Werte unbekannter Daten aus der Erfahrung oder mit Näherungsformeln abgeschätzt. Man nähert sich den endgültigen Werten dann iterativ (Entwurfsschleife). Manche Werte können direkt mit Hilfe der Vorschriften bestimmt werden.

Die hier vorgesehene automatische Dimensionierung erfolgt, indem eines der im folgenden beschriebenen Kriterien optimiert wird. Die freien Variablen bei dieser Optimierung sind die Unter- und Obergrenzen der zu dimensionierenden Daten.

Es ist vorgesehen, daß der Benutzer die Daten, die dimensioniert werden sollen, auswählt. In einem ersten Schritt kann hierfür mit *Prüfen* eine Liste aller Werte erstellt werden, die in die Regel und deren *Regelaufrufe* und *inneren Regeln* eingehen. Für jedes zu dimensionierende Datum muß der Suchraum, d.h. die maximal zulässige Obergrenze und die minimal zulässige Untergrenze, angegeben werden. Die Startwerte für die Optimierung werden auf die Grenzen des Suchraumes gesetzt.

Das erste Ziel der Dimensionierung ist, die Grenzen der zu dimensionierenden Daten so festzulegen, daß die Regeln erfüllt werden. Das naheliegendste Optimierungskriterium lautet daher:

$$\text{Maximiere } M_1 = P(O)$$

Die Ableitung der Funktion  $P(O)$  hat Unstetigkeitsstellen bei  $P(O) = 0$  und  $P(O) = 1$ . Das Optimierungskriterium kann aber wie folgt umformuliert werden, so daß die Unstetigkeiten in der Ableitung verschwinden.

$$\text{Maximiere } M_2 = \sin((P(O) - 0.5)\pi)$$

Damit dem Konstrukteur ein möglichst großer Spielraum für die endgültige Festlegung der Werte bleibt, ergibt sich als weiteres Kriterium, daß die Grenzen der zu dimensionierenden Werte möglichst weit auseinanderliegen sollten. Das heißt, der Abstand der Grenzen muß maximiert werden:

$$\text{Maximiere } M_3 = \frac{1}{n} \sum_{i=1}^n \frac{X_{max_i} - X_{min_i}}{X_{bezug_i}}$$

mit  $X_{bezug_i}$  als Mittelwert der Beträge der Grenzen des Suchraumes <sup>14</sup>.

Ein zusammengesetztes Optimierungskriterium kann wie folgt definiert werden:

$$\text{Maximiere } M_4 = M_1 + M_3G$$

oder in einer stetigen Form

$$\text{Maximiere } M_5 = M_2 + M_3G$$

Der Anteil aus dem Abstand der Grenzen  $M_3$  muß gewichtet werden. Der Faktor  $G$  sollte kleiner 0.1 sein. Das Optimum dieser zusammengesetzten Kriterien liegt allerdings, abhängig von  $G$ , außerhalb des Bereichs, in dem die Regel sicher erfüllt ist.

Daß die Untergrenzen kleiner als die Obergrenzen zu sein haben, sind die Nebenbedingungen  $N_x$ .

Als weitere Nebenbedingung kann eingeführt werden, daß die Regel erfüllt sein muß:

$$N_1 : P(O) \geq 1.0$$

oder mit einer in der Ableitung stetigen Funktion:

$$N_2 : \sin((P(O) - 0.5)\pi) \geq 1.0$$

Wenn es in dem vorgegebenen Suchraum keine Lösung gibt, bei der die Regel sicher erfüllt ist, sollte eines der zusammengesetzten Optimierungskriterien  $M_4$  oder  $M_5$

<sup>14</sup>Wenn ein Suchraum mit einer Ausdehnung  $(X_{max} - X_{min})$  größer 0 existiert, kann  $X_{bezug}$  nicht 0 werden

verwendet werden. Wenn die Regel sicher erfüllt werden kann, sollte als Kriterium der Abstand der Grenzen ( $M_3$ ) verwendet werden und eine der Nebenbedingungen, daß die Regel erfüllt sein soll ( $N_1$  oder  $N_2$ ), mit berücksichtigt werden.

Als Ergebnis der Dimensionierung wird neben einer Liste der aus der Dimensionierung ermittelten Grenzen die Ergebnisdarstellung wie beim Prüfen verwendet (siehe 3.2.1).

### 3.5.1 Verwendete Optimierungsmethoden

Als eine Optimierungsmethode wurde der Algorithmus OPT [Soe83] von Söding implementiert. Der Algorithmus OPT ist auf lineare und nichtlineare Probleme anwendbar. OPT minimiert eine reelle, stetige Zielfunktion mit reellen, stetigen Ungleichheits- und Gleichheitsnebenbedingungen in einem durch feste Grenzen für jede Unbekannte definierten Suchgebiet. Die Optimierung basiert auf der wiederholten Linearisierung der Zielfunktion und der Nebenbedingungen mit anschließender linearer Optimierung nach der modifizierten Simplexmethode aus [Soe85].

Alternativ wurde eine Zufallssuche als geschachtelte und exponentielle Zufallssuche implementiert (siehe z.B. [Now76]). Die Zufallssuche ist für multimodale und unstetige Zielfunktionen geeignet. Nach einer gegebenen Anzahl von Suchschritten wird die Suche in einem um einen Faktor, z.B. 0.5, kleineren Suchraum um das bis dahin gefundene Optimum fortgesetzt. Innerhalb des aktuellen Suchraumes werden die Suchvektoren häufiger in der Nähe des bisherigen Optimums angenommen. Hierzu werden die Zahlen  $x_i$  aus  $[-1, 1]$ , die unabhängige Zufallsgeneratoren mit konstanter Wahrscheinlichkeitsverteilung erzeugen, auf  $x'_i = \text{sign}(x)|x|^Z$  abgebildet. Die Zahlen  $x'_i$  liegen wieder im Bereich  $[-1, 1]$ , die Wahrscheinlichkeitsverteilung hat aber bei 0 ein Maximum für  $Z > 1$ . Aus den  $x'_i$ , dem aktuellen Suchraum und dem letzten Optimum werden dann die zufälligen Suchvektoren berechnet. Die Schachtelung und auch der Exponent  $Z$  können durch Parameter gesteuert werden.

Diese Implementation der Zufallssuche berücksichtigt Ungleichungen als Nebenbedingungen. Auch die Nebenbedingungen dürfen unstetig sein. Das letzte gefundene Optimum wird durch den aktuellen Suchschritt ersetzt, wenn der Suchschritt

- die Nebenbedingungen nicht verletzt, das letzte Optimum die Nebenbedingungen aber verletzt,
- die Nebenbedingungen nicht verletzt, das letzte Optimum die Nebenbedingungen auch nicht verletzt, aber der Suchschritt einen besseren Wert für des Optimierungskriterium liefert,
- die Nebenbedingungen weniger verletzt als das letzte Optimum.

Mit dieser Strategie wird erstrangig ein Lösungsvektor gesucht, der die Nebenbedingungen erfüllt.

| Vergleich | Ergebnis in zweiwertiger Logik                             |
|-----------|--|
| ==        | $links - rechts \leq grenz$ UND $links - rechts > -grenz$  |
| !=        | $links - rechts > grenz$ ODER $links - rechts \leq -grenz$ |
| <         | $links - rechts \leq -grenz$                               |
| <=        | $links - rechts \leq grenz$                                |
| >=        | $links - rechts > -grenz$                                  |
| >         | $links - rechts > grenz$                                   |

Vergleich: *links* Vergleich *rechts*  
*grenz* = 0.0000001 für reelle Zahlen  
*grenz* = 0.5 für ganze Zahlen und Attribute

Tabelle 3.1: Vergleiche

Für grobe Schätzungen und zur Abschätzung von Einflüssen einzelner Parameter wurde vorgesehen, daß der Benutzer jeden Optimierungsschritt auch selber eingeben kann.

### 3.6 Unbekannte und unsichere Daten

Für die Behandlung unbekannter Daten sind zwei Vorgehensweisen denkbar:

1. die Daten werden als unbekannt weiter behandelt, das heißt Ober- und Untergrenze werden auf einen sehr kleinen bzw. sehr großen Wert gesetzt (siehe 2.2),
2. das Programm erfragt die Werte der benötigten Daten vom Benutzer.

Die zweite Vorgehensweise ermöglicht es dem Benutzer mit einem Objekt zu arbeiten, für das noch keine Werte gespeichert wurden. Während der Bearbeitung werden nur die erforderlichen Werte vom System erfragt.

Eine weitere Möglichkeit bietet sich, wenn die Datenverwaltung Berechnungsmethoden mit verwaltet. Unbekannte Werte können dann, wie in [Jen90] beschrieben, auf der Basis bekannter Werte, berechnet werden.

Zwei reellen Zahlen werden als gleich angesehen, wenn der Betrag der Differenz der beiden Zahlen kleiner als ein Grenzwert, z.B. 0.0000001, ist. Bei den ganzen Zahlen und Attributen, die entsprechend Kapitel 2.3 auf reelle Zahlen abgebildet werden, wird dieser Grenzwert auf 0.5 gesetzt. Dieser Ansatz wird auch bei allen anderen Vergleichen verwendet (siehe Tabelle 3.1).

Für die numerische Integration von  $p$  (siehe 2.2.1) müssen für ganze Zahlen und Attribute die angegebenen Bereiche nach unten und oben um je 0.5 vergrößert werden.

Vergleiche und Bereichsabfragen, in die unbekannte Daten eingehen, erhalten als Ergebnis den Wert  $P = 0.5$ .

Durch unbekannte Werte kann es passieren, daß die Ergebnisse der Voraussetzungen nicht mehr voneinander unabhängig scheinen. In diesem Fall erhält die Ansonst-Voraussetzung auch das Ergebnis  $P = 0.5$ .

### 3.7 Berechnungen in Regelwerken

Für die in 3.1 mit *rechnen* eingeführten Berechnungen innerhalb der Regeln müssen die Ober- und Untergrenze der zu berechnenden Werte numerisch bestimmt werden. Man könnte dies z.B. mit dem Optimierungsalgorithmus OPT lösen, wie er in 3.5, *Dimensionieren* von Daten . . . , verwendet wurde.

Einfacher ist es, die Minima bzw. Maxima aus einer Anzahl von Ergebnissen zu suchen, die mit verschiedenen Wertekombinationen der unsicheren Werte, die in die Berechnung einfließen, berechnet wurden. Aus analogen Überlegungen wie in 2.2.1 zur Berechnung von  $p$ , werden auch hier zwei Methoden zur Bestimmung der Eingangswerte vorgeschlagen und implementiert:

1. für jeden unsicheren Eingangswert werden  $m$  äquidistante Stützstellen bestimmt und die Ergebnisse aus allen möglichen Werte-Kombinationen berechnet.
2. es werden  $m$  zufällige Wertekombinationen der unsicheren Eingangswerte verwendet.

Bei wenigen Eingangswerten sollte mit äquidistanten Stützstellen gerechnet werden.

## 4 Softwareengineering und Realisierung

### 4.1 Systemanalyse und Realisierungskonzept

Ein Programmsystem, das die in der Einleitung definierten Aufgaben für verschiedene Objekte und Regeln durchführen soll, kann in drei Moduln aufgeteilt werden, die nacheinander abgearbeitet werden:

1. Aufbereiten der Regelwerke und Bereitstellen des Inhalts (Wissensaquisition).
2. Ausführen der eigentlichen Aufgabe, z.B. Prüfen einer Regel in Bezug auf ein bestimmtes Objekt (Regelanwendungsmodul oder kurz Regelmodul).
3. Aufbereiten der Ergebnisse und Ausgabe (Erklärungskomponente).

Für die ersten beiden Moduln gibt es zwei mögliche Konzepte:

1. ein großes Modul, bestehend aus einem Interpreter, der den Regelcode aus einer Datei oder einer Datenbank liest, analysiert und anschließend sofort anwendet.
2. Getrennte Moduln für die Wissensaquisition und das Regelmodul. Ein Compiler analysiert den Regelcode und übersetzt ihn in eine Form, die das Regelmodul direkt verwenden kann, die Regeldatenstruktur oder kurz Regelstruktur, und speichert diese Struktur ab. Das Regelmodul arbeitet mit der gespeicherten Regelstruktur.

Für den Interpreter spricht, daß Regeln sofort angewendet werden können und daß schnelle Änderungen und Tests möglich sind.

Der Vorteil des Compiler-Konzepts sind kürzere Rechenzeiten bei der Bearbeitung. Dagegen spricht, daß zusätzliche Dateien der analysierten Regeln entstehen.

Da das Ziel mit diesem Instrument sein soll, größere Mengen von Objekten auf immer wieder dieselben Regeln hin zu untersuchen und nicht Regelwerke zu entwickeln, erscheint es sinnvoll, mit Compiler und getrenntem Regelmodul und den daraus resultierenden kürzeren Rechenzeiten zu arbeiten.

Bei der Benutzung eines Compilers gibt es zwei Möglichkeiten, die analysierten Regeln zu speichern und im Regelmodul zu verarbeiten:

1. Die Struktur der analysierten Regeln wird in Dateien oder auch einer geeigneten Datenbank gespeichert (Weg A, Abbildung 4.1). Hierbei müssen auch alle arithmetischen Berechnungen in Vergleichen, die Vergleiche selbst und die Berechnungen in den Syntax-Elementen *rechnen* soweit zerlegt werden, daß sie vom Regelmodul ausgeführt werden können.
2. Die Regelstruktur wird in einem ersten Schritt in eine Wirtssprache, z.B. Fortran oder C, übersetzt. Im zweiten Schritt wird der Wirtscode in Maschinensprache übersetzt (Objektcode) und mit einem allgemeinen Regelmodulkern gebunden (Weg B, Abbildung 4.2). Hierbei können die o.g. Berechnungen in Funktionen der Wirtssprache umgesetzt werden und damit schneller bearbeitet werden.

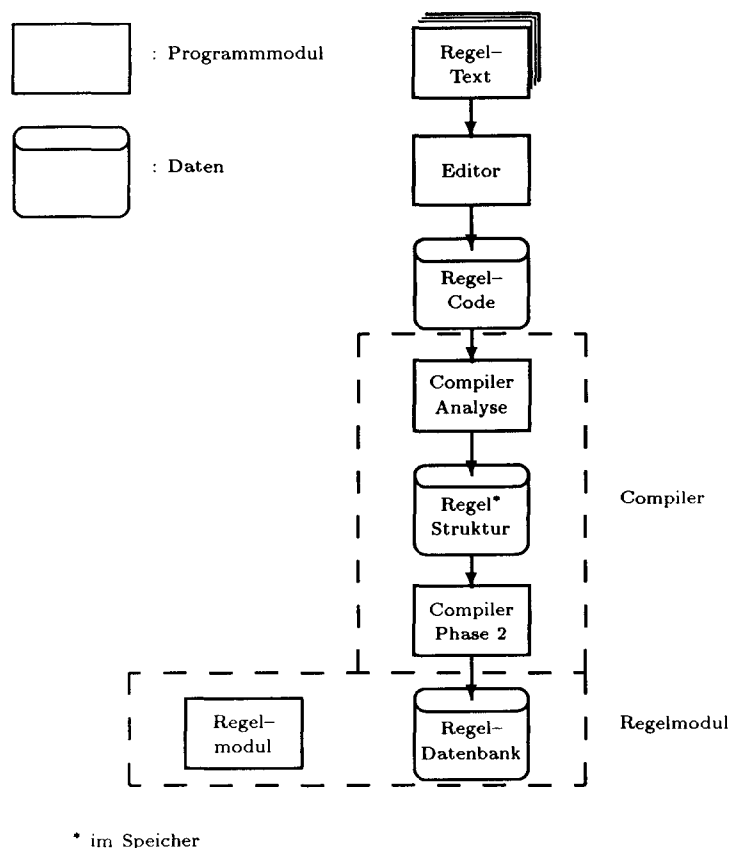


Abbildung 4.1: Weg A

Weg B führt zu kürzeren Rechenzeiten, aber auch zu großen ausführbaren Regel-Programmen, die jeweils nur ein Regelwerk bearbeiten. Letzteres hat zur Folge, daß komplexe Regelwerke nicht in einem Programm bearbeitet werden können und nicht von einem Regelwerk auf andere Regelwerke zugegriffen werden kann.

In einer ersten Implementierung wurde der Weg B realisiert, indem für jede Regel eine Funktion in C generiert wurde. Die sich daraus ergebenden Regelprogramme wurden sehr groß. In einer zweiten Implementierung wurden nur noch die rechenzeitintensiven arithmetischen Berechnungen in Vergleichen und *rechnen*-Elementen als C-Funktionen abgebildet. Der Rest wurde wie in Weg A bearbeitet (Weg AB, Abbildung 4.3).

Tests ergaben, daß der Weg AB nur geringfügig schneller gegenüber dem Weg A war. Da Weg A aber wesentlich flexibler ist, wurde für die endgültige Implementation dieser Weg gewählt. Man hat nur ein Programm und alle Regelstrukturen, auch verschiedener Regelwerke stehen immer über Datenbanken oder Dateien zur Verfügung.

Das dritte Modul, die Erklärungskomponente, wurde zuerst in das Regelmodul integriert. Wesentlich flexibler ist es, wenn das Aufbereiten der Ergebnisse und die Ausgabe von dem Regelmodul in einem Postprozessor getrennt durchgeführt werden, wie es endgültig realisiert wurde. Es ist damit möglich, die Ergebnisse einer Regel nach-

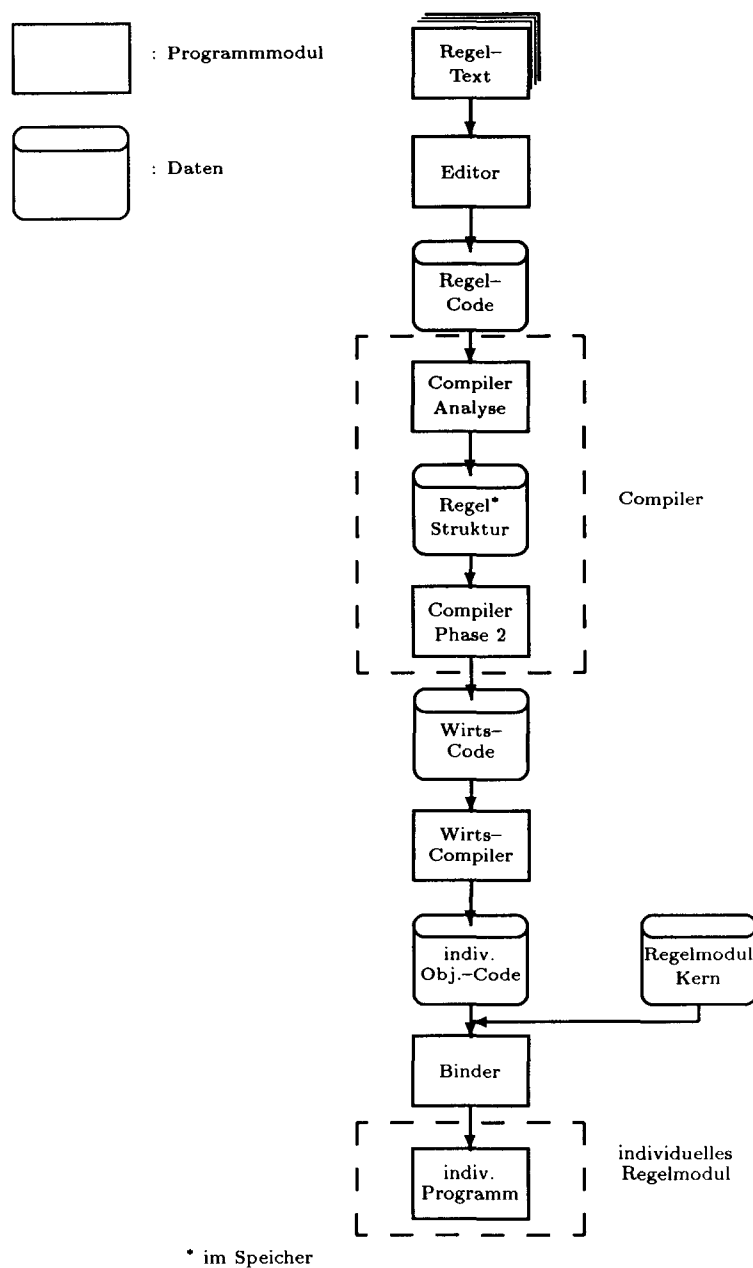


Abbildung 4.2: Weg B

einander in verschiedenen Ausgabemodi oder einzelne Teile der Ergebnisse auszugeben, ohne das Regelergebnis neu berechnen zu müssen. Die Ergebnisse werden über eine Datenbank von einem Modul an das andere übergeben. Die Ergebnisse können in der Datenbank permanent gespeichert werden, so daß zu einem späterem Zeitpunkt erneut darauf zugegriffen werden kann.

Jedes dieser drei Modulen wurde als eigenständiges Programm realisiert. Die Steuerung erfolgt über Parameter beim Programmaufruf.

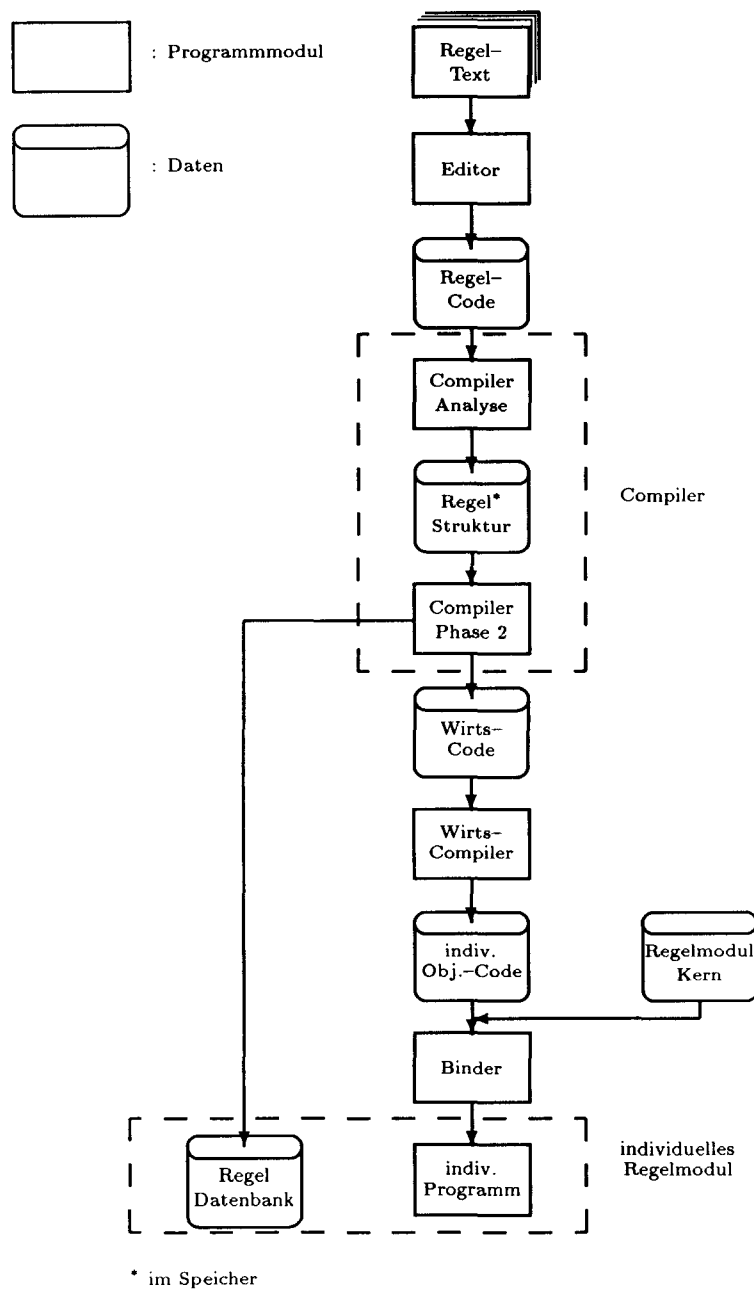


Abbildung 4.3: Weg AB

Für ein effektives Arbeiten ist die Benutzung dieser einzelnen Programme von der Betriebssystemebene aus nicht geeignet. Daher wurde auf der Basis einer einfachen, auf verschiedenen Betriebssystemen laufenden, fensterorientierten Benutzeroberfläche für ASCII-Terminals [Jen91] ein Steuerprogramm erstellt, das die Aufrufe der einzelnen Modulen sowie die Parameter für diese Aufrufe verwaltet (Abbildung 4.7). Zusätzlich zu den oben aufgeführten Modulen wurde der System-Editor zum Bearbeiten des Regelcodes und der Datei für die zu optimierenden Werte, die Objekt-Datenverwaltung sowie Hilfsmodule zum Auswählen von Regeln und Ergebnissen aus den entsprechenden Datenbanken eingebunden (siehe Abschnitt 4.5).

## 4.2 Compiler, Aufbereiten des Regelcodes

Der Benutzer muß den Regelttext in den in 3.1.1 definierten Regelcode überstezen und dem System als Textdatei zugänglich machen.

Der Compiler analysiert den Regelcode. Für jede Regel wird eine Datenstruktur *regel* erstellt. An diese Struktur werden, entsprechend der bearbeiteten Regel, weitere Datenstrukturen für Schleifen, Voraussetzung, Bedingungen, Vergleiche usw. angehängt, so daß ein sogenannter Strukturbaum entsteht. Nach der vollständigen Erstellung des Strukturbaumes einer Regel wird dieser in die Regel-Datenbank geschrieben und mit der Analyse der nächsten Regel fortgefahren. Ein vereinfachter Programmablaufplan ist als Nassi-Shneiderman-Diagramm in Abbildung 4.4 wiedergegeben.

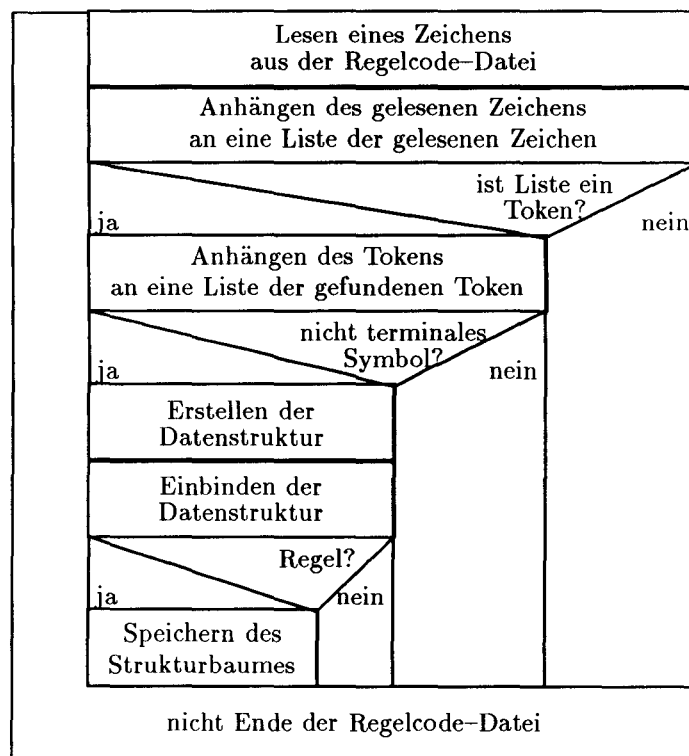


Abbildung 4.4: Programmablaufplan Compiler

Verknüpfungen von logischen und arithmetischen Ausdrücken werden als Befehlslisten in umgekehrter polnischer Notation gespeichert.

Die Steuerung erfolgt über Parameter beim Aufruf des Moduls (siehe Anhang D.1, S. 161).

### 4.2.1 Analyseroutinen

Die Syntax-Analyse erfolgt in zwei Schritten. Eine lexikalische Analyse zerlegt die Eingabe in terminale Symbole, im folgenden auch Token genannt, wie Schlüsselwörter, Zeichen, Namen, Zahlen und Leerzeichen. Diese Token werden an einen Parser weitergegeben, der aus der Reihenfolge der Token feststellt, welches der nicht-terminalen Symbole vorliegt. Wenn der Parser ein nicht-terminales Symbol gefunden hat, wird eine entsprechende Datenstruktur erstellt und in den Strukturbaum der Regel eingefügt.

Die Routine zur lexikalischen Analyse wurde mit Hilfe des Programms LEX [Lesk] generiert. Die Eingabe von LEX ist eine Sammlung von Beschreibungen der einzelnen Token und der vom Benutzer gewünschten Aktionen, wenn ein Token gefunden wurde. LEX generiert daraus eine C-Funktion, die einen Eingabestring analysieren kann. Zur Entlastung des Parsers wurde die Behandlung von *Blanks* in die lexikalische Analyse integriert.

Der Parser wurde mit Hilfe des Programms YACC [John] generiert. Die Eingabe von YACC ist eine Syntaxbeschreibung ähnlich der EBNF. Zusätzlich können, wie bei LEX, Benutzeraktionen als C-Code angegeben werden, die der Parser ausführt, wenn er ein nicht-terminales Symbol gefunden hat. Die von YACC generierte C-Funktion erwartet als Eingabe Token, wie sie von mit LEX generierten Funktionen geliefert werden. Außerdem überprüft YACC die vorgegebene Syntax auf logische Fehler.

Um den Analysevorgang verfolgen zu können, schreiben sowohl die lexikalische Analysefunktion als auch der Parser Meldungen über die gefundenen Token und nicht-terminalen Symbole in eine Protokolldatei. Auch sämtliche Fehlermeldungen werden in diese Datei geschrieben. Anhand dieser Datei können Syntaxfehler im Regelcode sicher gefunden werden.

Wenn der Parser einen Wertnamen findet, wird aus dem zugehörigen Datenmodell der Datentyp ermittelt. Damit wird gleichzeitig geprüft, ob der Werttyp im Datenmodell definiert ist. Bei Vergleichen mit Attributen werden die Attributkonstanten durch ihre Indizes in der Attributliste ersetzt.

### 4.2.2 Regel-Datenbank

Bei der Analyse wird der Strukturbaum einer Regel in einem zusammenhängenden Speicherbereich aufgebaut. Dieser zusammenhängende Bereich wird als Block binär in einer einfachen Datenbank bestehend aus einer Datendatei und einer Datei mit inversem Index verwaltet. Als Index wird eine Zeichenkette, bestehend aus Regelwerk-Name und Regelname, verwendet.

Nach dem Einlesen einer Regel durch das Regelmodul werden die Zeiger innerhalb des Blocks auf die neue Anfangsadresse umgerechnet und der komplette Strukturbaum der Regel steht zur Verfügung.

### 4.3 Regelmodul, Anwendung von Regeln auf Objekte

Die Bearbeitung der Regeln erfolgt für alle vier in der Einleitung definierten Aufgaben in demselben Modul. Nur an wenigen Stellen müssen Verzweigungen für unterschiedliche Initialisierungen oder Verknüpfungen verwendet werden.

Das Ergebnis einer Regel wird, ähnlich dem Strukturbaum für die Regeln, in eine Datenstruktur *pregel* sowie die weiteren sich daraus ergebenden Datenstrukturen geschrieben und dann analog zu den Regelstrukturen (4.2.2) in einer Ergebnis-Datenbank gespeichert. Als Index wird eine Zeichenkette, bestehend aus Aufgabe, Regelwerk-Name, Regelname, Objekt-Datenbank und Objekt-Name verwendet.

Die Algorithmen zum Berechnen des Ergebnisses einer Regel wurden in 2.1 beschrieben. Ein vereinfachter Programmablaufplan ist in Abbildung 4.5 wiedergegeben.

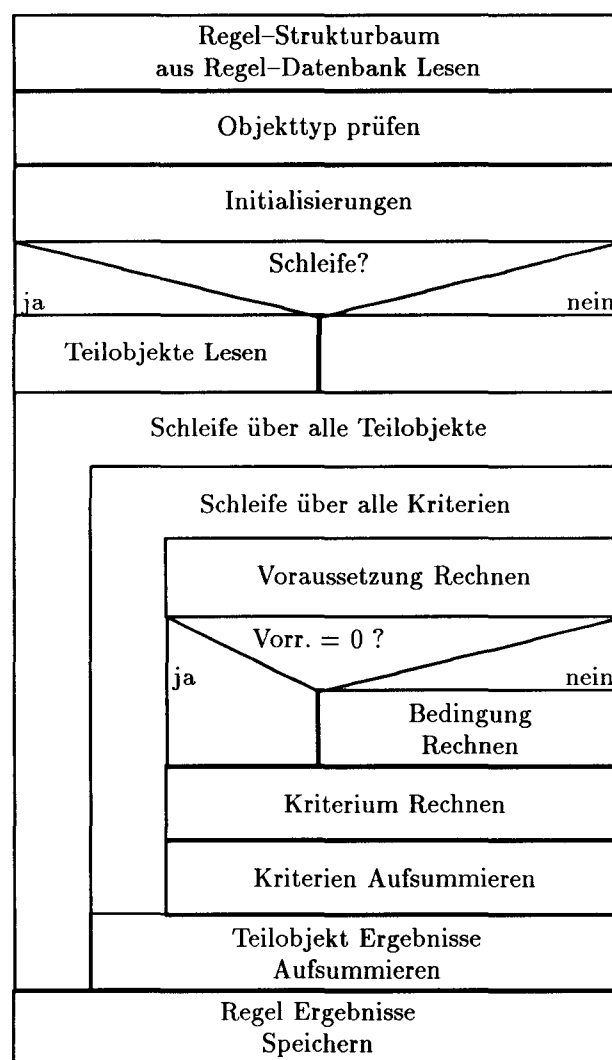


Abbildung 4.5: Programmablaufplan Regelmodul

Die Steuerung erfolgt wiederum über Parameter beim Aufruf des Moduls (siehe Anhang D.2, S. 162).

#### 4.3.1 Verwaltung der Objektdaten

Die Objektdaten werden in einem Stack verwaltet. Jeder Regel, die bearbeitet wird, wird der folgende Stackbereich zugeordnet. Dieser Bereich wird nach Abarbeiten der Regel wieder freigegeben.

Für globale berechnete Werte wird ein zweiter Stack eingerichtet, der die Namen der globalen Werte und einen Zeiger auf den Datenstack enthält.

Jeder Vergleich, jede Berechnung prüft, ob die benötigten Werte schon im Stack liegen. Wenn sie noch nicht vorhanden sind, werden sie aus der Datenbank gelesen. Wenn die Werte nicht in der Datenbank vorhanden sind, kann versucht werden, sie mit in der Datenbank vorhandenen Methoden zu berechnen oder interaktiv vom Benutzer zu erfragen. Berechnete und erfragte Werte werden in der Datenbank abgelegt (Abbildung 4.6).

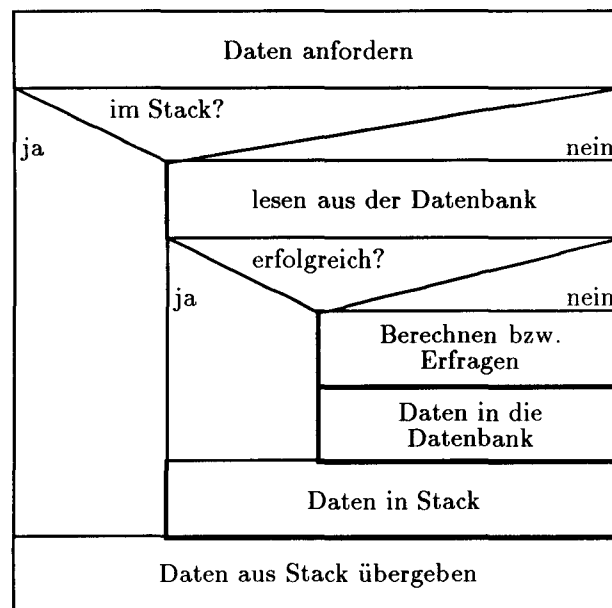


Abbildung 4.6: Unbekannte Daten

So werden Werte nur eingelesen, wenn sie tatsächlich benötigt werden. Sie werden nur einmal eingelesen und sie werden nur einmal berechnet bzw. erfragt.

## 4.4 Postprozessor, Darstellung der Ergebnisse

Der Postprozessor liest die Ergebnisse aus der entsprechenden Ergebnisdatenbank, formatiert sie und gibt sie in eine Text-Datei aus. Hierbei können die in 3.2.1 und 3.3.1 beschriebenen Ausgabemodi gewählt werden.

Die Steuerung erfolgt auch hier über Parameter beim Aufruf (siehe Anhang D.3, S. 164).

## 4.5 Benutzerschnittstelle RSHELL

Das Programm RSHELL stellt dem Benutzer Befehle zur Verfügung, um die drei Moduln Compiler, Regelmodul und Postprozessor sowie einige Hilfsmodule aufzurufen. Außerdem werden Befehle zur Verfügung gestellt, mit denen mehrere Moduln nacheinander aufgerufen werden. Zum Beispiel erfolgt nach dem Compilieren automatisch die Anzeige der Protokolldatei, wenn keine Fehler aufgetreten sind. Wenn Fehler aufgetreten sind, wird der Editor mit der Protokolldatei und der Regel-Source-Datei aufgerufen, so daß der Benutzer den Fehler suchen und korrigieren kann.

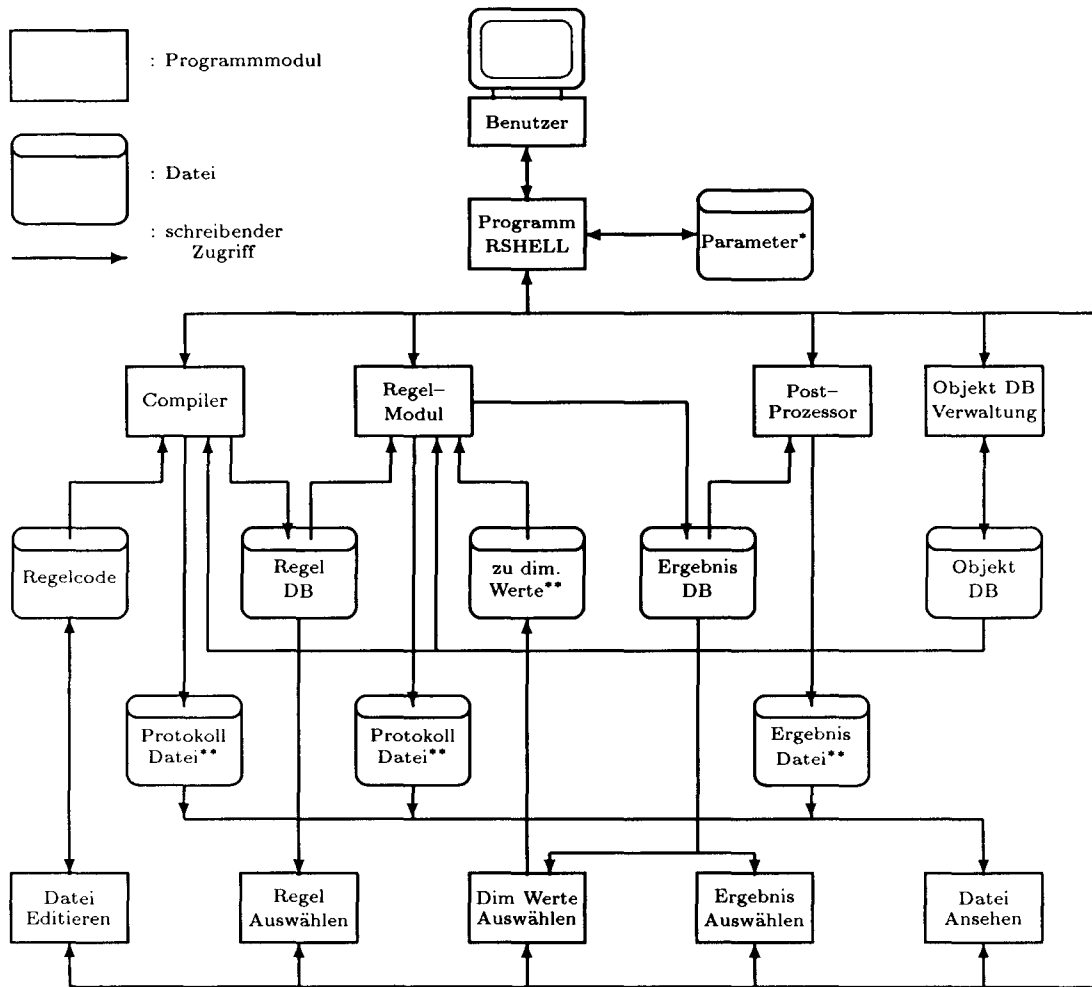
Als Hilfsmodule wurden der System-Editor, ein Programm zum Ansehen von ASCII-Dateien, die interaktive Schnittstelle zur Objektdatenbank und einige Routinen zum Setzen von Parametern integriert.

Die benötigten Parameter für jeden Aufruf eines Moduls werden einer gemeinsamen Parameter-Liste entnommen. Damit ist sichergestellt, daß z.B. beim Aufruf des Postprozessors nach einem Aufruf des Regelmoduls die Ergebnisse der vorher ausgeführten Berechnungen ausgegeben werden.

Der Zusammenhang der verschiedenen Moduln auf die verschiedenen Dateien und Datenbanken ist in Abbildung 4.7 dargestellt. Die Verbindungspfeile geben die schreibenden Zugriffe wieder.

RSHELL basiert auf der Funktionsbibliothek CW [Jen91] zum Verwalten von Fenstern auf ASCII-Terminals. Mit den Funktionen aus CW kann der Bildschirmbereich in mehrere Bereiche (Fenster) unterteilt werden. CW stellt verschieden Fenstertypen zur Verfügung, z.B. Fenster nur zum Anzeigen, zum Editieren etc. . Innerhalb der Fenster kann mit den Cursor-Tasten manövriert werden. Wenn der angezeigte Fensterausschnitt kleiner ist als der Inhalt, kann der Fensterausschnitt mit den Cursortasten in jede Richtung verschoben werden. Mit CONTROL-Cursor kann zwischen den Fenstern hin- und hergesprungen werden.

Von RSHELL wird der Bildschirm horizontal in zwei Bereiche geteilt (siehe Abbildung 4.8). In der oberen Bildschirmhälfte wird die Liste der Befehle aufgeführt. Mit den Cursortasten kann der gewünschte Befehl angewählt werden und mit der RETURN-Taste ausgeführt werden.



\* im Speicher  
 \*\* als Scratch-Dateien im Speicher möglich

Abbildung 4.7: Programm-Struktur des Benutzerschnittstelle RSHELL

Die untere Bildschirmhälfte wird vertikal in zwei Bereiche geteilt. In der linken Hälfte werden die Parameter beschrieben, in der rechten Hälfte können sie editiert werden.

Die vollständige Liste der Befehle und Parameter ist in Anhang D.4 aufgeführt.

Außer dem Editor wurden alle Moduln zu einem Programm gebunden, so daß keine Ladezeiten beim Aufruf entstehen.

### 4.6 Programmier-Sprache, Programmier-Umgebung

Als Programmiersprache wurde C [Ker83] gewählt. Entscheidungsgründe für C waren:

```

+-----RSHELL-----+
|   Bearbeiten   :           INHALT           |
|                :           RELEVANZ        |
|                :           PRUEFEN         |
|                :           DIMENSIONIEREN  |
|                : zu dim. Werte  AENDERN    |
|                : dim. Werte   SPEICHERN    |
|                : Protokolldatei ANSEHEN    |
|
|   Ausgabe     :           STARTEN          |
|                : letzte Ergebnisse ANSEHEN |
+-----+
| Allg. Parameter :           |
|   - Regel Source Datei | beispiel |
|   - Regel Datenbank   | bsp    |
|   - Vorschrift        | test   |
|   - Regel             | Paragraph_1 |
|   - Objektdatenbank   | testdb |
|   - Objekt            | ship=test |
|   - Ergebnisdatenbank oder SC | SC    |
|   - Ausgabedatei oder SC | SC    |
|
| Regelmodul   :           |
|   - Unbekannte lesen (0/1) | 1     |
+-----+

```

Abbildung 4.8: Bildschirm unter RSHELL

- dynamische Speicherverwaltung,
- rekursive Funktionsaufrufe,
- Werkzeuge für den Compilerbau waren verfügbar (YACC und LEX),
- verwendung selbstdefinierter Strukturen und Datentypen (die benötigten Strukturen für Regeln und Ergebnisse ließen sich direkt abbilden)

Hinzukommt, daß Compiler für C auf nahezu allen Rechnern existieren, so daß die Programme, die nur Sprachelemente und Funktionen nach dem ANSI-Standard verwenden, leicht portiert werden können.

Eine der Sprachen der sogenannten 4. Generation wie Prolog [Clock84], [Brat87] oder eine *Expert-System-Shell* zur besonders schnellen Erstellung von Expertensystemen wurde nicht gewählt. Generieren oder Verändern von Regeln und ähnliches, das von diesen Sprachen und Entwicklungswerkzeugen unterstützt wird, wurde für das hier behandelte Problem nicht benötigt.

Die Verwendung einer Expertensystemshell, wie sie für den Prototyp in der GL-Studie [Sans90] verwendet wurde, hat u.a. den Nachteil, daß die Struktur der Wissens-

basis nicht frei ist. Regelwerke können nicht so direkt abgebildet werden, wie es mit der speziell für diese Aufgabe definierten Regelcode-Syntax möglich ist.

H. Schildt [Schi87] stellt sogar die Behauptung auf, daß nicht nur alle KI-Techniken in C implementierbar wären, sondern in C auch noch 'sauberer' implementiert werden könnten als in den speziellen KI-Sprachen LISP und PROLOG.

Die Datenverwaltung für die Regel- und Ergebnis-Strukturen wurde in 4.2.2 beschrieben. Als Datenverwaltung für die Objekt-Daten wurde JOSEPH [Jen90] verwendet.

Die von der verwendeten Objekt-Datenverwaltung abhängige Manipulation von *Wertname* sowie das Lesen aus und Schreiben in die Objekt-Datenverwaltung erfolgt über von den anderen Teilen unabhängige Routinen, so daß eine andere Datenverwaltung oder Datenbank für die Objekte mit relativ geringem Aufwand integriert werden kann

Installiert und getestet wurde das System auf Atari ST unter TOS 1.04 mit dem C-Compiler TurboC [Bor89] .

## 5 Beispielanwendungen und Tests

### 5.1 Beispiel–Vorschrift

Die Beispiel–Vorschrift wurde entworfen, um an einem hinreichend kleinen, in sich geschlossenen Beispiel die volle Funktionalität der Moduln beschreiben und testen zu können. Es wurde ein *'Gesetzestext'* geschrieben, der in drei Paragraphen die wesentlichen Elemente eines Regelwerkes enthält (C.1.1). Dazu wurde ein Datenmodell definiert, das nur die für dieses Beispiel notwendigen Daten enthält (C.1.3).

#### 5.1.1 Übersetzen in den Regelcode

Das Übersetzen des *'Gesetzestextes'* entsprechend der Syntax aus 3.1.1 in den Regelcode ist in Anhang C.1.2 wiedergegeben und wurde schon in 2.1.1 und 3.1 kommentiert. Diese Übersetzung kann systematisch vorgenommen werden, ohne daß der Anwender die Hintergründe und auch die Zusammenhänge innerhalb der Vorschrift kennen muß. Nur an der Stelle in Paragraph 2:

für Zwischenwerte ist linear zu interpolieren

muß der Benutzer zusätzliche Informationen liefern.

Für eine verständliche Kommentierung der Ergebnisse ist es notwendig, die arithmetischen Ausdrücke der Vergleichsoperationen direkt mit Kommentaren zu versehen. Es ist aber relativ aufwendig, all diese Kommentartexte bei der Übersetzung zu formulieren. Bei der hier verwendeten Datenverwaltung mit sinnfälligen Namen im Datenmodell wäre es möglich, die Kommentare vom Rechner generieren zu lassen. Bei anderen Datenverwaltungen ist dies nicht gegeben, da statt Namen Indizes oder ähnliches angegeben werden müssen (z.B. COSMOS [Zieg87]).

Die Paragraphen 2 und 3 wurden geteilt in jeweils eine äußere Regel, die eine Schleife über alle Ladefälle und die Voraussetzungen 'leeres Schiff' bzw. 'nicht leeres Schiff' enthält und in eine weitere Regel, welche die Stabilitätskriterien enthält. Damit ist es möglich, die Stabilitätskriterien auch auf einzelne Ladefälle anzuwenden.

#### 5.1.2 Ergebnisse

Der Umfang der Ausgabe (C.1.5.1–C.1.8.2) zeigt, wie wichtig es ist, kurze und nur das wesentliche enthaltende Ausgaben zu erzeugen.

Die Ergebnisse der Überprüfung auf Erfülltsein der Vorschrift für das Test–Objekt, lassen sich unterschiedlich darstellen. In der tabellarischen Ausgabe mit nur den Regelaufrufen, deren Ergebnis ungleich 1.0 ist, kann die wesentliche Information, nämlich an welcher Stelle die Vorschrift nicht erfüllt ist, schnell gefunden werden (C.1.5.1, Seite 73). In diesem Beispiel ist das Kriterium zu Paragraph 2 für den Ladefall "c2"

nicht erfüllt. Da die Ausgabe von der Bearbeitung getrennt wurde, kann der Benutzer sich diese Regel dann auch sehr detailliert ansehen (C.1.5.4, Seite 81).

C.1.5.2 und C.1.5.3 zeigen verschiedene Zwischenstufen in der Detailierung der Ausgabe.

Anschließend wurde versucht, die unsicheren Werte der nicht erfüllten Regel `Paragraph_2.Kriterium` zu dimensionieren (siehe C.1.6). Das Ergebnis wird in 5.1.4 *Test der Dimensionierungsmöglichkeiten* eingehender diskutiert.

In C.1.7 wurde der Aufbau der Vorschrift mit *Inhalt* wiedergegeben. Dies wird in erster Linie zum Kontrollieren der Überstezung der Vorschrift in den Regelcode benötigt. Diese Funktionalität hilft aber auch dem Anwender am Rechner, sich über den Inhalt einer Vorschrift zu informieren. Oft ist es einfacher, die Vorschrift an Hand des Originaltextes zu verstehen.

Eine weitere, wesentliche Möglichkeit zeigen C.1.8.1 und C.1.8.2. Hier wird aufgezeigt, ob die Vorschrift auf das Objekt anzuwenden ist (*Relevanz*). Auf Seite 87 kann man schnell sehen, daß der Paragraph 2 nur auf die Ladefälle "c1" und "c2" anzuwenden ist, während Paragraph 3 auf den Ladefall "c0" anzuwenden ist.

### 5.1.3 Test der Integrationsmethoden

Zur Beurteilung der beiden in 2.2.1 vorgeschlagenen Integrationsmethoden wurden die Ergebnisse für die Regel

`Beispiel.Paragraph_2.Kriterium`

für das Objekt

`ship=stest.lc=c2`

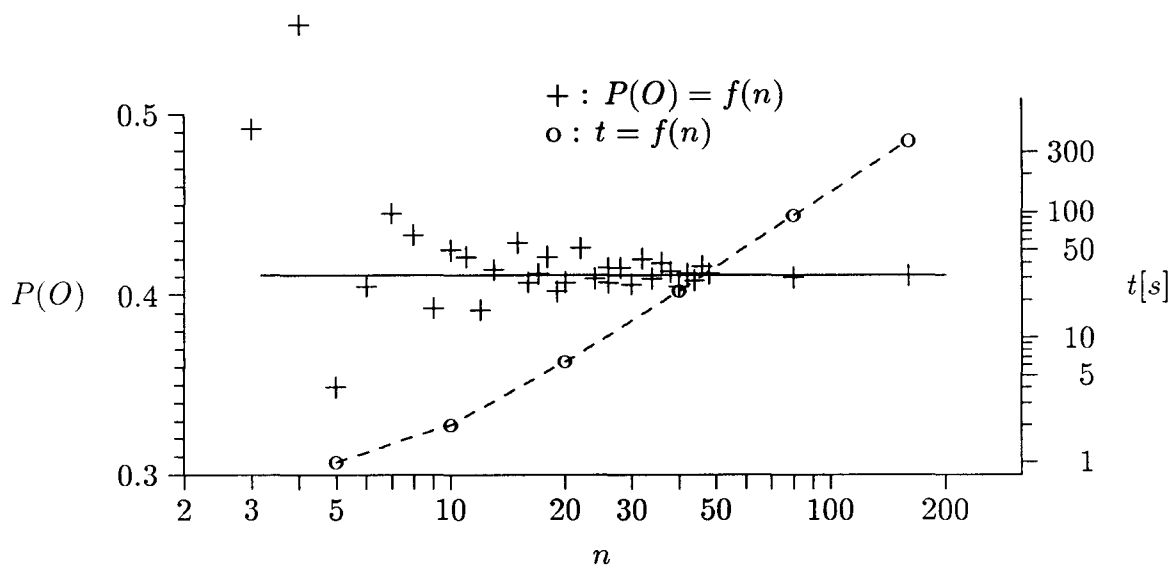
mit beiden Integrationsmethoden und verschiedenen Anzahlen von Stützstellen durchgerechnet. Mit den verwendeten Testdaten (Anhang C.1.4) müssen die Ergebnisse von je vier Vergleichen mit einem und mit zwei unsicheren Werten bestimmt werden.

Die Integration nach der Trapezregel mit äquidistanten Stützstellen ergaben die in Abbildung 5.1 wiedergegebenen Ergebnisse und Rechenzeiten.

Mit der Monte-Carlo-Integration wurden jeweils 20 Rechnungen durchgeführt und daraus die Standardabweichung berechnet (Abbildung 5.2). Der Mittelwert der Ergebnisse lag immer in der Nähe des exakten Wertes von 0.411.

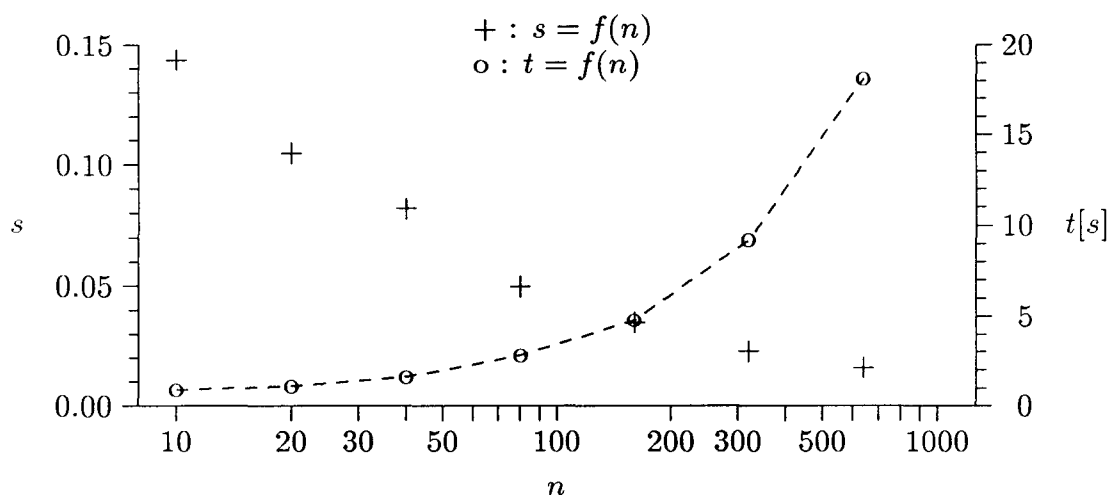
Als günstige Anzahlen von Stützstellen in Bezug auf Rechengenauigkeit und Rechenzeit können für die Integration mit der Trapezregel  $N = 10$  und bei der Monte-Carlo-Integration  $N = 50$  gewählt werden. Die Monte-Carlo-Integration ist zu bevorzugen, wenn häufiger mehr als zwei unsichere Werte in einem Vergleich vorkommen.

Die Berechnung von Werten mit dem Syntaxelement *rechnen* bei unsicheren Eingangswerten wurde jeweils mit derselben Methode wie die Integration durchgeführt. In



Erfülltsein der Regel  $P(O)$  und Rechenzeit  $t$  als Funktion der Anzahl der Stützstellen  $n$  je unbekanntem Wert

Abbildung 5.1: Integration mit äquidistanten Stützstellen



Mittlere, normierte quadratische Abweichung  $s$  und Rechenzeit  $t$  als Funktion der Anzahl der Stützstellen  $n$

Abbildung 5.2: Integration mit der Monte-Carlo-Methode

den meisten Fällen werden sich die Grenzen der Ergebnisse aus einer Kombination der Intervallgrenzen der Eingabewerte ergeben. Da die stochastische Methode diese Werte nicht zwingend erfaßt, ist es hier sinnvoll, mit äquidistanten Stützstellen zu rechnen, wobei wenige Stützstellen genügen.

#### 5.1.4 Test der Dimensionierungsmöglichkeiten

Aus dem schon im vorigen Abschnitt verwendeten Beispiel wurden die unsicheren Werte

`ship=test.lc=c2.gm`

und

`ship=test.lc=c2.range`

mit den verschiedenen Optimierungsmethoden, Kriterien und Nebenbedingungen (siehe 3.5) dimensioniert. Als Suchraum wurde für GM das Intervall  $[0.4, 0.8]$  und für den Umfang  $[30.0, 70.0]$  gewählt.

Die Methode OPT [Soe83] mit dem zusammengesetzten, stetigen Kriterium  $M_4$  findet ein Optimum. Die Regel kann in dem vorgegebenen Suchraum erfüllt werden. Das gefundene Optimum liegt bei Werten, die die Regel nicht sicher erfüllen. Eine Verkleinerung des Gewichtes  $G$  in dem Kriterium  $M_4$  verbessert das Ergebnis zwar, aber das Optimum liegt, bedingt durch den Ansatz, immer außerhalb des Bereichs, in dem die Regel erfüllt wird.

OPT arbeitet nur zuverlässig, wenn die lokalen Suchschritte, die ja numerische Ableitungen darstellen, hinreichend genau berechnet werden. Für diese Anwendung heißt das, daß die Integration mit der Trapezregel mit mindestens 50 Stützstellen durchgeführt werden muß. Kleinere Anzahlen oder die Verwendung der Monte-Carlo-Methode mit weniger als ca. 200 Stützstellen führen zum Abbruch von OPT.

Das vorliegende Beispiel wurde mit folgendem Ergebnis durchgerechnet:

```

Regel erfüllt = 0.977877,
Optimierungskriterium = 1.051961
ship=stest.lc=c2.gm = [0.492500,0.800000]
ship=stest.lc=c2.range = [41.250000,70.000000]
Anzahl Such-Schritte = 46
Bearbeitungszeit : 1714 sec.

```

Eine Erhöhung der Genauigkeiten sowohl in OPT als auch für die Integration ändert nichts wesentliches an dem Ergebnis, vor allem auch nicht die Anzahl der Suchschritte.

Für Dimensionierungsaufgaben, bei denen die Regel in dem gegebenen Suchraum erfüllt werden kann, wurde in 3.5 vorgeschlagen, als Kriterium den Abstand der Grenzen ( $M_3$ ) und die Nebenbedingung  $N_2$  zu verwenden. Dieser Ansatz war mit OPT nicht zu lösen. Nachdem das Optimum annähernd gefunden wurde, führte OPT einen globalen Suchschritt aus, der im allgemeinen ein Erfülltsein der Regel von 0 sowie die Verletzung der Nebenbedingungen  $N_i$  zur Folge hatte. Von diesem Punkt aus konnte OPT das Optimum nicht wieder erreichen.

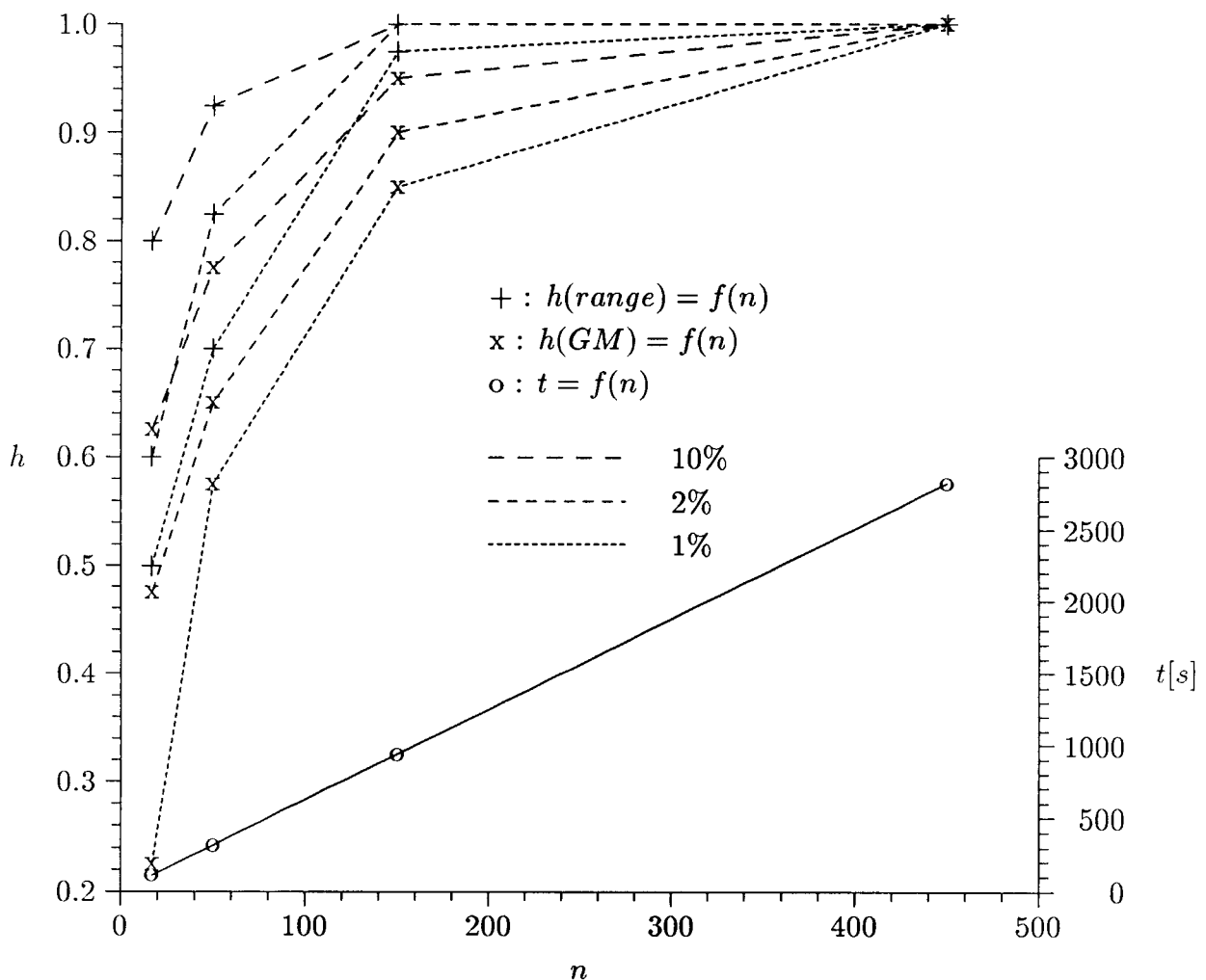
Das Problem ist in der Ableitung der Nebenbedingung  $N_2$  zu suchen, die für alle Werte-Kombinationen, die die Regel sicher erfüllen oder auch sicher nicht erfüllen den Wert 0 annimmt.

Mit der Zufallssuche wurde ebenfalls dimensioniert. Als Kriterium wurde hier  $M_3$

verwendet sowie die Nebenbedingung  $N_1$ .

Die Testrechnungen wurden mit 3 Schachtelungen auf jeweils 0.6 des vorherigen Bereichs und einem Exponenten  $Z$  von 2 durchgeführt. Diese Einstellungen erwiesen sich in vorangegangenen Testrechnungen als günstig. Für die Integration wurde die Trapezmethode mit 10 Stützstellen verwendet.

Mit verschiedenen Anzahlen von Suchschritten je Schachtelung wurden jeweils 40 Rechnungen durchgeführt. In Abbildung 5.3 ist die Wahrscheinlichkeit als Funktion der Anzahl der Suchschritte aufgetragen, mit der die Untergrenzen der zu dimensionierenden Werte **GM** und **range** nicht mehr als 1% , 2% oder 10% vom Optimum abweichen.



Wahrscheinlichkeit  $h$ , mit der die Abweichung vom Optimum kleiner als 1%, 2% oder 10% ist, als Funktion der Anzahl der Suchschritte  $n$  je Schachtelung

Abbildung 5.3: Dimensionieren mit Zufallssuche

Bei gleicher Rechenzeit wie mit OPT, ergibt sich nach Abbildung 5.3 eine Suchschrittzahl von 270. Die Untergrenze des GM wird damit mit einer Wahrscheinlichkeit von über 95% auf unter 2% entsprechend  $1\text{cm}$  genau bestimmt. Bei 100 Suchschritten, d.h. 38% der Rechenzeit von OPT, liegt diese Wahrscheinlichkeit noch bei 80%. Dies ist ausreichend.

Bei 100 Suchschritten und 3 Schachtelungen werden ca. 6 mal so viele Berechnungen durchgeführt werden, wie mit OPT. Weil die Genauigkeit zur Bestimmung der Integrale erheblich niedriger sein kann, ist die absolute Rechenzeit kürzer.

Für Dimensionierungsaufgaben, bei denen die Regel im Suchbereich erfüllt werden kann, erweist sich die Zufallssuche als günstiger.

## 5.2 IMO Resolution A167

Als zweites Beispiel wurde die IMO Resolution A167 [IMO87], Intakstabilität für Schiffe unter  $100\text{m}$  Länge, gewählt.

Die Übersetzung in den Regelcode (C.2.1) stößt bei einigen Regeln auf Schwierigkeiten. Es gibt in der Vorschrift Regeln, die nur einen hinweisenden Character haben. Zum Beispiel Regel 2.1:

```
2.1 Compliance with the stability criteria does not ensure
immunity against capsizing regardless of the circumstances
or absolve the master from his responsibilities. ...
```

Andere Regeln, wie Regel 7, die Inhalt und Aufbau der Stability information for the master vorschreibt, lassen sich nicht anhand technischer Daten prüfen. So etwas muß der Prüfer beurteilen und das Ergebnis dieser Regeln explizit angeben.

Für diese Fälle wurde das Syntax Element *Frage* geschaffen. Die hier beschriebenen Problematiken bestehen auch bei der Prüfprozedur ohne Rechnerunterstützung.

Das verwendete Datenmodell *ship* (Auszüge in C.2.2) entstand unabhängig von der Regelbearbeitung für ein Entwurfssystem mit der Daten- und Methodenverwaltung JOSEPH und deckt weite Bereiche des Schiffs-Entwurfs ab.

Die Beispieldaten (C.2.3) beschreiben die Stabilitätskennwerte eines Schiffes für mehrere Ladefälle, wobei ein Ladefall den Vorschriften nicht genügt.

Die komprimierteste Ausgabe der Ergebnisse von *Prüfen* (C.2.4) zeigt, daß die GM-Forderung beim Ladefall "Reise6" nicht erfüllt ist. Eine detailliertere Ausgabe wird nicht benötigt.

Mit *Relevanz* (C.2.5) erhält man einen guten Überblick über die Teile der Vorschrift, die für dieses Schiff anzuwenden sind.

### 5.3 SOLAS 83 II-2

Als letztes Beispiel wurde ein Teil aus SOLAS 74 einschließlich Ergänzungen bis '83 [SOLAS83] Kapitel II-2, Brandschutz, Feueranzeige und Feuerlöschung, gewählt. In den Regelcode wurden neben den Regeln, die allgemein den Anwendungsbereich der Vorschrift regeln (Part A, Reg. 1), diejenigen, die die Isolation von Schotten auf Frachtschiffe regeln (Part C, Reg. 43 und 44), übersetzt (C.3.1).

Das wesentliche Problem beim Umsetzen dieser Vorschrift war der Zugriff auf die benötigten Objekt-Daten. Für topologische Beschreibungen ist eine hierarchische Datenverwaltung nicht mehr geeignet. Mit der hier verwendeten Datenverwaltung JOSEPH und dem Datenmodell `ship` (die relevanten Teile in C.3.2) wird z.B. ein Raum (`room`) durch die Schotte (`surf`) beschrieben, die den Raum begrenzen. Zur Bestimmung der Räume, die einen beliebigen Teil eines beliebigen Schottes berühren, sind entweder zeitaufwendige Suchschleifen oder redundante Werte erforderlich. In dem Datenmodell `ship` wurde mit redundanten Werten gearbeitet.

Die Tabelle aus Regel 44, die die Mindestanforderungen an ein Schott zwischen zwei Räumen definiert, konnte direkt durch das Syntaxelement *tabelle* dargestellt werden.

Die Beispieldaten (C.3.3) beschreiben die Räume, Schotte etc. des in Abbildung 5.4 gezeigten Aufbaudecks. Die Schotte wurden mit `lxxx` bzw. `qxxx` bezeichnet. Mit `px` werden einzelne Bereiche eines Schotts benannt.

Die Beispieldaten wurden auf Einhaltung der Vorschrift überprüft. Elf der insgesamt 18 Schotte begrenzen Wohnbereiche. Davon erfüllen vier die Regel 43.2 nicht. Die Isolierung entspricht nicht mindestens dem 'C' Standard (C.3.4, Seite 151). Außerdem sind 5 Schotte nicht entsprechend den Forderungen der Tabelle 44 isoliert. Es sind zum Teil dieselben Schotte, die der Regel 43.2 schon nicht entsprachen.

Für das Schott `lpm2` zwischen `Galley` und `Staircase` wurde die erforderliche Isolierung bestimmt (C.3.5). Der Mindestwert 4 entspricht einer Isolierung nach A0 (siehe Datenmodell für `insclass`).

### 5.4 Ressourcenverbrauch

In der Tabelle 5.1 sind die Größen der Dateien in den verschiedenen Bearbeitungsschritten aufgeführt.

Anhand dieser Angaben kann man die in Tabelle 5.2 zusammengestellten Größen für andere Vorschriftenwerke abschätzen. Im Mittel kann mit 4 Regeln pro kB Text gerechnet werden, wobei die Anzahl der Regeln im Regeltext und übersetzt in den Regelcode in etwa gleich ist. Die Dateigrößen für den Regelcode können mit einem Faktor von 1.2 gegenüber dem Regeltext angenommen werden. Die Regelstrukturdatenbank benötigt etwa 1kB pro Regel, d.h. sie benötigt etwa den 4 fachen Speicherplatz wie der

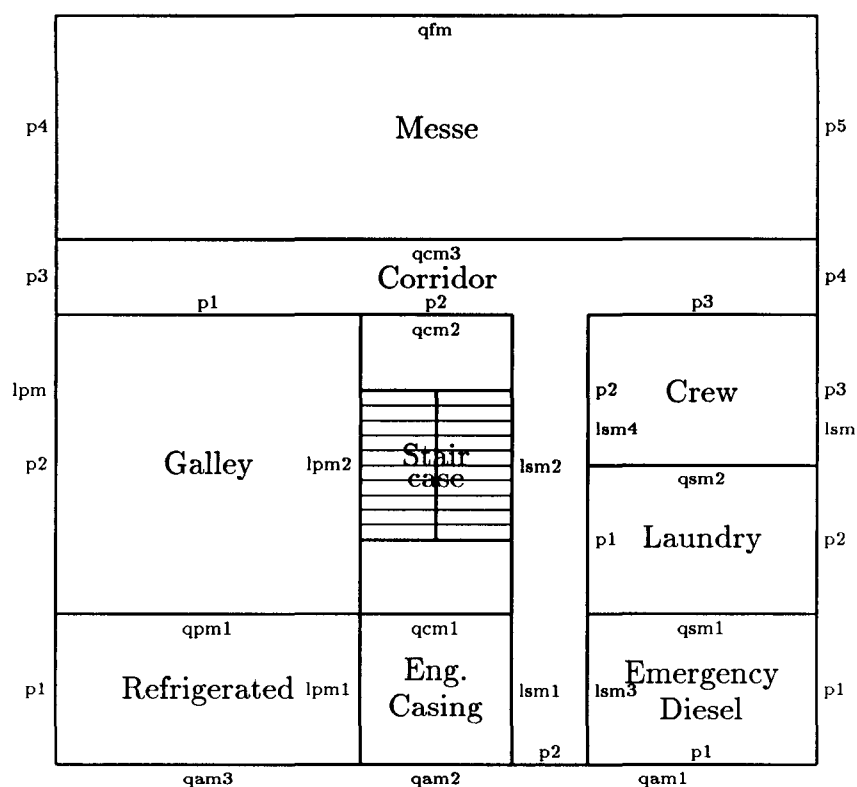


Abbildung 5.4: Deckshaus, 1. Deck

|                     | Textgröße<br>des<br>Regelwerkes | Anzahl<br>Regeln<br>im Text | Regel-<br>Code | Anzahl<br>Regeln im<br>Regelcode | Regel-<br>Struktur-<br>Datenbank |
|---------------------|---------------------------------|-----------------------------|----------------|----------------------------------|----------------------------------|
| Beispiel-Vorschrift | 0.9 kB                          | 5                           | 2.1 kB         | 5                                | 5 kB                             |
| IMO A167            | 6.2 kB                          | 26                          | 5.8 kB         | 25                               | 21 kB                            |
| SOLAS 83 II-2       | 15.3kB                          | 58                          | 19.7 kB        | 56                               | 56 kB                            |

Tabelle 5.1: Ressourcenverbrauch

Regeltext.

Die benötigte Hauptspeichergröße hängt von der *Schachtelungstiefe* der Regeln ab, d.h. wieviel Regeln wieder andere Regeln aufrufen. Je Schachtelung werden 150kB bis 250kB benötigt. Dieser Wert ist abhängig von z.B. der Komplexität der logischen Ausdrücke. Für den Programmcode werden ca. 1.3MB benötigt. Bei komplexen Vorschriften muß man mit einer Schachtelungstiefe von bis zu 15 Regeln rechnen. Damit ergibt sich ein benötigter Hauptspeicher für das Bearbeiten einer Regel von bis zu 5MB.

|  | Textgröße<br>des<br>Regelwerkes | Anzahl<br>Regeln | Regel-<br>Code | Regel-<br>Struktur-<br>Datenbank |
|--|---------------------------------|------------------|----------------|----------------------------------|
| SOLAS Kapitel II-1,<br>Leck-Stabilität<br>Passagierschiffe | 125 kB                          | 500              | 150 kB         | 500 kB                           |
| SOLAS Kapitel II-1B,<br>Leck-Stabilität<br>Trockenfrachter | 14 kB                           | 56               | 17 kB          | 56 kB                            |
| SOLAS Kapitel II-2,<br>Feuersicherheit                     | 200 kB                          | 800              | 240 kB         | 800 kB                           |
| SOLAS Kapitel III,<br>Rettungsmittel                       | 130 kB                          | 520              | 160 kB         | 520 kB                           |
| Marpol 78,<br>Öl-und Produkten Tanker                      | 90 kB                           | 360              | 110 kB         | 360 kB                           |
| Chemikalien Code,<br>Chemikalien Tanker                    | 200 kB                          | 800              | 240 kB         | 800 kB                           |
| SBG,<br>Bekanntmachung vom<br>Oktober 1984,<br>Abschnitt 3 | 8 kB                            | 32               | 10 kB          | 32 kB                            |

Tabelle 5.2: erwarteter Ressourcenverbrauch

## 6 Zusammenfassung

Wie in der Einleitung ausgeführt, werden technische Regelwerke immer umfangreicher und komplexer. Der einzelne Ingenieur ist so kaum noch in der Lage, alle Vorschriften für einen Entwurf sicher zu berücksichtigen. Daher erscheint es sinnvoll, den Ingenieur auch in diesem Bereich durch Rechnerhilfe zu entlasten. Ein allgemein verwendbares Werkzeug hierfür war bisher nicht verfügbar.

Die Analyse existierender Regelwerke zeigte, daß man diese durchgängig in einzelne logische Aussagen wie Regeln, Voraussetzungen, Bedingungen, Vergleiche usw. zerlegen kann.

Die Analyse der Aufgaben eines Ingenieurs im Zusammenhang mit Regelwerken führt zu folgenden vier Fragestellungen:

1. Was ist der Inhalt des Regelwerkes?
2. Ist das Regelwerk für ein gegebenes Objekt relevant?
3. Erfüllt ein gegebenes Objekt das Regelwerk?
4. Welche Werte muß ein Datum annehmen, damit das Objekt das Regelwerk erfüllt?

Für diese Fragestellungen wurde beschrieben, wie die entsprechenden Ergebnisse bestimmt werden können, wenn die Regelwerke in die oben genannten logischen Aussagen zerlegt worden sind.

Es wurde außerdem ein Ansatz beschrieben, mit dem unsichere Daten behandelt werden können und mit dem die daraus resultierenden unsicheren Ergebnisse einer Regel berechnet werden können. Die Berücksichtigung unsicherer Werte ist im Entwurf hilfreich, aber für z.B. Klassifikationsgesellschaften oder Verwaltungen nicht nötig.

Zur Implementation von Regelwerken in Rechnern wurde eine Sprache speziell für Regelwerke definiert. Es wurde darauf geachtet, daß die Syntaxelemente dieser Sprache den typischen Elementen in Regelwerken entsprechen.

Für die Ergebnisdarstellung wurden zwei Modi und unterschiedliche Detaillierungsstufen bereitgestellt. Die Ergebnisse werden vom Rechner der Fragestellung entsprechend aufbereitet.

Die sich aus diesen Ansätzen ergebenden drei Programm-Moduln Compiler, Regelpbearbeitung und Ergebnisdarstellung wurden als unabhängige C-Programme realisiert. Der Datentransfer zwischen den einzelnen Moduln erfolgt über eine Datenbank. Für eine einfache Handhabung wurde eine fensterbasierte Benutzerschnittstelle entwickelt, von der aus auf die drei Moduln zugegriffen werden kann.

Der Zugriff auf Objektdaten wurde über eine hierarchische Datenverwaltung realisiert.

Anhand der Beispiele wurde gezeigt, daß sich die in der Technik, speziell im Schiffbau, existierenden Regelwerke mit den vorgeschlagenen Methoden gut bearbeiten lassen.

Das Übersetzen der Regelwerke in den Regelcode ist durch die direkte Abbildung der Syntaxelemente der Regelwerke auf die Regelsprache möglich, ohne alle inneren Zusammenhänge des Regelwerkes zu kennen. Die Fehleranfälligkeit ist gering. Durch die Darstellung der Ergebnisse in den verschiedenen Modi und Detailierungsstufen, kann der Benutzer schnell zu den für ihn relevanten Aussagen kommen.

Von der erwarteten Größe der Dateien ausgehend (Tabelle 5.2), ist es realistisch, auch komplette Vorschriften-Systeme mit diesen Methoden bearbeitet zu können. Nach [Sans90] wurde geschätzt, daß es 2-3 Mannjahre kosten würde, das Kapitel II-2, Feuersicherheit, aus SOLAS für das Experten-System CARIES aufzubereiten. Durch die hier definierte Sprache für Regeln, die sich an den in Regelwerken vorkommenden Sprachelementen orientiert, sollte diese Übersetzung merklich schneller möglich sein. Trotzdem bleibt das Übersetzen ein erheblicher Aufwand.

Die Rechenzeiten sind auf dem verwendeten Rechner (Atari ST) mit einigen Minuten bis zu zwei Stunden für Dimensionierungsaufgaben zu hoch. Workstations mit heute üblicher Rechenleistung haben z.Zt. eine zehn- bis fünfzigfache Rechenleistungen, so daß die Rechenzeiten auf solchen Maschinen in einer akzeptablen Größenordnung liegen. Da die Programme in ANSI-C geschrieben wurden, ist eine Portierung auf solche Systeme möglich.

Ein wesentliches, aber von dieser Arbeit unabhängiges Problem ist die Bereitstellung der Daten (siehe auch [Top92]). Die für die Beispiele verwendete hierarchische Datenverwaltung reicht für Vorschriften, die z.B. auf topologische Zusammenhänge zurückgreifen, nicht aus (siehe 5.3). Grundsätzlich stellt es kein Problem dar, andere Datenverwaltungen bzw. Datenbanken wie z.B. in [Bron88] beschrieben, zu integrieren. Voraussetzung ist, daß die Syntax, über die man auf einzelne Werte zugreifen kann, eindeutig ist und daß ein definiertes Datenmodell vorliegt. Anstrengungen werden hier z.B. mit STEP [STEP91] und Neutrabas unternommen.

Die Arbeit von Sanusi und Tietgen [Sans90] und auch die Untersuchungen der IMO [IMO91b] haben gezeigt, wie mit Textdatenbanken und -retrievalsystemen der Ingenieur auch bei der Arbeit mit Regelwerken sinnvoll vom Rechner unterstützt werden kann. Das hier vorgestellte System geht über die genannten Systeme hinaus, indem nicht nur die Suche nach den zu berücksichtigenden Regeln unterstützt wird, sondern auch das Überprüfen eines Objektes auf Einhalten eines Regelwerkes und das Dimensionieren an Hand eines Regelwerkes vom Rechner durchgeführt wird.

Voraussetzung für eine breite Anwendung derartiger Hilfsmittel ist das Bereitstellen und Pflegen der Regelwerke für den Rechner. Dies bleibt ein erheblicher Aufwand und sollte von einer zentralen Stelle wie der IMO durchgeführt oder zumindest koordiniert werden.

Alle hier gemachten Ansätze und die Realisierungen sind selbstverständlich in beliebigen anderen technischen Bereichen verwendbar.

## 7 Begriffe, Abkürzungen und Symbole

### 7.1 Verzeichnis der Begriffe und Abkürzungen

|           |   |
|-----------|---|
| ABS       | American Bureau of Shipping   |
| BMV       | Bundes Ministerium für Verkehr                                      |
| CD-ROM    | Compact Disc-Read Only Memory                                       |
| DIN       | Deutsche Industrienorm  |
| DNV       | Det Norske Veritas  |
| GL        | Germanischer Lloyd  |
| IMDG Code | International Maritim Dangerous Goods Code                          |
| IMO       | International Maritime Organisation                                 |
| ISO       | International Standardisation Organisation                          |
| MARPOL    | International Convention for the Prevention of Pollution from Ships |
| OCR       | automatische Schrifterkennung                                       |
| SBG       | See-Berufsgenossenschaft  |
| SchSV     | Verordnung über die Sicherheit der Seeschiffe (SBG)                 |
| SOLAS     | Safety of Live at Sea   |
| Scanner   | Digitalisierer für Texte und Bilder                                 |
| USCG      | United States Coast Guard   |

## 7.2 Symbolverzeichnis

|                          |   |
|--------------------------|---|
| $A(X)$                   | logische Aussage, abhängig von den dem Datenvektor $X$  |
| $a(A)$                   | Funktion, die den Wert 1 annimmt, wenn die Aussage $A$ wahr ist,<br>Zuordnung zur Aussage $A$ |
| $E(O)$                   | zweidimensionale Tabelle von Bedingungen für das Objekt $O$                                   |
| $G$                      | Gewichtung bei zwei Optimierungskriterien   |
| <i>FALSCH</i>            | logische Konstante 'falsch', entspricht einer Wahrscheinlichkeit von 0.0.                     |
| $K(O)$                   | Kriterium für das Objekt $O$  |
| $L(O)$                   | Liste von Kriterien für das Objekt $O$  |
| $M_i$                    | Optimierungskriterium   |
| $N_i$                    | Nebenbedingungen bei der Optimierung  |
| $O$                      | Objekt  |
| $P(A)$                   | Wahrscheinlichkeit, daß die Aussage $A$ wahr ist  |
| $p$                      | $f$ über den Bereich der Werte  |
| $p_{ges}$                | $f$ über den Bereich der Werte  |
| $R(O)$                   | Regel für das Objekt $O$  |
| $T_i$                    | Teilbereich $i$ eines Objekts, Teilobjekt   |
| $V(O)$                   | Voraussetzung für das Objekt $O$  |
| $V_d(O)$                 | Ansonst-Voraussetzung für das Objekt $O$  |
| <i>WAHR</i>              | logische Konstante 'wahr', entspricht einer Wahrscheinlichkeit von 1.0.                       |
| $X =$                    |   |
| $(X_1, X_2, \dots, X_n)$ | Datenvektor $X$ bestehend aus den $n$ Werten $X_1, X_2, \dots, X_n$                           |
| !                        | logische Negation einer Aussage   |
| &                        | logisches UND zweier stochastisch unabhängiger Aussagen                                       |
|                          | logisches ODER zweier stochastisch unabhängiger Aussagen                                      |
|                          | logisches ODER zweier sich gegenseitig ausschließender Aussagen                               |

## 8 Literatur

- [ABS91] *ABS Steel Rules on CD-ROM.*  
American Bureau of Shipping, New Ark 1991
- [Appel85] H.-J. Appelrath.  
*Von Datenbanken zu Expertensystemen, Informatik Fachberichte 102.*  
Springer Verlag Berlin, Heidelberg 1985
- [Bor89] *Turbo C Handbuch.*  
Borland, München 1989
- [Brat87] I. Bratko.  
*Prolog, Programmierung für künstliche Intelligenz.*  
Addison-Wesley Publishing Company, Bonn 1987
- [Bron88] R. Bronsart.  
*Ein Datenmodell für schiffbauliche Stahlstrukturen.*  
Dissertation Uni Hamburg, Hamburg 1988
- [Carnap59] Carnap R. und W. Stegmüller.  
*Induktive Logik und Wahrscheinlichkeit.*  
Springer Verlag, Wien 1959
- [Clock84] W.F. Clocksin u. C.S. Mellish.  
*Programming in Prolog.*  
Springer Verlag, Berlin Heidelberg New York Tokyo 1984
- [DNV91] VEGA,  
*A Databank for Statuory Regulations.*  
Det Norske Veritas, Statuory Services, Oslo 1991
- [Hen78] W. Hengartner u. R. Theodorescu.  
*Einfuehrung in die Monte - Carlo - Methode.*  
Carl Hanser Verlag, Muenchen 1978
- [IMO87] International Maritime Organisation.  
*Intact stability criteria for passenger and cargo ships.*  
IMO, London 1987
- [IMO91a] International Maritime Organisation.  
*MSC 59/WP.8/Add.1.*  
IMO, London 1991
- [IMO91b] International Maritime Organisation.  
*MSC 59/22/3*  
*Computer Database for Applicable Requirements of IMO Instruments*  
*Feasibility study, Note by the Secretariat*  
IMO, London 1991
- [DNV92] *IMO Contract to DNV.*  
Det Norske Veritas, DNV Forum No. 1/92, Oslo 1992
- [Jen90] H. Jensen.  
*JOSEPH, eine hierarchische Daten- und Methodenverwaltung.*  
unveröffentlicht, 1990

- [Jen91] H. Jensen.  
*CW, eine fensterorientierte Benutzerschnittstelle für ASCII-Terminals.*  
unveröffentlicht, 1991
- [John] St.C. Johnson.  
*YACC : Yet Another Compiler Compiler.*  
ULTRIX-32 Supplementary Documents, Vol. II : Programmer, Digital  
Equipment Corp.
- [Ker83] B.W. Kernighan u. D.M. Ritchie.  
*Programmieren in C.*  
Carl Hanser Verlag, Muenchen Wien 1983
- [Krapp73] O. Krappinger.  
*Grundlagen der Wahrscheinlichkeitsrechnung.*  
Institut für Schiffbau, Hamburg 1973.
- [Lesk] M.E. Lesk and E. Schmidt.  
*LEX - A Lexical Analyzer Generator.*  
ULTRIX-32 Supplementary Documents, Vol. II : Programmer, Digital  
Equipment Corp.
- [Now76] H. Nowacki.  
*Einführung in die Methoden der Optimierung.*  
Kontaktstudium, Institut für Schiffbau, Hamburg 1976.
- [Sans90] Y. Sanusi, B. Tietgen.  
*Prototyp eines computergestützten Systems zur Anwendung von Bauvorschriften für Seeschiffe am Beispiel des Kapitels II-2 der SOLAS'74 (Expertensystem Brandschutz).*  
Forschungsbericht FE-Nr.40278/89 des BVM.  
Germanischer Lloyd, Hamburg 1990.
- [Schi87] H. Schildt.  
*Artificial Intelligence using C.*  
Osborne McGraw-Hill, Berkley 1987.
- [Soe83] H. Söding.  
*Beschreibung der Subroutine OPT.*  
Institut für Schiffbau, Hamburg 1983
- [Soe85] H. Söding.  
*Beschreibung der Subroutine DANTZIG.*  
Institut für Schiffbau, Schrift Nr. 2343, Hamburg 1985.
- [SOLAS83] International Maritime Organisation.  
*International Convention for the Safety of Live at Sea, Consolidated Text of the 1974 SOLAS Convention, the 1978 SOLAS Protocol, the 1981 and 1983 SOLAS Amendments.*  
IMO, London 1986
- [STEP91] *STEP – Standard for the Exchange of Product Model Data ISO draft Proposal 10303.*  
International Standardisation Organisation, 1991

- [Top92] I.A. Toparlak.  
*Interpretation von Bemessungsvorschriften am Produktmodell der schiffbaulichen Stahlstruktur.*  
STG Sprechtag Rechnereinsatz in der schiffbaulichen Konstruktion, Hamburg 1992
- [Zieg87] M. Ziegler.  
*COSMOS, Case Oriented Storage Management of Structures.*  
TU Berlin, SFB 203, Berlin 1987



# Anhang



## A Beschreibung existierender Programme

### A.1 Stabilität

Name : SFOLDS  
 Vorschrift : alle IMO Intakt- und Leckstabilitätsvorschriften incl. Getreide Stabilität und den probabilistischen Vorschriften A256 und SOLAS Chapt. II-1 Part B1.  
 Hersteller : BMT CORTEC Ltd., Wallsend, UK  
 Beschreibung : Das Programm verwaltet Ladefälle und berechnet für Intakt- und Leckstabilität die Anfangsstabilität und Hebelarmkurven einschließlich Zwischenzuständen. Berechnung von KG-Grenzkurven.

Name : AutoHYDRO  
 Vorschrift : alle IMO Intakt- und Leckstabilitätsvorschriften incl. SOLAS Chapt. II-1 Part B1.  
 Hersteller : Coastdesign, Vancouver, Kanada  
 Beschreibung : Das Programm verwaltet Ladefälle und berechnet für Intakt- und Leckstabilität die Anfangsstabilität und Hebelarmkurven.

Name : Pilot  
 Vorschrift : IMO A167  
 Hersteller : DNV, Det norske Veritas, Oslo, Norway  
 Beschreibung : Das Programm berechnet Anfangsstabilität und Hebelarmkurven für Intaktstabilität. Berechnung von KG-Grenzkurven.

Name : SEASAFE 1  
 Vorschrift : IMO Intakt- und Leckstabilitätsvorschriften einschl. der probabilistischen Vorschriften  
 Hersteller : Lloyd's Register, London, UK  
 Beschreibung : Mehrere Programme für die verschiedenen Berechnungsaufgaben mit einem Datei-Management-System

Name : NAPA  
 Vorschrift : alle IMO Intakt- und Leckstabilitätsvorschriften incl. Getreide Stabilität und den probabilistischen Vorschriften A256 und SOLAS Chapt. II-1 Part B1. Intaktstabilität: Rahola Kriterien, SBG Richtlinien vom Okt. 1984, USSR-rules. Leckstabilität: USSR-rules  
 Hersteller : NAPA OY, Helsinki, Finnland  
 Beschreibung : Das Programm verwaltet Ladefälle und berechnet für Intakt- und Leckstabilität die Anfangsstabilität und Hebelarmkurven einschließlich Zwischenzuständen. Berechnung von KG-Grenzkurven.

## A.2 Festigkeit

Name : SFOLDS  
 Vorschrift : Längsfestigkeit nach gegebenen Grenzkurven  
 Hersteller : BMT CORTEC Ltd., Wallsend, UK  
 Beschreibung : Das Programm verwaltet Ladefälle und Leerschiffsgewicht und berechnet Querkraft- und Biegemomentenverlauf.

Name : AutoCLASS  
 Vorschrift : Längsfestigkeit nach DNV-Rules, Lokale Festigkeit (Plattenstärken, Profile) nach DNV-Rules  
 Hersteller : Coastdesign, Vancouver, Kanada  
 Beschreibung : Das Programm berechnet Querkraft- und Biegemomentenverlauf sowie erforderliche Plattenstärken, Profile etc. für örtliche Strukturen (Schiffsquerschnitte, I-Träger, Trägerroste, ...) nach DNV-Rules.

Name : Pilot  
 Vorschrift : Längsfestigkeit nach DNV-Rules, Lokale Festigkeit (Plattenstärken, Profile) nach DNV-Rules  
 Hersteller : DNV, Det norske Veritas, Oslo, Norway  
 Beschreibung : Das Programm berechnet Querkraft- und Biegemomentenverlauf sowie erforderliche Plattenstärken, Profile etc. für örtliche Strukturen (Schiffsquerschnitte, I-Träger, Trägerroste, ...) nach DNV-Rules.

Name : LR.PASS  
Vorschrift : LR Rules für den Stahlschiffskörper  
Hersteller : Lloyd's Register, London, UK  
Beschreibung : Das Programm berechnet erforderliche Plattenstärken, Profile etc. für örtliche Strukturen (Schiffsquerschnitte, I-Träger, Trägerroste, ...) nach LR-Rules.

Name : GLRULES  
Vorschrift : GL Vorschriften für den Stahlschiffskörper  
Hersteller : Germanischer Lloyd, Hamburg, Deutschland  
Beschreibung : Das Programm berechnet erforderliche Plattenstärken, Profile etc. für örtliche Strukturen (Schiffsquerschnitte, I-Träger, Trägerroste, ...) nach GL-Vorschriften.

Name : NAPA  
Vorschrift : Längsfestigkeit nach gegebenen Grenzkurven  
Hersteller : NAPA OY, Helsinki, Finnland  
Beschreibung : Das Programm verwaltet Ladefälle und Leerschiffsgewicht und berechnet Querkraft- und Biegemomentenverlauf.

### **A.3 andere Bereiche**

Name : S2PROD16  
Vorschrift : Chemical Code  
Hersteller : Lloyd's Register, London  
Beschreibung : Erstellen der Produktenliste nach IMO Chemical-Code

Name : CHEMSHIP II  
Vorschrift : Chemical- und Gas-Code  
Hersteller : Lloyd's Register, London, UK  
Beschreibung : Erstellen der Produktenliste nach IMO Chemical- und Gas-Code

## B EBNF – Erweiterte Backus-Naur Form

Diese erweiterte Backus–Naur Form (EBNF) ist ein Hilfsmittel um die Syntax von Sprachen, spez. Programmiersprachen, darzustellen. Sie besteht aus einer Anzahl von Regeln und Produktionen, einer Grammatik, mit der man die Form der Sätze dieser Sprache beschreiben kann.

Jede Produktion besteht aus einem nicht-terminalen Symbol und einem EBNF–Ausdruck, die durch ein Gleichheitszeichen getrennt sind und mit einem Punkt abgeschlossen werden. Die nicht-terminalen Symbole sind *Meta-Identifizier* und der EBNF–Ausdruck ist die Definition des *Meta-Identifizier*. Terminale Symbole sind die Syntax–Konstanten der zu beschreibenden Sprache.

Ein EBNF–Ausdruck besteht aus einem oder mehreren terminalen Symbolen, nicht-terminalen Symbolen und anderen *Meta-Symbolen* (siehe Tabelle B.1).

| <i>Meta-Symbol</i>      | Definition   |
|-------------------------|--|
| =                       | wird definiert als                                 |
|                         | entweder oder                                      |
| .                       | Ende der Produktion                                |
| [X]                     | X darf 0 oder 1 mal vorkommen                      |
| {X}                     | X darf 0 oder mehrmals vorkommen                   |
| < X >                   | X kommt einmal oder mehrmals vor                   |
| (X Y)                   | eine Gruppe: entweder X oder Y                     |
| "XYZ"                   | das terminale Symbol XYZ                           |
| <i>MetaIdentifizier</i> | das nicht-terminale Symbol <i>MetaIdentifizier</i> |
| /Text/                  | Text   |

Tabelle B.1: *Meta-Symbole*

Als Beispiel kann man die EBNF mit sich selbst definieren (Tabelle B.2). Jede Syntax hat ein Start–Symbol. In diesem Beispiel ist es *Syntax*.

Diese Beschreibung einer EBNF ist in Anlehnung an Anhang D aus  
 Pascal : User Manual and Report  
 K.Jensen, N.Wirth, B.Andrew, J.F.Miner  
 Springer-Verlag New York 1974.

erstellt worden. Die Meta-Symbole < X > und /Text/ sind neu hinzugekommen. Mit /Text/ kann die Syntax mit Worten beschrieben werden bzw. auf eine bekannte oder an anderer Stelle beschriebene Syntax verwiesen werden. Dies führt an einigen Stellen zu erheblich übersichtlicheren und kürzeren Syntaxbeschreibungen.

|                      |   |  |
|----------------------|---|--|
| <i>Syntax</i>        | = | <i>Produktion</i> .  |
| <i>Produktion</i>    | = | <i>NichtTerminal</i> ”=“ <i>Ausdruck</i> ”.“.  |
| <i>Ausdruck</i>      | = | <i>Term</i> {”  “ <i>Term</i> }.   |
| <i>Term</i>          | = | <i>Faktor</i> { <i>Faktor</i> }.   |
| <i>Faktor</i>        | = | <i>NichtTerminal</i>  <br><i>Terminal</i>  <br>”(“ <i>Ausdruck</i> ”)“  <br>”[“ <i>Ausdruck</i> ”]“  <br>”<“ <i>Ausdruck</i> ”>“  <br>”{“ <i>Ausdruck</i> ”}“. |
| <i>Terminal</i>      | = | ””“ <i>Character</i> { <i>Character</i> } ”““.   |
| <i>NichtTerminal</i> | = | <i>Letter</i> { <i>Letter</i>   <i>Digit</i> }.  |
| <i>Character</i>     | = | / druckbares Zeichen /.  |
| <i>Letter</i>        | = | / Buchstabe /.   |
| <i>Digit</i>         | = | / Ziffer /.  |

Tabelle B.2: EBNF definiert mit EBNF

## C Beispiele

### C.1 Beispiel Vorschrift

#### C.1.1 Regeltext

§1 : Um die Sicherheit von Schiffen zu gewährleisten müssen die in den Paragraphen 2 und 3 geforderten Grenzwerte eingehalten werden.

§2 : Für alle Ladefälle außer 'Leeres Schiff' muss gelten,

wenn  $L_{pp} \geq 100\text{m}$       $a \geq 0.2$  für  $B \cdot L_{pp} \geq 2000$   
                                   $a \geq 0.4$  für  $B \cdot L_{pp} \leq 1000$   
                                  für Zwischenwerte ist linear zu interpolieren

mit  $a = GM \cdot \text{Umfang}^2$  ,  
                                  Umfang im Bogenmaß

andernfalls              $GM \geq 0.5\text{m}$  und Umfang  $\geq 50$  Grad  
                                  oder  $GM \cdot \text{Umfang} \geq 30 \text{ m} \cdot \text{Grad}$

§3 : Für den Ladefall 'Leeres Schiff' muss gelten

$GM \geq 0.2\text{m}$  und Umfang  $\geq 20$  Grad  
 oder  $GM \cdot \text{Umfang} \geq 5 \text{ m} \cdot \text{Grad}$

## C.1.2 Regelcode

```
Beispiel_Vorschrift
e:\uni\promo\zzzdat\ptest
```

```
regel Beispiel.Paragraph_1 fuer ship
\ Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,
  muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte
  eingehalten werden.\
```

```
( dann ( ->^.Paragraph_2 & ->^.Paragraph_3 ) \ Paragraph 2 & 3 \
)
```

```
regel Beispiel.Paragraph_2 fuer ship
\ Paragraph 2 : Fuer alle Ladefaelle
  ausser 'Leeres Schiff' muss gelten \
```

```
( fueralle .lc \fuer alle Ladefaelle \
  wenn ( .log != "light_ship" ) \ nicht leeres Schiff \
  dann ( ->.Kriterium ) \ Kriterium \
)
```

```
regel Beispiel.Paragraph_2.Kriterium fuer ship.lc
\ Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff \
```

```
( wenn ( ^.lpp >= 100. ) \ Lpp >= 100 m \
  dann ( rechnen ( reallokal a ;
    a = .gm * pow ( .range / 57.3, 2 ) ;
    ) \ a = GM * ( Umfang / 57.3 ) ** 2 \

    wenn ( ^.b * ^.lpp >= 2000 ) \ B * Lpp >= 2000 m**2 \
    dann ( a >= 0.2 ) \ a >= 0.2 \

    wenn ( ^.b * ^.lpp <= 1000 ) \ B * Lpp <= 1000 m**2 \
    dann ( a >= 0.4 ) \ a >= 0.4 \

    wenn ( default ) \ 1000 m**2 <= B * Lpp <= 2000 m**2 \
    dann ( rechnen ( reallokal agrenz ;
      agrenz =
        0.4 - 0.2 * ( ^.b * ^.lpp - 1000 ) /
          1000 ;
      ) \ agrenz linear interpolieren \
    dann ( a >= agrenz ) \ a >= agrenz \
  )
)
```

```
wenn ( default ) \ Lpp < 100 m \
dann ( ( .gm >= 0.5 \ GM >= 0.5 m\
      & .range >= 50 \ Umfang >= 50 Grad \ )
      | .gm * .range >= 30 \ GM * Umfang >= 30 mGrad \ )
      \ Gm >= 0.5 m und Umfang >= 50 Grad
      oder Gm * Umfang >= 30 m*Grad \
)
```

```
regel Beispiel.Paragraph_3 fuer ship
\ Paragraph 3 : Fuer den Ladefall 'Leeres Schiff' muss gelten: \
```

```
( fueralle .lc
  wenn ( .log == "light_ship" ) \ leeres Schiff \
  dann ( ->.Kriterium ) \ Kriterium \
)
```

```
regel Beispiel.Paragraph_3.Kriterium fuer ship.lc
\ Stabilitaetskriterium fuer Ladefaelle leeres Schiff \
```

```
( dann ( ( .gm >= 0.2 \ GM >= 0.2 m \
      & .range >= 20 \ Umfang >= 20 Grad \ )
      | .gm * .range >= 5 \ GM * Umfang >= 5 mGrad \ )
      \ Gm >= 0.2 m und Umfang >= 20 Grad
      oder Gm * Umfang >= 5 m*Grad \
)
```

## C.1.3 Datenmodell

Datenmodelldatei : e:\uni\promo\zzzdat\ptest Wed Mar 11 10:22:16 1992

Beginn : file=e:\uni\promo\zzzdat\ptest

- k = startknoten

```
1 1 - ship , k = schiff ( =schiffsnummer)
2 4 - lpp , p = laenge zwischen den loten : m 1.000000
2 3 - b , p = breite auf spanten : m 1.000000
2 2 - lc , k = ladefall ( = name )
3 5 - log , a = logischer name : 3
      light_ship
      ballast
      homogen
3 4 - range , p = hebelarmumfang : grad 1.000000
3 3 - a , p = flaeche unter dem dreieck unter der gm-geraden :
      m*rad 1.000000
3 2 - gm , p = metazentrische hoehe : m 1.000000
3 1 - regel , r = regelergebnis
2 1 - regel , r = regelergebnis
```

## C.1.4 Daten

Wertedatei : e:\uni\promo\zzzdat\ptest Fri Mar 06 14:44:16 1992

Datenmodelldatei : e:\uni\promo\zzzdat\ptest Wed Mar 11 10:22:16 1992

Beginn : file=e:\uni\promo\zzzdat\ptest

```
- k = startknoten

1   - ship = stest
2   - lpp = 98.000000 m , 105.000000 m
2   - b = 16.000000 m , 21.000000 m
2   - lc = c0
3   - regel :
      Beispiel.Paragraph_3.Kriterium = 1.000000
3   - log = light_ship
3   - gm = 0.200000 m , 0.200000 m
3   - range = 30.000000 grad , 40.000000 grad
2   - lc = c1
3   - log = ballast
3   - gm = 0.600000 m , 0.800000 m
3   - range = 60.000000 grad , 65.000000 grad
2   - lc = c2
3   - log = homogen
3   - gm = 0.400000 m , 0.500000 m
3   - range = 35.000000 grad , 45.000000 grad
```

## C.1.5 Prüfen

## C.1.5.1 Prüfen, tabellarische Ausgabe, kurz

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'Beispiel\_Vorschrift'  
folgende Regel erfuehlt ist :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf  | Wert                        |
|-------|---|---|-----------------------------|
| 0     | Beispiel.Paragraph_1<br>Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,<br>muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte<br>eingehalten werden.     | ship=stest  | 0.425                       |
| 1     | Beispiel.Paragraph_2<br>Paragraph 2 : Fuer alle Ladefaelle<br>ausser 'Leeres Schiff' muss gelten<br>-fuer alle Schleife<br>fuer alle Ladefaelle<br>1. Teilbereich<br>2. Teilbereich | ship=stest<br>.lc<br>ship=stest.lc=c1<br>ship=stest.lc=c2 | 0.425<br><br>1.000<br>0.425 |
| 2     | Beispiel.Paragraph_2.Kriterium<br>Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff   | ship=stest.lc=c2  | 0.425                       |

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10

## C.1.5.2 Prüfen, tabellarische Ausgabe, Überblick

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'Beispiel\_Vorschrift'  
folgende Regel erfuehlt ist :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf   | er-<br>gibt |
|-------|---|------------------|-------------|
| 0     | Beispiel.Paragraph_1  | ship=stest       | 0.425       |
|       | Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten, |                  |             |
|       | muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte   |                  |             |
|       | eingehalten werden.   |                  |             |
|       | 0. Kriterium  |                  | 0.425       |
|       | Paragraph 2 & 3   |                  |             |
| 1     | Beispiel.Paragraph_2  | ship=stest       | 0.425       |
|       | Paragraph 2 : Fuer alle Ladefaelle                              |                  |             |
|       | ausser 'Leeres Schiff' muss gelten                              |                  |             |
|       | -fuer alle Schleife   | .lc              |             |
|       | fuer alle Ladefaelle  |                  |             |
|       | 1. Teilbereich  | ship=stest.lc=c1 | 1.000       |
|       | 0. Kriterium  |                  | 1.000       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  | 1.000       |
|       | Kriterium   |                  |             |
|       | 2. Teilbereich  | ship=stest.lc=c2 | 0.425       |
|       | 0. Kriterium  |                  | 0.425       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  | 0.425       |
|       | Kriterium   |                  |             |
| 2     | Beispiel.Paragraph_2.Kriterium                                  | ship=stest.lc=c2 | 0.425       |
|       | Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff       |                  |             |
|       | 0. Kriterium  |                  | 0.425       |
|       | Vorr.   |                  | 0.722       |
|       | Lpp >= 100 m  |                  |             |
|       | Bed.  |                  | 0.589       |

|       |                                   |                  |       |
|-------|-----------------------------------|------------------|-------|
|       | innere Regel 1                    |                  |       |
|       | 1. Kriterium                      |                  | 0.000 |
|       | Vorr.                             |                  | 0.278 |
|       | Lpp < 100 m                       |                  |       |
|       | Bed.                              |                  | 0.000 |
|       | Gm >= 0.5 m und Umfang >= 50 Grad |                  |       |
|       | oder Gm * Umfang >= 30 m*Grad     |                  |       |
| ----- |                                   |                  |       |
|       | innere Regel 1                    | ship=stest.lc=c2 | 0.589 |
|       | 0. Kriterium                      |                  | 0.183 |
|       | Vorr.                             |                  | 0.253 |
|       | B * Lpp >= 2000 m**2              |                  |       |
|       | Bed.                              |                  | 0.722 |
|       | a >= 0.2                          |                  |       |
|       | 2. Kriterium                      |                  | 0.406 |
|       | Vorr.                             |                  | 0.747 |
|       | 1000 m**2 <= B * Lpp <= 2000 m**2 |                  |       |
|       | Bed.                              |                  | 0.543 |
|       | innere Regel 0                    |                  |       |
| ----- |                                   |                  |       |
|       | innere Regel 0                    | ship=stest.lc=c2 | 0.543 |
|       | 0. Kriterium                      |                  | 0.543 |
|       | a >= agrenz                       |                  |       |
| ----- |                                   |                  |       |

## benutzte Werte

| Name                   | Unter Gr. | Ober Gr. |
|------------------------|-----------|----------|
| ship=stest.b           | 16.0000   | 21.0000  |
| ship=stest.lc=c0.gm    | 0.2000    |          |
| ship=stest.lc=c0.log   | 0.0000    |          |
| ship=stest.lc=c0.range | 30.0000   | 40.0000  |
| ship=stest.lc=c1.gm    | 0.6000    | 0.8000   |
| ship=stest.lc=c1.log   | 1.0000    |          |
| ship=stest.lc=c1.range | 60.0000   | 65.0000  |
| ship=stest.lc=c2.gm    | 0.4000    | 0.5000   |
| ship=stest.lc=c2.log   | 2.0000    |          |
| ship=stest.lc=c2.range | 35.0000   | 45.0000  |
| ship=stest.lpp         | 98.0000   | 105.0000 |

## Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10

## C.1.5.3 Prüfen, tabellarische Ausgabe, vollständig

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'Beispiel\_Vorschrift'  
folgende Regel erfuellt ist :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf   | er-<br>gibt |
|-------|---|------------------|-------------|
| 0     | Beispiel.Paragraph_1  | ship=stest       | 0.425       |
|       | Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,<br>muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte<br>eingehalten werden. |                  |             |
|       | 0. Kriterium  |                  | 0.425       |
|       | Paragraph 2 & 3   |                  |             |
| 1     | Beispiel.Paragraph_2  | ship=stest       | 0.425       |
|       | Paragraph 2 : Fuer alle Ladefaelle<br>ausser 'Leeres Schiff' muss gelten  |                  |             |
|       | -fuer alle Schleife   | .lc              |             |
|       | fuer alle Ladefaelle  |                  |             |
|       | 0. Teilbereich  | ship=stest.lc=c0 | 1.000       |
|       | 0. Kriterium  |                  | 0.000       |
|       | Vorr.   |                  | 0.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed. ist nicht anzuwenden   |                  |             |
|       | Regel ist nicht anzuwenden  |                  | 1.000       |
|       | 1. Teilbereich  | ship=stest.lc=c1 | 1.000       |
|       | 0. Kriterium  |                  | 1.000       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  | 1.000       |
|       | Kriterium   |                  |             |
|       | 2. Teilbereich  | ship=stest.lc=c2 | 0.425       |
|       | 0. Kriterium  |                  | 0.425       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  | 0.425       |
|       | Kriterium   |                  |             |

|       |   |                  |       |
|-------|---|------------------|-------|
| 2     | Beispiel.Paragraph_2.Kriterium                            | ship=stest.lc=c1 | 1.000 |
|       | Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff |                  |       |
|       | 0. Kriterium  |                  | 0.722 |
|       | Vorr.   |                  | 0.722 |
|       | Lpp >= 100 m  |                  |       |
|       | Bed.  |                  | 1.000 |
|       | innere Regel 1  |                  |       |
|       | 1. Kriterium  |                  | 0.278 |
|       | Vorr.   |                  | 0.278 |
|       | Lpp < 100 m   |                  |       |
|       | Bed.  |                  | 1.000 |
|       | Gm >= 0.5 m und Umfang >= 50 Grad                         |                  |       |
|       | oder Gm * Umfang >= 30 m*Grad                             |                  |       |
| ----- |   |                  |       |
|       | innere Regel 1  | ship=stest.lc=c1 | 1.000 |
|       | 0. Kriterium  |                  | 0.253 |
|       | Vorr.   |                  | 0.253 |
|       | B * Lpp >= 2000 m**2                                      |                  |       |
|       | Bed.  |                  | 1.000 |
|       | a >= 0.2  |                  |       |
|       | 1. Kriterium  |                  | 0.000 |
|       | Vorr.   |                  | 0.000 |
|       | B * Lpp <= 1000 m**2                                      |                  |       |
|       | Bed. ist nicht anzuwenden                                 |                  |       |
|       | 2. Kriterium  |                  | 0.747 |
|       | Vorr.   |                  | 0.747 |
|       | 1000 m**2 <= B * Lpp <= 2000 m**2                         |                  |       |
|       | Bed.  |                  | 1.000 |
|       | innere Regel 0  |                  |       |
| ----- |   |                  |       |
|       | innere Regel 0  | ship=stest.lc=c1 | 1.000 |
|       | 0. Kriterium  |                  | 1.000 |
|       | a >= agrenz   |                  |       |
| ----- |   |                  |       |
| 2     | Beispiel.Paragraph_2.Kriterium                            | ship=stest.lc=c2 | 0.425 |
|       | Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff |                  |       |
|       | 0. Kriterium  |                  | 0.425 |
|       | Vorr.   |                  | 0.722 |
|       | Lpp >= 100 m  |                  |       |
|       | Bed.  |                  | 0.589 |
|       | innere Regel 1  |                  |       |
|       | 1. Kriterium  |                  | 0.000 |
|       | Vorr.   |                  | 0.278 |
|       | Lpp < 100 m   |                  |       |
|       | Bed.  |                  | 0.000 |
|       | Gm >= 0.5 m und Umfang >= 50 Grad                         |                  |       |
|       | oder Gm * Umfang >= 30 m*Grad                             |                  |       |
| ----- |   |                  |       |

|  |                  |       |
|--|------------------|-------|
| innere Regel 1   | ship=stest.lc=c2 | 0.589 |
| 0. Kriterium   |                  | 0.183 |
| Vorr.  |                  | 0.253 |
| B * Lpp >= 2000 m**2   |                  |       |
| Bed.   |                  | 0.722 |
| a >= 0.2   |                  |       |
| 1. Kriterium   |                  | 0.000 |
| Vorr.  |                  | 0.000 |
| B * Lpp <= 1000 m**2   |                  |       |
| Bed. ist nicht anzuwenden                                    |                  |       |
| 2. Kriterium   |                  | 0.406 |
| Vorr.  |                  | 0.747 |
| 1000 m**2 <= B * Lpp <= 2000 m**2                            |                  |       |
| Bed.   |                  | 0.543 |
| innere Regel 0   |                  |       |
| -----  |                  |       |
| innere Regel 0   | ship=stest.lc=c2 | 0.543 |
| 0. Kriterium   |                  | 0.543 |
| a >= agrenz  |                  |       |
| -----  |                  |       |
| 1 Beispiel.Paragraph_3                                       | ship=stest       | 1.000 |
| Paragraph 3 : Fuer den Ladefall 'Leeres Schiff' muss gelten: |                  |       |
| -fuer alle Schleife  | .lc              |       |
| 0. Teilbereich   | ship=stest.lc=c0 | 1.000 |
| 0. Kriterium   |                  | 1.000 |
| Vorr.  |                  | 1.000 |
| leeres Schiff  |                  |       |
| Bed.   |                  | 1.000 |
| Kriterium  |                  |       |
| 1. Teilbereich   | ship=stest.lc=c1 | 1.000 |
| 0. Kriterium   |                  | 0.000 |
| Vorr.  |                  | 0.000 |
| leeres Schiff  |                  |       |
| Bed. ist nicht anzuwenden                                    |                  |       |
| Regel ist nicht anzuwenden                                   |                  | 1.000 |
| 2. Teilbereich   | ship=stest.lc=c2 | 1.000 |
| 0. Kriterium   |                  | 0.000 |
| Vorr.  |                  | 0.000 |
| leeres Schiff  |                  |       |
| Bed. ist nicht anzuwenden                                    |                  |       |
| Regel ist nicht anzuwenden                                   |                  | 1.000 |
| -----  |                  |       |
| 2 Beispiel.Paragraph_3.Kriterium                             | ship=stest.lc=c0 | 1.000 |
| Stabilitaetskriterium fuer Ladefaelle leeres Schiff          |                  |       |
| 0. Kriterium   |                  | 1.000 |
| Gm >= 0.2 m und Umfang >= 20 Grad                            |                  |       |
| oder Gm * Umfang >= 5 m*Grad                                 |                  |       |
| -----  |                  |       |

benutzte Werte

-----

| Name                   | Unter Gr. | Ober Gr. |
|------------------------|-----------|----------|
| ship=stest.b           | 16.0000   | 21.0000  |
| ship=stest.lc=c0.gm    |           | 0.2000   |
| ship=stest.lc=c0.log   |           | 0.0000   |
| ship=stest.lc=c0.range | 30.0000   | 40.0000  |
| ship=stest.lc=c1.gm    | 0.6000    | 0.8000   |
| ship=stest.lc=c1.log   |           | 1.0000   |
| ship=stest.lc=c1.range | 60.0000   | 65.0000  |
| ship=stest.lc=c2.gm    | 0.4000    | 0.5000   |
| ship=stest.lc=c2.log   |           | 2.0000   |
| ship=stest.lc=c2.range | 35.0000   | 45.0000  |
| ship=stest.lpp         | 98.0000   | 105.0000 |

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet

Integration mittels Trapezregel

Anzahl Stützstellen : 10

#### C.1.5.4 Prüfen einer aufgerufenen Regel, chronologische Ausgabe, vollständig

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'Beispiel\_Vorschrift'  
folgende Regel erfuehlt ist :

```
Regel 'Beispiel.Paragraph_2.Kriterium'
  fuer 'ship=stest.lc=c2' aus e:\uni\promo\zzzdat\ptest
  _ Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff
mit
V = 0.722 : _ Lpp >= 100 m
und
(
. Rechnen :
. _ a = GM * ( Umfang / 57.3 ) ** 2
. . a = [ 0.149241, 0.308380 ]
. mit
. V = 0.253 : _ B * Lpp >= 2000 m**2
. und
. B = 0.722 : _ a >= 0.2
. ergibt K = 0.183 fuer Kriterium 0
. und mit
. V = 0.000 : _ B * Lpp <= 1000 m**2
. ist die Bedingung nicht anzuwenden
. ergibt K = 0.000 fuer Kriterium 1
. sonst mit V = 0.747 : _ 1000 m**2 <= B * Lpp <= 2000 m**2
. und
. (
. . Rechnen :
. . _ agrenz linear interpolieren
. . . agrenz = [ 0.159000, 0.286400 ]
. . mit V = 1.000
. . und
. . B = 0.543 : _ a >= agrenz
. . ergibt K = 0.543 fuer Kriterium 0
. . ergibt P = 0.543 : interne Regel
. )
. B = 0.543 : _ innere Regel 0
. ergibt K = 0.406 fuer Kriterium 2
. ergibt P = 0.589 : interne Regel
)
B = 0.589 : _ innere Regel 1
ergibt K = 0.425 fuer Kriterium 0
```

```
sonst mit V = 0.278 : _ Lpp < 100 m
und
(
. _ GM >= 0.5 m
. ergibt P = 0.056
. &
. _ Umfang >= 50 Grad
. ergibt P = 0.000
. |
. _ GM * Umfang >= 30 mGrad
. ergibt P = 0.000
)
B = 0.000 : _ Gm >= 0.5 m und Umfang >= 50 Grad
           _                oder Gm * Umfang >= 30 m*Grad
ergibt K = 0.000 fuer Kriterium 1
ergibt P = 0.425 : Regel 'Beispiel.Paragraph_2.Kriterium'
                  fuer 'ship=stest.lc=c2'
```

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10

## C.1.6 Ausgabe Dimensionieren, tabellarisch, kurz

```
+-----+
| Dimensionieren |
+-----+
```

Wahrscheinlichkeit, dass aus  
'Beispiel\_Vorschrift'  
folgende Regel erfuehlt ist :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf | er-<br>gibt |
|-------|---|----------------|-------------|
| 0     | Beispiel.Paragraph_1  | ship=stest     | 1.000       |
|       | Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,<br>muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte<br>eingehalten werden. |                |             |

## Dimensionierung

```
-----
Optimierungsmethode : Zufallssuche mit Nebenbedingung N1
Optimierungskriterium : Abstand der Grenzen
                      = 0.44126
```

| Name                   | Unter Gr. | Ober Gr. |
|------------------------|-----------|----------|
| ship=stest.lc=c2.gm    | 0.5004    | 0.8000   |
| ship=stest.lc=c2.range | 50.8380   | 70.0000  |

## Schalter

```
-----
Unbekannte wurden erfragt bzw. ueber die DB berechnet
Integration mittels Trapezregel
Anzahl Stuetzstellen : 10
```

## C.1.7 Ausgabe Inhalt, chronologische Ausgabe

```
+-----+
| Inhalt |
+-----+
```

Folgende Kriterien sind in  
'Beispiel\_Vorschrift'  
zu erfuellen :

Regel 'Beispiel.Paragraph\_1'

```
_ Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,
_           muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte
_           eingehalten werden.
```

muss

(

```
. Regel 'Beispiel.Paragraph_2'
```

```
. _ Paragraph 2 : Fuer alle Ladefaelle
```

```
. _           ausser 'Leeres Schiff' muss gelten
```

```
. (
```

```
. . ( fuer alle Teilbereiche '.lc'
```

```
. . . _ fuer alle Ladefaelle
```

```
. . . mit
```

```
. . . : _ nicht leeres Schiff
```

```
. . . muss
```

```
. . . (
```

```
. . . . Regel 'Beispiel.Paragraph_2.Kriterium'
```

```
. . . . _ Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff
```

```
. . . . mit
```

```
. . . . : _ Lpp >= 100 m
```

```
. . . . muss
```

```
. . . . (
```

```
. . . . . Rechnen :
```

```
. . . . . _ a = GM * ( Umfang / 57.3 ) ** 2
```

```
. . . . . mit
```

```
. . . . . : _ B * Lpp >= 2000 m**2
```

```
. . . . . muss
```

```
. . . . . : _ a >= 0.2
```

```
. . . . . und mit
```

```
. . . . . : _ B * Lpp <= 1000 m**2
```

```
. . . . . muss
```

```
. . . . . : _ a >= 0.4
```

```
. . . . . sonst : _ 1000 m**2 <= B * Lpp <= 2000 m**2
```

```
. . . . . muss
```

```
. . . . . (
```

```
. . . . . . Rechnen :
```

```
. . . . . . _ agrenz linear interpolieren
```

```
. . . . . . muss
```

```

. . . . . : _ a >= agrenz
. . . . . )
. . . . . : _ innere Regel 0
. . . . . )
. . . . . : _ innere Regel 1
. . . . . sonst : _ Lpp < 100 m
. . . . . muss
. . . . . (
. . . . . _ GM >= 0.5 m
. . . . . &
. . . . . _ Umfang >= 50 Grad
. . . . . |
. . . . . _ GM * Umfang >= 30 mGrad
. . . . . )
. . . . . : _ Gm >= 0.5 m und Umfang >= 50 Grad
. . . . . - oder Gm * Umfang >= 30 m*Grad
. . . . . )
. . . . . : _ Kriterium
. . . . . )
. . . . . )
. . . . . &
. . . . . Regel 'Beispiel.Paragraph_3'
. . . . . _ Paragraph 3 : Fuer den Ladefall 'Leeres Schiff' muss gelten:
. . . . . (
. . . . . ( fuer alle Teilbereiche '.lc'
. . . . . mit
. . . . . : _ leeres Schiff
. . . . . muss
. . . . . (
. . . . . Regel 'Beispiel.Paragraph_3.Kriterium'
. . . . . _ Stabilitaetskriterium fuer Ladefaelle leeres Schiff
. . . . . muss
. . . . . (
. . . . . _ GM >= 0.2 m
. . . . . &
. . . . . _ Umfang >= 20 Grad
. . . . . |
. . . . . _ GM * Umfang >= 5 mGrad
. . . . . )
. . . . . : _ Gm >= 0.2 m und Umfang >= 20 Grad
. . . . . - oder Gm * Umfang >= 5 m*Grad
. . . . . )
. . . . . : _ Kriterium
. . . . . )
. . . . . )
. . . . . )
. . . . . : _ Paragraph 2 & 3

```

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet

Integration mittels Trapezregel

Anzahl Stützstellen : 10

## C.1.8 Ausgabe Relevanz

## C.1.8.1 Relevanz, tabellarische Ausgabe, Überblick

```
+-----+
| Relevanz |
+-----+
```

Folgende Kriterien sind in  
'Beispiel\_Vorschrift'  
zu erfuellen :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf  | er- gibt                             |
|-------|---|---|--------------------------------------|
| 0     | Beispiel.Paragraph_1<br>Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten,<br>muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte<br>eingehalten werden.                       | ship=stest  | 1.000                                |
| 1     | Beispiel.Paragraph_2<br>Paragraph 2 : Fuer alle Ladefaelle<br>ausser 'Leeres Schiff' muss gelten<br>-fuer alle Schleife<br>fuer alle Ladefaelle<br>0. Teilbereich<br>1. Teilbereich<br>2. Teilbereich | ship=stest<br>.lc<br>ship=stest.lc=c0<br>ship=stest.lc=c1<br>ship=stest.lc=c2 | 1.000<br><br>0.000<br>1.000<br>1.000 |
| 2     | Beispiel.Paragraph_2.Kriterium<br>Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff   | ship=stest.lc=c1  | 1.000                                |
| 2     | Beispiel.Paragraph_2.Kriterium<br>Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff   | ship=stest.lc=c2  | 1.000                                |
| 1     | Beispiel.Paragraph_3<br>Paragraph 3 : Fuer den Ladefall 'Leeres Schiff' muss gelten:<br>-fuer alle Schleife<br>0. Teilbereich<br>1. Teilbereich<br>2. Teilbereich                                     | ship=stest<br>.lc<br>ship=stest.lc=c0<br>ship=stest.lc=c1<br>ship=stest.lc=c2 | 1.000<br><br>1.000<br>0.000<br>0.000 |
| 2     | Beispiel.Paragraph_3.Kriterium<br>Stabilitaetskriterium fuer Ladefaelle leeres Schiff   | ship=stest.lc=c0  | 1.000                                |

-----  
Schalter  
-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10  
'rechnen' wurde ausgeführt

## C.1.8.2 Relevanz, tabellarische Ausgabe, vollständig

```
+-----+
| Relevanz |
+-----+
```

Folgende Kriterien sind in  
'Beispiel\_Vorschrift'  
zu erfuellen :

Regel 'Beispiel.Paragraph\_1'  
fuer 'ship=stest' aus e:\uni\promo\zzzdat\ptest

| Ebene | Regel   | angewendet auf   | er-<br>gibt |
|-------|---|------------------|-------------|
| 0     | Beispiel.Paragraph_1  | ship=stest       | 1.000       |
|       | Paragraph 1 : Um die Sicherheit von Schiffen zu gewaehrleisten, |                  |             |
|       | muessen die in den Paragraphen 2 und 3 geforderten Grenzwerte   |                  |             |
|       | eingehalten werden.   |                  |             |
|       | 0. Kriterium  |                  | 1.000       |
|       | Paragraph 2 & 3   |                  |             |
| 1     | Beispiel.Paragraph_2  | ship=stest       | 1.000       |
|       | Paragraph 2 : Fuer alle Ladefaelle                              |                  |             |
|       | ausser 'Leeres Schiff' muss gelten                              |                  |             |
|       | -fuer alle Schleife   | .lc              |             |
|       | fuer alle Ladefaelle  |                  |             |
|       | 0. Teilbereich  | ship=stest.lc=c0 | 0.000       |
|       | 0. Kriterium  |                  | 0.000       |
|       | Vorr.   |                  | 0.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  |             |
|       | 1. Teilbereich  | ship=stest.lc=c1 | 1.000       |
|       | 0. Kriterium  |                  | 1.000       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  |             |
|       | Kriterium   |                  |             |
|       | 2. Teilbereich  | ship=stest.lc=c2 | 1.000       |
|       | 0. Kriterium  |                  | 1.000       |
|       | Vorr.   |                  | 1.000       |
|       | nicht leeres Schiff   |                  |             |
|       | Bed.  |                  |             |
|       | Kriterium   |                  |             |
| 2     | Beispiel.Paragraph_2.Kriterium                                  | ship=stest.lc=c1 | 1.000       |

|   |                  |       |
|---|------------------|-------|
| Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff |                  |       |
| 0. Kriterium  |                  | 0.722 |
| Vorr.   |                  | 0.722 |
| Lpp >= 100 m  |                  |       |
| Bed.  |                  |       |
| innere Regel 1  |                  |       |
| 1. Kriterium  |                  | 0.278 |
| Vorr.   |                  | 0.278 |
| Lpp < 100 m   |                  |       |
| Bed.  |                  |       |
| Gm >= 0.5 m und Umfang >= 50 Grad                         |                  |       |
| oder Gm * Umfang >= 30 m*Grad                             |                  |       |
| -----   |                  |       |
| innere Regel 1  | ship=stest.lc=c1 | 1.000 |
| 0. Kriterium  |                  | 0.253 |
| Vorr.   |                  | 0.253 |
| B * Lpp >= 2000 m**2                                      |                  |       |
| Bed.  |                  |       |
| a >= 0.2  |                  |       |
| 1. Kriterium  |                  | 0.000 |
| Vorr.   |                  | 0.000 |
| B * Lpp <= 1000 m**2                                      |                  |       |
| Bed.  |                  |       |
| 2. Kriterium  |                  | 0.747 |
| Vorr.   |                  | 0.747 |
| 1000 m**2 <= B * Lpp <= 2000 m**2                         |                  |       |
| Bed.  |                  |       |
| innere Regel 0  |                  |       |
| -----   |                  |       |
| innere Regel 0  | ship=stest.lc=c1 | 1.000 |
| 0. Kriterium  |                  | 1.000 |
| a >= agrenz   |                  |       |
| -----   |                  |       |
| 2 Beispiel.Paragraph_2.Kriterium                          | ship=stest.lc=c2 | 1.000 |
| Stabilitaetskriterium fuer Ladefaelle nicht leeres Schiff |                  |       |
| 0. Kriterium  |                  | 0.722 |
| Vorr.   |                  | 0.722 |
| Lpp >= 100 m  |                  |       |
| Bed.  |                  |       |
| innere Regel 1  |                  |       |
| 1. Kriterium  |                  | 0.278 |
| Vorr.   |                  | 0.278 |
| Lpp < 100 m   |                  |       |
| Bed.  |                  |       |
| Gm >= 0.5 m und Umfang >= 50 Grad                         |                  |       |
| oder Gm * Umfang >= 30 m*Grad                             |                  |       |
| -----   |                  |       |
| innere Regel 1  | ship=stest.lc=c2 | 1.000 |

|       |  |                  |       |
|-------|--|------------------|-------|
|       | 0. Kriterium   |                  | 0.253 |
|       | Vorr.  |                  | 0.253 |
|       | B * Lpp >= 2000 m**2   |                  |       |
|       | Bed.   |                  |       |
|       | a >= 0.2   |                  |       |
|       | 1. Kriterium   |                  | 0.000 |
|       | Vorr.  |                  | 0.000 |
|       | B * Lpp <= 1000 m**2   |                  |       |
|       | Bed.   |                  |       |
|       | 2. Kriterium   |                  | 0.747 |
|       | Vorr.  |                  | 0.747 |
|       | 1000 m**2 <= B * Lpp <= 2000 m**2                            |                  |       |
|       | Bed.   |                  |       |
|       | innere Regel 0   |                  |       |
| ----- |  |                  |       |
|       | innere Regel 0   | ship=stest.lc=c2 | 1.000 |
|       | 0. Kriterium   |                  | 1.000 |
|       | a >= agrenz  |                  |       |
| ----- |  |                  |       |
| 1     | Beispiel.Paragraph_3   | ship=stest       | 1.000 |
|       | Paragraph 3 : Fuer den Ladefall 'Leeres Schiff' muss gelten: |                  |       |
|       | -fuer alle Schleife  | .lc              |       |
|       | 0. Teilbereich   | ship=stest.lc=c0 | 1.000 |
|       | 0. Kriterium   |                  | 1.000 |
|       | Vorr.  |                  | 1.000 |
|       | leeres Schiff  |                  |       |
|       | Bed.   |                  |       |
|       | Kriterium  |                  |       |
|       | 1. Teilbereich   | ship=stest.lc=c1 | 0.000 |
|       | 0. Kriterium   |                  | 0.000 |
|       | Vorr.  |                  | 0.000 |
|       | leeres Schiff  |                  |       |
|       | Bed.   |                  |       |
|       | 2. Teilbereich   | ship=stest.lc=c2 | 0.000 |
|       | 0. Kriterium   |                  | 0.000 |
|       | Vorr.  |                  | 0.000 |
|       | leeres Schiff  |                  |       |
|       | Bed.   |                  |       |
| ----- |  |                  |       |
| 2     | Beispiel.Paragraph_3.Kriterium                               | ship=stest.lc=c0 | 1.000 |
|       | Stabilitaetskriterium fuer Ladefaelle leeres Schiff          |                  |       |
|       | 0. Kriterium   |                  | 1.000 |
|       | Gm >= 0.2 m und Umfang >= 20 Grad                            |                  |       |
|       | oder Gm * Umfang >= 5 m*Grad                                 |                  |       |
| ----- |  |                  |       |

benutzte Werte

-----

| Name                   |  | Unter Gr. |  | Ober Gr. |  |
|------------------------|--|-----------|--|----------|--|
| ship=stest.b           |  | 16.0000   |  | 21.0000  |  |
| ship=stest.lc=c0.log   |  |           |  | 0.0000   |  |
| ship=stest.lc=c1.gm    |  | 0.6000    |  | 0.8000   |  |
| ship=stest.lc=c1.log   |  |           |  | 1.0000   |  |
| ship=stest.lc=c1.range |  | 60.0000   |  | 65.0000  |  |
|                        |  |           |  |          |  |
| ship=stest.lc=c2.gm    |  | 0.4000    |  | 0.5000   |  |
| ship=stest.lc=c2.log   |  |           |  | 2.0000   |  |
| ship=stest.lc=c2.range |  | 35.0000   |  | 45.0000  |  |
| ship=stest.lpp         |  | 98.0000   |  | 105.0000 |  |

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel

Anzahl Stützstellen : 10

'rechnen' wurde ausgeführt

## C.2 IMO Resolution A167

### C.2.1 Regelcode

IMO\_A167

e:\uni\promo\zzzdat\ship

```
regel IMO167 fuer ship
\ IMO Resolution A167, Amended A206
Recommendation on intact stability for passenger
and cargo ships under 100 meters in length
as amended with respect to ships carrying deck cargoes\
```

```
( dann ( ->.x1 )
)
```

```
regel IMO167.x1 fuer ship
\ 1. Scope \
```

```
( dann ( ->.x1 )
)
```

```
regel IMO167.x1.x1 fuer ship
\ 1.1 The provision given hereunder are recommended
for new decked sea-going passenger and cargo ships
(other than fishing vessel) under 100 meters in length.\
```

```
( wenn ( .main=main.typ <= "TankShip" \ check type of ship \
        & .main=main.lfb <= 100 \ freeboard length <= 100 meters \ )
  dann ( ->^.x2 & ->^^.x2to4 & ->^^.x6 & ->^^.x7 )
)
```

```
regel IMO167.x1.x2 fuer ship
\ 1.2 Administrations are invited to adopt,
for all conditions of loading, the stability
criteria given in paragraph 5 unless they are
satisfied that operating experience justifies
departures therefrom \
```

```
( fueralle .stab=stab.lc
```

```
wenn ( .group >= "ballast" \ not for light ship and docking conditions \ )
dann ( ->^^.x5 )
```

)

```
regel IMO167.x2to4 fuer ship
\ 2. General Precautions
3. Calculation of Stability Curves
4. Assessment of Compliance with Criteria
paragraphs 2, 3, 4 are fulfilled \
```

```
( frage \ Compliance with Information, Calculation and Assumptions
to Paragraphs 2,3,4 ? \
)
```

```
regel IMO167.x5 fuer ship.stab.lc
\ 5. Recommended Criteria \
```

```
( wenn ( ^^main=main.typ == "PassengerShip" ) \ Passenger Ship \
dann ( ->.x1 ( .res=intact.stab=stab ) & ->.x3 ( .res=intact.stab=stab )
& ->.x4to6 )
```

```
wenn ( default ) \ not a Passenger Ship \
dann ( wenn ( .deckcargo == "timber" ) \ cargo ship with timber deck cargo \
dann ( ->.x2 ( .res=intact.stab=stab )
& ->.x4to6 & ->.x7 )
wenn ( default ) \ cargo ship \
dann ( ->.x1 ( .res=intact.stab=stab )
& ->.x4to6 & ->.x7 )
```

```
)
)
```

```
regel IMO167.x5.x1 fuer ship.stab.lc.res.stab
\ 5.1. The following criteria are recommended
for passenger and cargo ships \
```

```
( dann ( ->.a & ->.b & ->.c & ->.d )
)
```

```
regel IMO167.x5.x1.a fuer ship.stab.lc.res.stab
\ 5.1 (a) The area under the righting lever curve
(GZ curve) should not be less than the following values:
up to 30 deg >= 0.055 m*rad,
up to 40 deg or angle of progr. flooding >= 0.09 m*rad
between 30 and 40 deg >= 0.03 m*rad\
```

```
( wenn ( .open=prog.Php >= 40 / 57.3 )
```

```

        \ angle of progressive flooding > 40 deg \
dann ( .A30p >= 0.055 \ area up to 30 deg >= 0.055 m*rad \
      & .A40p >= 0.09 \ area up to 40 deg >= 0.09 m*rad \
      & .A40p - .A30p >= 0.03 \ area between 30 and 40 deg >= 0.03 m*rad \
    )

wenn ( default ) \ angle of progressive flooding < 40 deg \
dann ( .A30p >= 0.055 \ area up to 30 deg >= 0.055 m*rad \
      & .open=prog.Ap >= 0.09
      \ area up to angle of progressive flooding >= 0.09 m*rad \
      & .open=prog.Ap - .A30p >= 0.03
      \ area between angle of progressive flooding and 30 deg
      >= 0.03 m*rad \
    )
)

```

```

regel IMO167.x5.x1.b fuer ship.stab.lc.res.stab
\ 5.1 (b) The righting lever GZ should be at least 0.20 m
at an angle of heel equal to or greater than 30 deg \

```

```

( wenn ( .Phmaxp >= 30. / 57.3 ) \ angle of heel at max. GZ >= 30 deg \
  dann ( .hmaxp >= 0.20 ) \ max righting lever >= 0.20 m \

  wenn ( default ) \ angle of heel at max. GZ < 30 deg \
  dann ( .GZ30p >= 0.20 ) \ GZ at 30 deg >= 0.20 m \
)

```

```

regel IMO167.x5.x1.c fuer ship.stab.lc.res.stab
\ 5.1 (c) The maximum righting arm should occur at
an angle of heel preferably exceeding 30 deg
but not less than 25 deg. \

```

```

( dann ( .Phmaxp >= 25. / 57.3 )
  \ max righting lever at an angle of heel >= 25 deg \
)

```

```

regel IMO167.x5.x1.d fuer ship.stab.lc.res.stab
\ 5.1 (d) The initial metacentric height GMO should not
be less than 0.15 . \

```

```

( dann ( .GMO >= 0.15 ) \ GMO >= 0.15 m \
)

```

```
regel IMO167.x5.x2 fuer ship.stab.lc.res.stab
\ 5.2. For ships loaded with timber deck cargoes,
the administration may apply the following criteria
in substitution for criteria given in 5.1 above\
```

```
( wenn ( ->.assu )
  dann ( ->.a & ->.b & ->.c ) \ timber deck cargo criteria \

  wenn ( default )
  dann ( ->^.x1 ) \ standard criteria \
)
```

```
regel IMO167.x5.x2.assu fuer ship.stab.lc.res.stab
\ Assumptions for timber deck criteria:
The timber deck cargo extends longitudinally between superstructures
(Where there is no limiting superstructure at the after end, the cargo shall
extend at least to the after end of the aftermost hatchway)
transversly for the full beam of ship after due allowance
for a rounded gunwale not exceeding 4% of the breadth of the ship
and/or securing the supporting uprights and which remains securely
fixed at large angle of heel
```

```
\
( frage
  \ Are the assumptions to use the timberdeckcargo criteria fullfiled ?\
)
```

```
regel IMO167.x5.x2.a fuer ship.stab.lc.res.stab
\ 5.2 (a) The area under the righting lever curve
(GZ curve) should not be less than 0.08 m*rad
up to 40 deg or the angle of flooding, if this
angle is less than 40 deg.\
```

```
( wenn ( .open=prog.Php >= 40 / 57.3 )
  \ angle of progressive flooding > 40 deg \
  dann ( .A40p >= 0.08 ) \ area up to 40 deg >= 0.09 m*rad \

  wenn ( default ) \ angle of progressive flooding < 40 deg \
  dann ( .open=prog.Ap >= 0.08 )
    \ area up to angle of progressive flooding >= 0.09 m*rad \
)
```

```
regel IMO167.x5.x2.b fuer ship.stab.lc.res.stab
\ 5.1 (b) The amximum value of the righting lever (GZ)
should be at least 0.25 m. \
```

```
( dann ( .hmaxp >= 0.25 ) \ max righting lever >= 0.25 m \
)
```

```
regel IMO167.x5.x2.c fuer ship.stab.lc.res.stab
\ 5.1 (c) At all times during a voyage the metacentric height
GMO should be positive after the correction for the free surface
effects of liquid in tanks and, where appropriate,
the absorption of water by the deck cargo and/or ice accretion
on the exposed surfaces. Additionally, in the departure condition
the metacentric height should be not less than 0.10m. \
```

```
( wenn ( ^^deparrr == "departure" ) \ Departure Condition \
  dann ( .GMO >= 0.10 ) \ GMO >= 0.10 m \

  wenn ( default )
  dann ( .GM > 0 ) \ positive corrected GM \
)
```

```
regel IMO167.x5.x3 fuer ship.stab.lc.res.stab
\ 5.3. The following criteria are recommended
for passenger ships \
```

```
( dann ( ->.a & ->.b )
)
```

```
regel IMO167.x5.x3.a fuer ship.stab.lc.res.stab
\ 5.3 (a) The angle of heel on account of crowding passengers
to one side as defined in Appendix II 2.(9) should
not exceed 10 deg. \
```

```
( dann ( ^^depl=depl.passmom / ^^depl=depl.mass <= .GZ10p )
  \ passenger moment / depl. <= leverarm of 10 deg \
)
```

```
regel IMO167.x5.x3.b fuer ship.stab.lc.res.stab
\ 5.3 (b) The angle of heel on account of turning should
not exceed 10 deg when calculated using the
formula:
```

$$MR = 0.02 * v0 **2 / lwl * depl * ( KG - draught / 2 ) \$$

```
( rechnen ( realloc mr ;
  mr = 0.02 * ^^main=main.vservice * ^^main=main.vservice
  / ^^main=main.lwl * ^^depl=depl.mass
```

```

        * ( ^^ .depl=depl.zgcorr - ^ .hydro=hydro.draught / 2.0 ) ; )
    \ turning moment as defined in 5.3 (b) \

dann ( mr / ^^ .depl=depl.mass <= .GZ10p )
    \ turning moment / depl. <= leverarm of 10 deg \
)

regel IMO167.x5.x4to6 fuer ship.stab.lc
\ 5.4 - avoid excessive high values due to accelerations
5.5 - concerning anti rolling devices
5.6 - other influences concerning stability, e.g.ice, wind... \

( frage \ Taken into account: not too high stab. values,
    anti rolling devices, ice, wind etc. ?\
)

regel IMO167.x5.x7 fuer ship.stab.lc
\ 5.6. Regard should be paid to the possible adverse
effects on stability where certain bulk cargoes are
carried. \

( wenn ( .holdcargo == "bulk" )
    dann ( ->^^.IMO123 ( ^^ ) ) \ Code of Safe Practice for Bulk Cargoes \

    wenn ( .holdcargo == "grain" )
    dann ( ->^^.SOLASVI ( ^^ .ship=objekt2 ) )
        \ requirements in Chapter VI SOLAS \
)

regel IMO167.x6 fuer ship
\ 6. Inclining Test \

( frage \ light weight is checked with inclining test ? \
)

regel IMO167.x7 fuer ship
\ 6. Stability Information \

( frage
    \ The Stability Information for the Master comply to this paragraph ? \
)

regel IMO123 fuer ship

```

\ Code of Safe Practice for Bulk Cargoes \

( frage \ fullfilled ? \ )

regel SOLASVI fuer ship

\ SOLAS '74 Chapter VI \

( frage \ fullfilled ? \ )

## C.2.2 Datenmodell

Gekürztes Datenmodell fuer Regel IMO A167

Datenmodelldatei : e:\uni\promo\zzzdat\ship Thu Mar 12 10:28:54 1992

Beginn : file=e:\uni\promo\zzzdat\solasii2

- k = startknoten

```

1 1    - ship , k = Ships Id (No. of Ship)
2 7    - regel , r = Regulations
2 6    - mach , k = Machinery (= mach)
      ... (gekuerzt)
2 5    - repro , k = Resistance and Propusion (= repro)
      ... (gekuerzt)
2 4    - weight , k = Leight Ship Weight Calculation (= weight)
3 4    - coeff , k = Coefficients for Weight Calculation-Methods (= coeff)
      ... (gekuerzt)
3 3    - group , k = Weight Group (= Item No., one should be mls)
4 9    - superstr , k = Referenz to ship.ga.superstr and
      ship.weight.coeff.superstr
5 1    - deck , k = Referenz to ship.ga.superstr.deck and
      ship.weight.coeff.superstr.deck
4 8    - subgroup , k = refers to groups, building this group (=group name)
5 1    - add , p = Factor for adding the subgroups : 1.0 or -1.0 :
      - 1.000000
4 7    - des , s = Description of Weight
4 6    - x fwd , p = forward end of weight distribution : m 1.000000
4 5    - xaft , p = aft end of weight distribution : m 1.000000
4 4    - zg , p = vertical centre of gravity : m 1.000000
4 3    - yg , p = transverse centre of gravity : m 1.000000
4 2    - xg , p = longitudinal centre of gravity : m 1.000000
4 1    - mass , p = Mass : t 1.000000
3 2    - incl , k = Inclining Test (= incl)
2 3    - stab , k = Loading Cases and Stability (= stab)
3 2    - dam , k = Damage (= Short Name)
4 10   - room , k = rooms building the lost bouyancy volume for
      this damage (room Id)
5 2    - deltad , p = draught difference for intermediat stages of
      flooding. Actual draught in room = draught
      of outer waterlevel - deltad : m 1.000000
5 1    - add , i = positive volume = 1, negative volume = -1
4 9    - yss , p = Transversal penetration from stb side : m 1.000000
4 8    - yps , p = Transversal penetration from ps : m 1.000000
4 7    - zup , p = upper limit of penetration : m 1.000000

```

```

4 6      - zlow , p = lower limit of penetration : m 1.000000
4 5      - x fwd , p = fwd end of penetration : m 1.000000
4 4      - xaft , p = aft end of penetration : m 1.000000
4 2      - type , a = Type of Damage : 5
          intact
          sscollision
          pscollision
          stranding
          ramming
4 1      - des , s = Description
3 1      - lc , k = Loading Condition (= Short Name)
4 12     - sgw , p = specific gravity of water : t/m^3 1.000000
4 11     - regel , r = Regulations
4 10     - res , k = Results, Floating Condition and Stability
          (= intact or Damage Id)
5 4      - regel , r = Regulations
5 3      - stab , k = Stability Values (= stab)
6 21     - regel , r = Regulations
6 20     - gz10n , p = leverarm at -30 deg : m 1.000000
6 19     - GZ10p , p = leverarm at +30 deg : m 1.000000
6 18     - GZ30n , p = leverarm at -30 deg : m 1.000000
6 17     - GZ30p , p = leverarm at +30 deg : m 1.000000
6 16     - open , k = values depending on a List of Openings
          (Open Id acc. to ship.ga.open)
7 8      - posfb , k = corresponding Item (Item Id acc. to
          ship.ga.open.item)
7 7      - fb , p = Freeboard at Equilibrium : m 1.000000
7 6      - An , p = area from equilibrium to neg. angle of submersion :
          m*rad 1.000000
7 5      - Ap , p = area equilibrium to pos. angle of submersion :
          m*rad 1.000000
7 4      - posPhn , k = corresponding position to .phn
          ( = ship.topo.fbl.pos )
7 3      - Phn , p = neg. Angle of heel where opening will submerge :
          deg 57.300000
7 2      - posPhp , k = corresponding position to .php
          ( = ship.topo.fbl.pos )
7 1      - Php , p = pos. Angle of heel where opening will submerge :
          deg 57.300000
6 12     - A40n , p = Area up to -40 Deg : m*rad 1.000000
6 11     - A40p , p = Area up to +40 Deg : m*rad 1.000000
6 10     - A30n , p = Area up to -30 Deg : m*rad 1.000000
6 9      - A30p , p = Area up to +30 Deg : m*rad 1.000000
6 8      - secin , p = second Intercept on neg. Angles of Heel :
          deg 57.300000
6 7      - secip , p = second Intercept on pos. Angles of Heel :
          deg 57.300000
6 6      - Phmaxn , p = Angle of Heel according to hmaxn :

```

```

                                deg 57.300000
6 5      - hmaxn , p = Max. Righting Lever at neg. Angles of Heel :
                                m 1.000000
6 4      - Phmaxp , p = Angle of Heel according to hmaxp :
                                deg 57.300000
6 3      - hmaxp , p = Max. Righting Lever at pos. Angles of Heel :
                                m 1.000000
6 2      - GMO , p = GMO : m 1.000000
6 1      - GM , p = GM corrected for free surfaces : m 1.000000
5 1      - hydro , k = Hydrostatics (= hydro)
6 17     - rooms , k = values for the damaged rooms (room id)
7 16     - lever , k = leverarm values (= lever )
8 5      - lever , v = righting lever : m 1.000000
8 4      - mom , v = Righting Moments corresponding to Heels :
                                t*m 1.000000
8 3      - heel , v = Heeling Angles : Deg 57.300000
8 2      - trim , v = Trims corresponding to Heels : m 1.000000
8 1      - draught , v = Draughts corresponding to Heels :
                                m 1.000000
7 15     - Surf , p = wetted Surface : m^2 1.000000
7 14     - Ixy , p = moment of Inertia xy : m^4 1.000000
7 13     - Iy , p = moment of Inertia y : m^4 1.000000
7 12     - Ix , p = moment of Inertia x : m^4 1.000000
7 11     - zw , p = z Waterline : m 1.000000
7 10     - yw , p = y Waterline : m 1.000000
7 9      - xw , p = x Waterline : m 1.000000
7 8      - Aw , p = Area Waterline : m^2 1.000000
7 7      - zb , p = z bouyancy : m 1.000000
7 6      - yb , p = y bouancy : m 1.000000
7 5      - xb , p = x bouancy : m 1.000000
7 4      - Vol , p = netto volume : m^3 1.000000
7 3      - ullage , p = actual ullage acc to filling : m 1.000000
7 2      - sound , p = actual sounding acc to filling : m 1.000000
7 1      - draught , p = actual draught acc to filling : m 1.000000
6 16     - dwpos , k = values for a room building a dwpos
                                (dwpos Id = room Id)
7 16     - lever , k = leverarm values (= lever )
8 5      - lever , v = righting lever : m 1.000000
8 4      - mom , v = Righting Moments corresponding to Heels :
                                t*m 1.000000
8 3      - heel , v = Heeling Angles : Deg 57.300000
8 2      - trim , v = Trims corresponding to Heels : m 1.000000
8 1      - draught , v = Draughts corresponding to Heels :
                                m 1.000000
7 15     - Surf , p = wetted Surface : m^2 1.000000
7 14     - Ixy , p = moment of Inertia xy : m^4 1.000000
7 13     - Iy , p = moment of Inertia y : m^4 1.000000
7 12     - Ix , p = moment of Inertia x : m^4 1.000000

```

```

7 11      - zw , p = z Waterline : m 1.000000
7 10      - yw , p = y Waterline : m 1.000000
7 9       - xw , p = x Waterline : m 1.000000
7 8       - Aw , p = Area Waterline : m^2 1.000000
7 7       - zb , p = z bouyancy : m 1.000000
7 6       - yb , p = y bouancy : m 1.000000
7 5       - xb , p = x bouancy : m 1.000000
7 4       - Vol , p = netto volume : m^3 1.000000
7 3       - ullage , p = actual ullage acc to filling : m 1.000000
7 2       - sound , p = actual sounding acc to filling : m 1.000000
7 1       - draught , p = actual draught acc to filling : m 1.000000
6 15     - bouyancy , k = values for a room building the total volume
          (room Id or total)
7 14     - lever , k = Leverarm values (= lever)
8 5      - lever , v = righting lever : m 1.000000
8 4      - mom , v = Righting Moments corresponding to Heels :
          t*m 1.000000
8 3      - heel , v = Heeling Angles : Deg 57.300000
8 2      - trim , v = Trims corresponding to Heels : m 1.000000
8 1      - draught , v = Draughts corresponding to Heels :
          m 1.000000
7 13     - Surf , p = wetted Surface : m^2 1.000000
7 12     - Ixy , p = moment of Inertia xy : m^4 1.000000
7 11     - Iy , p = moment of Inertia y : m^4 1.000000
7 10     - Ix , p = moment of Inertia x : m^4 1.000000
7 9      - zw , p = z Waterline : m 1.000000
7 8      - yw , p = y Waterline : m 1.000000
7 7      - xw , p = x Waterline : m 1.000000
7 6      - Aw , p = Area Waterline : m^2 1.000000
7 5      - zb , p = z bouyancy : m 1.000000
7 4      - yb , p = y bouancy : m 1.000000
7 3      - xb , p = x bouancy : m 1.000000
7 2      - Vol , p = netto volume : m^3 1.000000
7 1      - draught , p = actual draught, may differ from ^^draught
          acc. to ^^dam.deltad : m 1.000000
6 14     - cw , p = actual cw : - 1.000000
6 13     - cb , p = actual cb : - 1.000000
6 12     - trffwd , p = trimfaktor forward acc. to SBG : - 1.000000
6 11     - trfaft , p = trimfaktor aft acc to SBG : - 1.000000
6 10     - MCT , p = Moment to change trim : mt/m 1.000000
6 9      - KM , p = KM, keel to metacenter : m 1.000000
6 8      - BM , p = BM, Bouancy to metacenter : m 1.000000
6 7      - KB , p = KB, Keel to bouancy : m 1.000000
6 6      - draughtfwd , p = draught at VP : m 1.000000
6 5      - draughtaft , p = draught at AP : m 1.000000
6 3      - heel , p = Heeling Angle of Equilibrium Floating Condition :
          Deg 57.300000
6 2      - trim , p = Trim of Equilibrium Floating Condition : m 1.000000

```

```

6 1          - draught , p = Draught of Equilibrium Floating Condition :
                m 1.000000
4 9          - depl , k = Displacement (= depl)
5 7          - passmom , p = heeling moment crowding passengers on one side :
                mt 1.000000
5 6          - zgcorr , p = zg corrected for influence of free surfaces :
                m 1.000000
5 5          - itsg , p = influence of free surface : t*m 1.000000
5 4          - zg , p = vertical centre of gravity : m 1.000000
5 3          - yg , p = transverse centre of gravity : m 1.000000
5 2          - xg , p = longitudinal centre of gravity : m 1.000000
5 1          - mass , p = Mass : t 1.000000
4 8          - dw , k = Deadweight (= dw)
5 5          - itsg , p = influence of free surfaces : t*m 1.000000
5 4          - zg , p = vertical centre of gravity : m 1.000000
5 3          - yg , p = transverse centre of gravity : m 1.000000
5 2          - xg , p = longitudinal centre of gravity : m 1.000000
5 1          - mass , p = Mass : t 1.000000
4 7          - dwgroup , k = Deadweight Group
                (= Group, see ship.stab.lc.pos.group)
5 5          - itsg , p = influence of free surfaces : t*m 1.000000
5 4          - zg , p = vertical centre of gravity : m 1.000000
5 3          - yg , p = transverse centre of gravity : m 1.000000
5 2          - xg , p = longitudinal centre of gravity : m 1.000000
5 1          - mass , p = Mass : t 1.000000
4 6          - dwpos , k = Deadweight Position
                (= RoomId of corresponding Room)
5 9          - itsg , p = Influence of free Surface : t*m 1.000000
5 8          - zg , p = vertical centre of gravity : m 1.000000
5 7          - yg , p = transverse centre of gravity : m 1.000000
5 6          - xg , p = longitudinal centre of gravity : m 1.000000
5 5          - mass , p = Mass : t 1.000000
5 4          - sg , p = specific Gravity : t/m**3 1.000000
5 3          - fill , p = actual Filling : % 100.000000
5 2          - dwgroup , k = Deadweight Group
                (= Name (Crew, Hfo, Do, ...))
5 1          - medium , s = Medium / Contents
4 5          - holdcargo , a = type of cargo in holds : 9
                empty
                general
                containers
                bulk
                grain
                ore
                oil
                chemicals
                liqgas
4 4          - deckcargo , a = type of deckcargo : 4

```

```

empty
general
containers
timber
4 3 - deparr , a = Consumables : 4
empty
arrival
half
departure
4 2 - group , a = logical Grouping : 4
light_ship
docking
ballast
cargo
4 1 - title , s = Title of Loading Condition
2 2 - ga , k = General Arrangement (= ga) Definition of Decks,
Bulkheads, Volumes, Supersrtucture, ...
... (gekuerzt)
2 1 - main , k = Main Characteristics (=main)
3 37 - add , k = Additional Values (= add)
4 38 - FreeBoard , p = Freeboard : m 1.000000
4 37 - cw , p = Coefficient of Waterline on Draught d
(acc to Summer Freeboard) : - 1.000000
4 36 - cm , p = Coefficient of Mainframe on Draught d
(acc to Summer Freeboard) : - 1.000000
4 35 - cb , p = Coefficient of Volume on Draught d
(acc to Summer Freeboard) : - 1.000000
4 33 - cp , p = Prismatic Coefficient of Volume on Draught d
(acc to Summer Freeboard) : - 1.000000
4 32 - Volume , p = Volume on max Draught
(acc. to summer freeboard) : M^3 1.000000
4 31 - coeff , k = Coefficients for Calculations (= coeff)
5 2 - cCbAyre , p = Coefficient for Cb Approximation acc.to Ayre :
Cb = c + 1.68 * Fn : [ 1.06 ] : - 1.000000
5 1 - cLSchnee , p = Coefficient for Lpp Approximation
acc.to Schneekluth:
Lpp = c * Depl**0.3 * (v/0.51444)**0.3 :
- 1.000000
4 30 - FnTrial , p = Froude Number for Trail Speed : - 1.000000
4 29 - Depl , p = Displacement on Summer freeboard d : t 1.000000
4 28 - dwzDepl , p = dw / Displacement : - 1.000000
4 27 - Raeumte , p = Raeumte : m^3/t 1.000000
4 26 - nDecks , i = Number of Decks up to Main deck at H
4 25 - hDB , p = Height of Double Bootem : m 1.000000
4 24 - mls , k = Reference to ship.weight.group for Light Ship Weight Group
4 23 - HullFact , p = Hull Factor, [ 1.003, 1.008 ], std = 1.005 :
- 1.000000
4 22 - RhoW , p = spez Gravity Seawater, [ 1.025 ] : t/m^3 1.000000

```

```

4 21      - LHRatio , p = Length Height Ratio L / H : - 1.000000
4 20      - BHRatio , p = Breadth Height Ratio B / H : - 1.000000
4 19      - BDRatio , p = Breadth Draught Ratio B / D : - 1.000000
4 18      - LBRatio , p = Length Breadth Ratio Lpp / B : - 1.000000
4 14      - firemethod , a = Fire protection method acc. to SOLAS II-2 Part C : 3
          IC
          IIC
          IIIC
4 13      - AEtoAP , p = Length between aftmost Part of Vessel to Aft Perp. :
          m 1.000000
4 12      - VolBDpBackpHatches , p = Volume below Deck + Back + Hatches :
          m^3 1.000000
4 11      - VolBDpHatches , p = Volume Below Main Deck incl. Hatches :
          m^3 1.000000
4 10      - coeff , k = Coefficients for Calculation Methods ( = coeff )
5 1      - C4CBD , p = Coefficient for CBD Approximation acc. to Schneekluth,
          [0.25 for small, 0.5 for heavy Flare (Spantausfall)] :
          - 1.000000
4 9       - Hcorr , p = Depth H corrected for Volume above H : m 1.000000
4 8       - CBD , p = cb up to Maindeck (H) : - 1.000000
4 3       - VolSuperstr , p = Total Volume of Superstructures : m^3 1.000000
4 1       - VolBelowDeck , p = Belowdecks Volume (bis H) : m^3 1.000000
3 36      - id , s = International Ships Id (LR Number)
3 35      - name , s = Name of Ship
3 34      - callsign , s = Call Sign
3 33      - typ , a = Type of Vessel : 10
          PassengerShip
          ROROShip
          GeneralCargoContainer
          BulkShip
          TankShip
          FisheryShip
          SpecialShip
          Barge
          MobileOffshoreUnit
          InlandCraft
3 32      - tsys , i = Type of Vessel acc. to IfS/GL Classification
3 31      - yard , s = Name of Yard
3 30      - yardno , s = Yard No.
3 29      - keeldate , i = Date of stretching the Keel (jjmdd)
3 28      - delivery , i = Date of Delivery (jjmdd)
3 27      - conversion , k = Conversions, Repairs, ... (short id)
4 3       - date , i = Date of Conversion, Repair (jjmdd)
4 2       - type , a = Type of Conversion, Repair : 2
          minor
          major
4 1       - descr , s = Description
3 26      - owner , s = Owners Name

```

```
3 25      - country , s = Country of Registration
3 24      - classs , a = Classification Society : 5
          gl
          abs
          lr
          dnv
          bv
3 23      - class , s = Classification Sign
3 22      - lpp , p = Length between Perpendiculars : m 1.000000
3 21      - loa , p = Length over all : m 1.000000
3 20      - lwl , p = Length in Waterline : m 1.000000
3 19      - lfb , p = Length acc. to Freeboard convention : m 1.000000
3 18      - b , p = Breadth Moulded : m 1.000000
3 17      - bwl , p = Breadth in Waterline : m 1.000000
3 16      - dscantl , p = max. Draught acc. to Scantlings : m 1.000000
3 15      - d , p = max. Draught acc. to Summer Freeboard : m 1.000000
3 14      - h , p = Height of 1. Deck (Freeboarddeck) : m 1.000000
3 13      - dw , p = Deadweight : t 1.000000
3 12      - npass , i = No of Passengers
3 11      - ncrew , i = No of Crew
3 10      - brz , p = BRZ acc. to Oslo 1978 : - 1.000000
3 9       - nrz , p = NRZ acc. to Oslo 1978 : - 1.000000
3 8       - brt , p = gross tonnes : - 1.000000
3 7       - nrt , p = netto tonnes : - 1.000000
3 6       - VolBale , p = Cargo Hold Volume Bale : m^3 1.000000
3 5       - VolGrain , p = Cargo Hold Volume Grain : m^3 1.000000
3 4       - vtrial , p = Trial Speed : kn 0.514444
3 3       - vservice , p = Service Speed : kn 0.514444
3 2       - pb , p = total Breakpower of Main Engines : kW 1.000000
3 1       - nprop , i = No. of Propellers
```

## C.2.3 Daten

Wertedatei : e:\uni\promo\zzzdat\imoa167 Wed Mar 11 16:20:52 1992

Datenmodelldatei : e:\uni\promo\zzzdat\ship Thu Mar 12 12:59:50 1992

Beginn : file=e:\uni\promo\zzzdat\imoa167

```
- k = startknoten

1   - ship = B227
2     - stab = stab
3       - lc = Cond7
4         - holdcargo = containers
4         - res = intact
5           - stab = stab
6             - GM0 = 0.450000 m , 0.450000 m
6             - hmaxp = 0.300000 m , 0.300000 m
6             - Phmaxp = 44.999982 deg , 44.999982 deg
6             - A40p = 0.132000 m*rad , 0.132000 m*rad
6             - A30p = 0.074000 m*rad , 0.074000 m*rad
6             - open = prog
7               - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = containers
4         - group = cargo
3       - lc = Cond6
4         - holdcargo = containers
4         - res = intact
5           - stab = stab
6             - GZ30p = 0.158000 m , 0.158000 m
6             - GM0 = 0.200000 m , 0.200000 m
6             - hmaxp = 0.250000 m , 0.250000 m
6             - Phmaxp = 25.999875 deg , 25.999875 deg
6             - A40p = 0.105000 m*rad , 0.105000 m*rad
6             - A30p = 0.058000 m*rad , 0.058000 m*rad
6             - open = prog
7               - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = containers
4         - group = cargo
3       - lc = Cond5
4         - holdcargo = general
4         - res = intact
5           - stab = stab
6             - GM0 = 0.800000 m , 0.800000 m
6             - hmaxp = 0.620000 m , 0.620000 m
6             - Phmaxp = 44.999982 deg , 44.999982 deg
6             - A40p = 0.192000 m*rad , 0.192000 m*rad
6             - A30p = 0.106000 m*rad , 0.106000 m*rad
```

```
6         - open = prog
7           - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = empty
4         - group = cargo
3       - lc = Cond4
4         - holdcargo = general
4         - res = intact
5         - stab = stab
6           - GM0 = 0.870000 m , 0.870000 m
6           - hmaxp = 0.610000 m , 0.620000 m
6           - Phmaxp = 44.999982 deg , 47.999981 deg
6           - A40p = 0.170000 m*rad , 0.170000 m*rad
6           - A30p = 0.093000 m*rad , 0.093000 m*rad
6         - open = prog
7           - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = empty
4         - group = cargo
3       - lc = Cond3
4         - holdcargo = empty
4         - res = intact
5         - stab = stab
6           - GM0 = 2.660000 m , 2.660000 m
6           - hmaxp = 1.950000 m , 1.950000 m
6           - Phmaxp = 56.999977 deg , 56.999977 deg
6           - A40p = 0.673000 m*rad , 0.673000 m*rad
6           - A30p = 0.387000 m*rad , 0.387000 m*rad
6         - open = prog
7           - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = empty
4         - group = ballast
3       - lc = Cond2
4         - holdcargo = empty
4         - res = intact
5         - stab = stab
6           - GM0 = 2.580000 m , 2.580000 m
6           - hmaxp = 1.810000 m , 1.810000 m
6           - Phmaxp = 54.999978 deg , 54.999978 deg
6           - A40p = 0.613000 m*rad , 0.613000 m*rad
6           - A30p = 0.351000 m*rad , 0.351000 m*rad
6         - open = prog
7           - Php = 90.000021 deg , 90.000021 deg
4         - deckcargo = empty
4         - group = ballast
3       - lc = Cond1
4         - group = light_ship
2     - main = main
3       - lfb = 78.000000 m , 79.000000 m
3       - typ = GeneralCargoContainer
```

## C.2.4 Prüfen, tabellarische Ausgabe, kurz

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'IMO\_A167'  
folgende Regel erfuehlt ist :

Regel 'IMO167'  
fuer 'ship=B227' aus e:\uni\promo\zzzdat\imoa167

| Ebene | Regel  | angewendet auf  | er- gibt   |
|-------|--|---|--|
| 0     | IMO167<br>  IMO Resolution A167, Amended A206<br>  Recommendation on intact stability for passenger<br>  and cargo ships under 100 meters in length<br>  as amended with respect to ships carrying deck cargoes  | ship=B227   | 0.000  |
| 1     | IMO167.x1<br>  1. Scope  | ship=B227   | 0.000  |
| 2     | IMO167.x1.x1<br>  1.1 The provision given hereunder are recommended<br>  for new decked sea-going passenger and cargo ships<br>  (other than fishing vessel) under 100 meters in length.   | ship=B227   | 0.000  |
| 3     | IMO167.x1.x2<br>  1.2 Administrations are invited to adopt,<br>  for all conditions of loading, the stability<br>  criteria given in paragraph 5 unless they are<br>  satisfied that operating experience justifies<br>  departures therefrom<br>  -fuer alle Schleife<br>  0. Teilbereich<br>  1. Teilbereich<br>  2. Teilbereich<br>  3. Teilbereich<br>  4. Teilbereich<br>  5. Teilbereich | ship=B227<br> .stab=stab.lc<br> ship=B227.stab=stab.lc=Cond7<br> ship=B227.stab=stab.lc=Cond6<br> ship=B227.stab=stab.lc=Cond5<br> ship=B227.stab=stab.lc=Cond4<br> ship=B227.stab=stab.lc=Cond3<br> ship=B227.stab=stab.lc=Cond2 | 0.000 <br> <br> 1.000 <br> 0.000 <br> 1.000 <br> 1.000 <br> 1.000 <br> 1.000 |
| 4     | IMO167.x5<br>  5. Recommended Criteria   | ship=B227.stab=stab.lc=Cond6  | 0.000  |

```
5|IMOA167.x5.x1                |b.lc=Cond6.res=intact.stab=stab|0.000|
| 5.1. The following criteria are recommended          |
| for passenger and cargo ships                       |
-----
6|IMOA167.x5.x1.b              |b.lc=Cond6.res=intact.stab=stab|0.000|
| 5.1 (b) The righting lever GZ should be at least 0.20 m |
| at an angle of heel equal to or greater than 30 deg    |
-----
```

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10

## C.2.5 Relevanz, tabellarische Ausgabe, Überblick

```
+-----+
| Relevanz |
+-----+
```

Folgende Kriterien sind in  
'IMO\_A167'  
zu erfuellen :

Regel 'IMOA167'  
fuer 'ship=B227' aus e:\uni\promo\zzzdat\imoa167

| Ebene | Regel   | angewendet auf   | er- gibt  |
|-------|---|--|---|
| 0     | IMOA167<br>  IMO Resolution A167, Amended A206<br>  Recommendation on intact stability for passenger<br>  and cargo ships under 100 meters in length<br>  as amended with respect to ships carrying deck cargoes  | ship=B227  | 1.000   |
| 1     | IMOA167.x1<br>  1. Scope  | ship=B227  | 1.000   |
| 2     | IMOA167.x1.x1<br>  1.1 The provision given hereunder are recommended<br>  for new decked sea-going passenger and cargo ships<br>  (other than fishing vessel) under 100 meters in length.   | ship=B227  | 1.000   |
| 3     | IMOA167.x1.x2<br>  1.2 Administrations are invited to adopt,<br>  for all conditions of loading, the stability<br>  criteria given in paragraph 5 unless they are<br>  satisfied that operating experience justifies<br>  departures therefrom<br>  -fuer alle Schleife<br>  0. Teilbereich<br>  1. Teilbereich<br>  2. Teilbereich<br>  3. Teilbereich<br>  4. Teilbereich<br>  5. Teilbereich<br>  6. Teilbereich | ship=B227<br> .stab=stab.lc<br> ship=B227.stab=stab.lc=Cond7<br> ship=B227.stab=stab.lc=Cond6<br> ship=B227.stab=stab.lc=Cond5<br> ship=B227.stab=stab.lc=Cond4<br> ship=B227.stab=stab.lc=Cond3<br> ship=B227.stab=stab.lc=Cond2<br> ship=B227.stab=stab.lc=Cond1 | 1.000 <br> <br> 1.000 <br> 1.000 <br> 1.000 <br> 1.000 <br> 1.000 <br> 1.000 <br> 0.000 |
| 4     | IMOA167.x5  | ship=B227.stab=stab.lc=Cond7   | 1.000   |

|   |                                 |       |
|---|---------------------------------|-------|
| 5. Recommended Criteria                                       |                                 |       |
| -----   |                                 |       |
| innere Regel 0  | ship=B227.stab=stab.lc=Cond7    | 1.000 |
| -----   |                                 |       |
| 5 IMOA167.x5.x1   | b.lc=Cond7.res=intact.stab=stab | 1.000 |
| 5.1. The following criteria are recommended                   |                                 |       |
| for passenger and cargo ships                                 |                                 |       |
| -----   |                                 |       |
| 6 IMOA167.x5.x1.a   | b.lc=Cond7.res=intact.stab=stab | 1.000 |
| 5.1 (a) The area under the righting lever curve               |                                 |       |
| (GZ curve) should not be less than the following values:      |                                 |       |
| up to 30 deg $\geq 0.055$ m*rad,                              |                                 |       |
| up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad    |                                 |       |
| between 30 and 40 deg $\geq 0.03$ m*rad                       |                                 |       |
| -----   |                                 |       |
| 6 IMOA167.x5.x1.b   | b.lc=Cond7.res=intact.stab=stab | 1.000 |
| 5.1 (b) The righting lever GZ should be at least 0.20 m       |                                 |       |
| at an angle of heel equal to or greater than 30 deg           |                                 |       |
| -----   |                                 |       |
| 6 IMOA167.x5.x1.c   | b.lc=Cond7.res=intact.stab=stab | 1.000 |
| 5.1 (c) The maximum righting arm should occur at              |                                 |       |
| an angle of heel preferably exceeding 30 deg                  |                                 |       |
| but not less than 25 deg.                                     |                                 |       |
| -----   |                                 |       |
| 6 IMOA167.x5.x1.d   | b.lc=Cond7.res=intact.stab=stab | 1.000 |
| 5.1 (d) The initial metacentric height GM0 should not         |                                 |       |
| be less than 0.15 .   |                                 |       |
| -----   |                                 |       |
| 5 IMOA167.x5.x4to6  | ship=B227.stab=stab.lc=Cond7    | 1.000 |
| 5.4 - avoid excessive high values due to accelerations        |                                 |       |
| 5.5 - concerning anti rolling devices                         |                                 |       |
| 5.6 - other influences concerning stability, e.g.ice, wind... |                                 |       |
| -----   |                                 |       |
| 5 IMOA167.x5.x7   | ship=B227.stab=stab.lc=Cond7    | 0.000 |
| 5.6. Regard should be paid to the possible adverse            |                                 |       |
| effects on stability where certain bulk cargoes are           |                                 |       |
| carried.  |                                 |       |
| -----   |                                 |       |
| 4 IMOA167.x5  | ship=B227.stab=stab.lc=Cond6    | 1.000 |
| 5. Recommended Criteria                                       |                                 |       |
| -----   |                                 |       |
| innere Regel 0  | ship=B227.stab=stab.lc=Cond6    | 1.000 |
| -----   |                                 |       |
| 5 IMOA167.x5.x1   | b.lc=Cond6.res=intact.stab=stab | 1.000 |
| 5.1. The following criteria are recommended                   |                                 |       |
| for passenger and cargo ships                                 |                                 |       |
| -----   |                                 |       |
| 6 IMOA167.x5.x1.a   | b.lc=Cond6.res=intact.stab=stab | 1.000 |

|   |                                       |  |
|---|---------------------------------------|--|
| 5.1 (a) The area under the righting lever curve               |                                       |  |
| (GZ curve) should not be less than the following values:      |                                       |  |
| up to 30 deg $\geq 0.055$ m*rad,                              |                                       |  |
| up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad    |                                       |  |
| between 30 and 40 deg $\geq 0.03$ m*rad                       |                                       |  |
| -----   |                                       |  |
| 6 IMOA167.x5.x1.b   | b.lc=Cond6.res=intact.stab=stab 1.000 |  |
| 5.1 (b) The righting lever GZ should be at least 0.20 m       |                                       |  |
| at an angle of heel equal to or greater than 30 deg           |                                       |  |
| -----   |                                       |  |
| 6 IMOA167.x5.x1.c   | b.lc=Cond6.res=intact.stab=stab 1.000 |  |
| 5.1 (c) The maximum righting arm should occur at              |                                       |  |
| an angle of heel preferably exceeding 30 deg                  |                                       |  |
| but not less than 25 deg.                                     |                                       |  |
| -----   |                                       |  |
| 6 IMOA167.x5.x1.d   | b.lc=Cond6.res=intact.stab=stab 1.000 |  |
| 5.1 (d) The initial metacentric height GM0 should not         |                                       |  |
| be less than 0.15 .   |                                       |  |
| -----   |                                       |  |
| 5 IMOA167.x5.x4to6  | ship=B227.stab=stab.lc=Cond6  1.000   |  |
| 5.4 - avoid excessive high values due to accelerations        |                                       |  |
| 5.5 - concerning anti rolling devices                         |                                       |  |
| 5.6 - other influences concerning stability, e.g.ice, wind... |                                       |  |
| -----   |                                       |  |
| 5 IMOA167.x5.x7   | ship=B227.stab=stab.lc=Cond6  0.000   |  |
| 5.6. Regard should be paid to the possible adverse            |                                       |  |
| effects on stability where certain bulk cargoes are           |                                       |  |
| carried.  |                                       |  |
| -----   |                                       |  |
| 4 IMOA167.x5  | ship=B227.stab=stab.lc=Cond5  1.000   |  |
| 5. Recommended Criteria                                       |                                       |  |
| -----   |                                       |  |
| innere Regel 0  | ship=B227.stab=stab.lc=Cond5  1.000   |  |
| -----   |                                       |  |
| 5 IMOA167.x5.x1   | b.lc=Cond5.res=intact.stab=stab 1.000 |  |
| 5.1. The following criteria are recommended                   |                                       |  |
| for passenger and cargo ships                                 |                                       |  |
| -----   |                                       |  |
| 6 IMOA167.x5.x1.a   | b.lc=Cond5.res=intact.stab=stab 1.000 |  |
| 5.1 (a) The area under the righting lever curve               |                                       |  |
| (GZ curve) should not be less than the following values:      |                                       |  |
| up to 30 deg $\geq 0.055$ m*rad,                              |                                       |  |
| up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad    |                                       |  |
| between 30 and 40 deg $\geq 0.03$ m*rad                       |                                       |  |
| -----   |                                       |  |
| 6 IMOA167.x5.x1.b   | b.lc=Cond5.res=intact.stab=stab 1.000 |  |
| 5.1 (b) The righting lever GZ should be at least 0.20 m       |                                       |  |
| at an angle of heel equal to or greater than 30 deg           |                                       |  |

|   |                                       |
|---|---------------------------------------|
| 6 IMOA167.x5.x1.c   | b.lc=Cond5.res=intact.stab=stab 1.000 |
| 5.1 (c) The maximum righting arm should occur at              |                                       |
| an angle of heel preferably exceeding 30 deg                  |                                       |
| but not less than 25 deg.                                     |                                       |
| 6 IMOA167.x5.x1.d   | b.lc=Cond5.res=intact.stab=stab 1.000 |
| 5.1 (d) The initial metacentric height GMO should not         |                                       |
| be less than 0.15 .   |                                       |
| 5 IMOA167.x5.x4to6  | ship=B227.stab=stab.lc=Cond5  1.000   |
| 5.4 - avoid excessive high values due to accelerations        |                                       |
| 5.5 - concerning anti rolling devices                         |                                       |
| 5.6 - other influences concerning stability, e.g.ice, wind... |                                       |
| 5 IMOA167.x5.x7   | ship=B227.stab=stab.lc=Cond5  0.000   |
| 5.6. Regard should be paid to the possible adverse            |                                       |
| effects on stability where certain bulk cargoes are           |                                       |
| carried.  |                                       |
| 4 IMOA167.x5  | ship=B227.stab=stab.lc=Cond4  1.000   |
| 5. Recommended Criteria                                       |                                       |
| innere Regel 0  | ship=B227.stab=stab.lc=Cond4  1.000   |
| 5 IMOA167.x5.x1   | b.lc=Cond4.res=intact.stab=stab 1.000 |
| 5.1. The following criteria are recommended                   |                                       |
| for passenger and cargo ships                                 |                                       |
| 6 IMOA167.x5.x1.a   | b.lc=Cond4.res=intact.stab=stab 1.000 |
| 5.1 (a) The area under the righting lever curve               |                                       |
| (GZ curve) should not be less than the following values:      |                                       |
| up to 30 deg $\geq 0.055$ m*rad,                              |                                       |
| up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad    |                                       |
| between 30 and 40 deg $\geq 0.03$ m*rad                       |                                       |
| 6 IMOA167.x5.x1.b   | b.lc=Cond4.res=intact.stab=stab 1.000 |
| 5.1 (b) The righting lever GZ should be at least 0.20 m       |                                       |
| at an angle of heel equal to or greater than 30 deg           |                                       |
| 6 IMOA167.x5.x1.c   | b.lc=Cond4.res=intact.stab=stab 1.000 |
| 5.1 (c) The maximum righting arm should occur at              |                                       |
| an angle of heel preferably exceeding 30 deg                  |                                       |
| but not less than 25 deg.                                     |                                       |
| 6 IMOA167.x5.x1.d   | b.lc=Cond4.res=intact.stab=stab 1.000 |
| 5.1 (d) The initial metacentric height GMO should not         |                                       |
| be less than 0.15 .   |                                       |

|   |                                 |       |
|---|---------------------------------|-------|
| 5 IMOA167.x5.x4to6  | ship=B227.stab=stab.lc=Cond4    | 1.000 |
| 5.4 - avoid excessive high values due to accelerations        |                                 |       |
| 5.5 - concerning anti rolling devices                         |                                 |       |
| 5.6 - other influences concerning stability, e.g.ice, wind... |                                 |       |
| 5 IMOA167.x5.x7   | ship=B227.stab=stab.lc=Cond4    | 0.000 |
| 5.6. Regard should be paid to the possible adverse            |                                 |       |
| effects on stability where certain bulk cargoes are           |                                 |       |
| carried.  |                                 |       |
| 4 IMOA167.x5  | ship=B227.stab=stab.lc=Cond3    | 1.000 |
| 5. Recommended Criteria                                       |                                 |       |
| innere Regel 0  | ship=B227.stab=stab.lc=Cond3    | 1.000 |
| 5 IMOA167.x5.x1   | b.lc=Cond3.res=intact.stab=stab | 1.000 |
| 5.1. The following criteria are recommended                   |                                 |       |
| for passenger and cargo ships                                 |                                 |       |
| 6 IMOA167.x5.x1.a   | b.lc=Cond3.res=intact.stab=stab | 1.000 |
| 5.1 (a) The area under the righting lever curve               |                                 |       |
| (GZ curve) should not be less than the following values:      |                                 |       |
| up to 30 deg $\geq 0.055$ m*rad,                              |                                 |       |
| up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad    |                                 |       |
| between 30 and 40 deg $\geq 0.03$ m*rad                       |                                 |       |
| 6 IMOA167.x5.x1.b   | b.lc=Cond3.res=intact.stab=stab | 1.000 |
| 5.1 (b) The righting lever GZ should be at least 0.20 m       |                                 |       |
| at an angle of heel equal to or greater than 30 deg           |                                 |       |
| 6 IMOA167.x5.x1.c   | b.lc=Cond3.res=intact.stab=stab | 1.000 |
| 5.1 (c) The maximum righting arm should occur at              |                                 |       |
| an angle of heel preferably exceeding 30 deg                  |                                 |       |
| but not less than 25 deg.                                     |                                 |       |
| 6 IMOA167.x5.x1.d   | b.lc=Cond3.res=intact.stab=stab | 1.000 |
| 5.1 (d) The initial metacentric height GMO should not         |                                 |       |
| be less than 0.15 .   |                                 |       |
| 5 IMOA167.x5.x4to6  | ship=B227.stab=stab.lc=Cond3    | 1.000 |
| 5.4 - avoid excessive high values due to accelerations        |                                 |       |
| 5.5 - concerning anti rolling devices                         |                                 |       |
| 5.6 - other influences concerning stability, e.g.ice, wind... |                                 |       |
| 5 IMOA167.x5.x7   | ship=B227.stab=stab.lc=Cond3    | 0.000 |
| 5.6. Regard should be paid to the possible adverse            |                                 |       |
| effects on stability where certain bulk cargoes are           |                                 |       |

|   |   |                                       |
|---|---|---------------------------------------|
|   | carried.  |                                       |
| 4 | IMOA167.x5<br>  5. Recommended Criteria   | ship=B227.stab=stab.lc=Cond2  1.000   |
|   | innere Regel 0  | ship=B227.stab=stab.lc=Cond2  1.000   |
| 5 | IMOA167.x5.x1<br>  5.1. The following criteria are recommended<br>  for passenger and cargo ships   | b.lc=Cond2.res=intact.stab=stab 1.000 |
| 6 | IMOA167.x5.x1.a<br>  5.1 (a) The area under the righting lever curve<br>  (GZ curve) should not be less than the following values:<br>  up to 30 deg $\geq 0.055$ m*rad,<br>  up to 40 deg or angle of progr. flooding $\geq 0.09$ m*rad<br>  between 30 and 40 deg $\geq 0.03$ m*rad | b.lc=Cond2.res=intact.stab=stab 1.000 |
| 6 | IMOA167.x5.x1.b<br>  5.1 (b) The righting lever GZ should be at least 0.20 m<br>  at an angle of heel equal to or greater than 30 deg   | b.lc=Cond2.res=intact.stab=stab 1.000 |
| 6 | IMOA167.x5.x1.c<br>  5.1 (c) The maximum righting arm should occur at<br>  an angle of heel preferably exceeding 30 deg<br>  but not less than 25 deg.  | b.lc=Cond2.res=intact.stab=stab 1.000 |
| 6 | IMOA167.x5.x1.d<br>  5.1 (d) The initial metacentric height GMO should not<br>  be less than 0.15 .   | b.lc=Cond2.res=intact.stab=stab 1.000 |
| 5 | IMOA167.x5.x4to6<br>  5.4 - avoid excessive high values due to accelerations<br>  5.5 - concerning anti rolling devices<br>  5.6 - other influences concerning stability, e.g.ice, wind...  | ship=B227.stab=stab.lc=Cond2  1.000   |
| 5 | IMOA167.x5.x7<br>  5.6. Regard should be paid to the possible adverse<br>  effects on stability where certain bulk cargoes are<br>  carried.  | ship=B227.stab=stab.lc=Cond2  0.000   |
| 3 | IMOA167.x2to4<br>  2. General Precautions<br>  3. Calculation of Stability Curves<br>  4. Assessment of Compliance with Criteria<br>  paragraphs 2, 3, 4 are fulfilled  | ship=B227  1.000                      |
| 3 | IMOA167.x6  | ship=B227  1.000                      |

```
| 6. Inclining Test |
-----|
3|IMO A167.x7 |ship=B227 |1.000|
| 6. Stability Information |
-----|
```

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet

Integration mittels Trapezregel

Anzahl Stützstellen : 10

'rechnen' wurde ausgeführt

## C.3 SOLAS 83 II-2

### C.3.1 Regelcode

SOLASII-2

e:\uni\promo\zzzdat\ship

```
regel SOLASII2 fuer ship
\SOLAS '74 incl. Ammendments up to '83
Chapter II-2
Construction - Fire Protection, Fire Detection and Fire Extinction\
```

```
( dann ( ->.PartA )
)
```

```
regel SOLASII2.PartA fuer ship
\General\
```

```
( dann ( ->.Reg1 )
)
```

```
regel SOLASII2.PartA.Reg1 fuer ship
\Application\
```

```
( dann ( ->.x1 & ->.x2 & ->.x3 & ->.x4 )
)
```

```
regel SOLASII2.PartA.Reg1.x1 fuer ship
\ 1.1 Unless expressly provided otherwise, this chapter
shall apply to ships the keel of which are laid or which
are at a similar stage of construction on or after
1.July 1986.\
```

```
( wenn ( .main=main.keeldate >= 860701 )
      \ date of laying the keel is younger than 1.July 86\
      dann ( ->^.appl ) \ Application for different ships \
)
```

```
regel SOLASII2.PartA.Reg1.x2 fuer ship
\ Unless expressly provided otherwise, for ship constructed
before 1. July 86 the Administration shall ensure that the
requirements which are applicable under chapter II-2 of
```

SOLAS '74 as amended Nov. '81 are complied with.\

```
( wenn ( .main=main.keeldate < 860701 )
      \ date of laying the keel is older than 1.July 86\
      dann ( ->~~~~.SOLAS81III2 )
    )
```

regel SOLASII2.PartA.Reg1.x3 fuer ship  
 \ All ships which undergoing repairs ....  
 Repairs, ... of a major character and outfitting related  
 thereto shall meet the requirements for ships constructed  
 on or after 1.July 86 in so far as the administration  
 deems reasonable and practicable. \

```
( wenn ( einervon .main=main.conversion
      dann ( .date >= 860701 & .type == "major" )
      \ this conversion is a major one and is newer than 1. July 86 \
    )
    dann ( ->^.appl )
  )
```

regel SOLASII2.PartA.Reg1.x4 fuer ship

```
( dann ( ->.x1 & ->.x2 )
  )
```

regel SOLASII2.PartA.Reg1.x4.x1 fuer ship  
 \The Administration of a State may, if it considers that  
 the shelterd nature and conditions of the voyage are such  
 ... exempt of the requirements ... \

```
( wenn ( ->.exception )
      dann ( ->.exfullf )
    )
```

regel SOLASII2.PartA.Reg1.x4.x1.exception fuer ship  
 \The Administration of a State may, if it considers that  
 the shelterd nature and conditions of the voyage are such  
 ... exempt of the requirements ... \

```
( frage \ Werden Ausnahmen nach Regel 1.4.1 zugelassen ? \ )
```

regel SOLASII2.PartA.Reg1.x4.x1.exfullf fuer ship  
 \ Die Vorschrift mit den Ausnahmen nach Regel 1.4.1 \

```
( frage \ Ist die Vorschrift mit den Ausnahmen erfuehlt ?
Jede Regel muB einzeln geprueft werden!\ )
```

```
regel SOLASII2.PartA.Reg1.x4.x2 fuer ship
\ In the case of passenger ships which are employed in
special trades for the carriage of large numbers of
special trade passengers ...
may exempt from this requirements ...\
```

```
( wenn ( .main=main.typpsys == 020 ) \ Special Trade Passenger Ship \
dann ( wenn ( ->.exception )
dann ( ->~~~~~.SpecTradePass )
)
)
```

```
regel SOLASII2.PartA.Reg1.x4.x2.exception fuer ship
\ In the case of passenger ships which are employed in
special trades for the carriage of large numbers of
special trade passengers ...
may exempt from this requirements ...\
```

```
( frage \ Werden Ausnahmen nach Regel 1.4.2 zugelassen ? \ )
```

```
regel SOLAS81II2 fuer ship
\SOLAS 74 mit 81er amendments II 2\
```

```
( frage \ erfuehlt? \
)
```

```
regel SpecTradePass fuer ship
\ 1.the rules annexed to the Special Trade Passenger Ships
Agreement, 1971; and
2. the rules annexed to the Protocol on Space Requirements
for Special Trade Passenger Ships, 1973.\
```

```
( frage \ erfuehlt? \ )
```

```
regel SOLASII2.PartA.Reg1.appl fuer ship
\ Application of the Parts B, C, D \
```

```
( wenn ( .main=main.typ == "PassengerShip" \ Passenger Ship \
& .main=main.brt >= 500 \Chapter I,Part A, Reg3\ )
dann ( ->~~~.PartB )
```

```

wenn ( "ROROShip" <= .main=main.typ <= "BulkShip" \ Cargo Ship \
      & .main=main.brt >= 500 \Chapter I,Part A, Reg3\ )
dann ( ->^^.PartC )

wenn ( .main=main.typ == "TankShip" \ Tanker \
      & .main=main.brt >= 500 \Chapter I,Part A, Reg3\ )
dann ( ->^^.PartC & ->^^.PartD )
)

regel SOLASII2.PartA.Reg3.x10 fuer ship.ga.room
\ Def. 10: Accomodation spaces are those spaces used for
public spaces, corridors, lvatories,cabins,offices, hospitals,
cinemas, games and hobbies rooms, barber shops, pantries
containing no cooking appliances and similar spaces \

( wenn ( "PassengerCorridor" <= .solastype <= "CrewLobby" |
      "CabinFurnishedRestrictedFireRisk" <= .solastype <=
          "PublicSpaceFurnishedOtherThanRestrictedFireRisk" |
      "SmallStoreRoomInAccomodation" <= .solastype <= "Pharmacie" |
      "BarberShop" <= .solastype <= "IsolatedPantryNoCookingAppliances"
      )
dann ( wahr )

wenn ( default )
dann ( falsch )

)

regel SOLASII2.PartA.Reg3.x11 fuer ship.ga.room
\Def. 11: Public spaces are those portions of the
accomodation which are used for halls, dining rooms,
lounges and similar permanently enclosed spaces.\

( wenn ( "PassengerLobby" <= .solastype <= "CrewLobby" |
      .solastype == "PublicSpaceFurnishedRestrictedFireRisk" |
      .solastype == "PublicSpaceFurnishedOtherThanRestrictedFireRisk" |
      .solastype == "IndoorSwimmingPool"
      )
dann ( wahr ) \Space is a public space\

wenn ( default )
dann ( falsch ) \Space is not a public space\

)

```

regel SOLASII2.PartA.Reg3.x12 fuer ship.ga.room  
 \Def. 12: Service spaces are those spaces used for  
 galleys, pantries containing cooking appliances, lockers,  
 mail and specie rooms, store rooms, workshops other than  
 those forming part of the machinery spaces and similar  
 spaces and trunks to such spaces\

```
( wenn ( .solastype == "IsolatedLocker" |
  "DryingRoom" <= .solastype <= "SpecieRoom" |
  "MainGalley" <= .solastype <= "Workshop" |
  "StoreRoomContainingFlammableLiquids" <= .solastype <= "Laboratories"
  )
  dann ( wahr ) \Space is a service space\

  wenn ( default )
  dann ( falsch ) \Space is not a service space\
)
```

regel SOLASII2.PartA.Reg3.x14 fuer ship.ga.room  
 \Ro-ro cargo spaces are spaces not ....\

```
( wenn ( "OpenRoRoCargoSpace" <= .solastype <= "ClosedRoRoCargoSpace" )
  dann ( wahr ) \Space is a ro-ro cargo space\

  wenn ( default )
  dann ( falsch ) \Space is not a ro-ro cargo space\
)
```

regel SOLASII2.PartA.Reg3.x19 fuer ship.ga.room  
 \Machinery spaces of category A are those spaces and trunks  
 to such spaces which contain:  
 .1 internal combustion machinery used for main propulsion  
 .2 internal combustion machinery ... not less than 375kW  
 .3 any oil fired boiler or oil fuel unit\

```
( wenn ( "MainPropulsionMachineryRoom" <= .solastype <= "BoilerCasing" |
  ( "InternalCombustionEngineDrivingEmergencyGenRoomOrServingTrunk"
    <=.solastype <=
    "InternalCombustionEngineDrivingBilgePumpRoomOrServingTrunk" |
    .solastype == "AuxiliaryMachineryRoom"
  ) & .instpower >= 375 \ Inst. Power >= 375 kw \
  dann ( wahr ) \space is macinery space of category A\

  wenn ( default )
  dann ( falsch ) \space is not a machinery space of category A\
)
```

```

regel SOLASII2.PartA.Reg3.x20 fuer ship.ga.room
\Machinery spaces are all machinery spaces of category A and
all other spaces containing....\

( wenn ( "VentilationRoom" <= .solastype <= "CableTrunk" |
        "ShaftAlleyAndStorageOfCombustiblesOrServingTrunk" <= .solastype <=
        "AuxiliaryMachineryRoom" )
    dann ( wahr ) \Space is a macinery space\

    wenn ( default )
    dann ( falsch ) \Space is not a machinery space\

)

```

```

regel SOLASII2.PartB fuer ship
\SOLAS II 2, Part B
Fire Safety Measures for Passenger Ships\

( wenn ( .main=main.typ == "PassengerShip" )
    dann ( ->.fullf ) \SOLAS II 2, Part B fullfilled\
)

```

```

regel SOLASII2.PartB.fullf fuer ship
\SOLAS II 2, Part B \

( frage \ erfuehlt ? \
)

```

```

regel SOLASII2.PartD fuer ship
\SOLAS II 2, Part D
Fire Safety Measures for Tankers\

( wenn ( .main=main.typ == "TankShip" )
    dann ( ->.fullf ) \SOLAS II 2, Part D fullfilled\
)

```

```

regel SOLASII2.PartD.fullf fuer ship
\SOLAS II 2, Part D\

( frage \ erfuehlt ? \
)

```

```

regel SOLASII2.PartC fuer ship
\Fire Safety Measures for Cargo Ships\

( wenn ( .main=main.typ >= "ROROShip" & .main=main.typ <= "TankShip" )
      \ Cargo Ship \
      dann ( ->.Reg42 & ->.Reg43 & ->.Reg44 & ->.Reg45to53 & ->.Reg54 )
      \SOLAS II 2, Part C fullfilled\
)

```

```

regel SOLASII2.PartC.Reg42 fuer ship
\ Structure \

```

```

( dann ( ->.fullf )
)

```

```

regel SOLASII2.PartC.Reg42.fullf fuer ship
\ Structure \

```

```

( frage \ Erfuelllt ? \
)

```

```

regel SOLASII2.PartC.Reg45to53 fuer ship
\ Means of escape
Protection of stairways and lift trunks in accom.....
Doors in fire resisting divisions
Ventilation systems
Restricted use of combustibile materials
Details of construction
Arrangement for gaseous fuel for domestic purpose
Fixed fire detection etc.
Fire protection arr. in cargo spaces \

```

```

( dann ( ->.fullf )
)

```

```

regel SOLASII2.PartC.Reg45to53.fullf fuer ship
\ Means of escape
Protection of stairways and lift trunks in accom.....
Doors in fire resisting divisions
Ventilation systems
Restricted use of combustibile materials
Details of construction
Arrangement for gaseous fuel for domestic purpose
Fixed fire detection etc.
Fire protection arr. in cargo spaces \

```

```
( frage \ Erfuelllt ? \
)
```

```
regel SOLASII2.PartC.Reg54 fuer ship
\ Special requirements for ships carrying dangerous goods \
```

```
( dann ( ->.fullf )
)
```

```
regel SOLASII2.PartC.Reg54.fullf fuer ship
\ Special requirements for ships carrying dangerous goods \
```

```
( frage \ Erfuelllt ? \
)
```

```
regel SOLASII2.PartC.Reg43 fuer ship
\ Bulkheads within the accomodation and service spaces\
```

```
( dann ( ->.x1 & ->.x2 & ->.x3 & ->.x4 )
)
```

```
regel SOLASII2.PartC.Reg43.x1 fuer ship
\ All bulkheads required to be "B" class division shall extend
from deck to deck and to shell or other boundaries, unless
continuous "B" class ceilings or linings are fitted on both
sides of the bulkhead in which case the bulkhead may terminate at the
continuous ceiling or lining \
```

```
( fueralle .ga=ga.surf \ alle definierten Flaechen \
  wenn ( .type == "bulkhead" \ ist Schott \
    & "B0" <= .insclass <= "B15" \ required ins. class is type "B" \
    & ->^.accosurf ) \ Grenzt an "accomodation" \
  dann ( fueralle .boundary \ Schleife ueber die Grenzen \
    dann ( fueralle ^^ .line=objekt.surf
      wenn ( ^^ .surf=objekt.type == "ceiling" )
        \ angrenzende Flaechen ist "ceiling" \
        dann ( ^^ .surf=objekt.insclass >= "B0" )
          \ Ceilings muessen Iso class Typ "B" \

      wenn ( ^^ .surf=objekt.type == "deck" ||
        ^^ .surf=objekt.type == "shell" )
        \ angrenzende Flaechen ist vom Typ "deck" oder "shell" \
        dann ( wahr ) \ keine weiteren Forderungen \

      wenn ( ^^ .surf=objekt.type == "bulkhead" )
```

```

        \ angrenzende Flaeche ist wieder ein Schott \
        dann ( wahr ) \ keine direkte Forderung \
    )
)

regel SOLASII2.PartC.Reg43.accosurf fuer ship.ga.surf
\ check, if surface meets accomodation or service space \

( einervon .panel
  dann ( ->^.accopanel ) \ surf meets accomodation or service space \
)

regel SOLASII2.PartC.Reg43.accopanel fuer ship.ga.surf.panel
\ check, if panel meets accomodation or service space \

( einervon .room
  wenn ( ->^^.PartA.Reg3.x10 ( ^^^.room=objekt ) |
        ->^^.PartA.Reg3.x11 ( ^^^.room=objekt ) )
  dann ( wahr ) \ panel to accomodation or service space \
)

regel SOLASII2.PartC.Reg43.x2 fuer ship
\Method IC - All bulkheads not required by this or
other regulations of this part to be A or B class
divisions, shall be of at least C class construction\

( wenn ( .main=main.add=add.firemethod == "IC" ) \ Method IC \
  dann ( fueralle .ga=ga.surf
    wenn ( ->^.accosurf )
      dann ( .insclass >= "C" ) \ required insulation C \
    )
)

regel SOLASII2.PartC.Reg43.x3 fuer ship
\Method IIC - There shall be no restrictions on the construction
of bulkheads not required by this or other regulations of this
part to be A or B class divisions except in individual cases
where C class bulkheads are required in accordance with
table 44.1.\

( wenn ( .main=main.add=add.firemethod == "IIC" ) \ Method IIC \
  dann ( wahr ) \ no additional requirements \
)

```

```

regel SOLASII2.PartC.Reg43.x4 fuer ship
\Method IIIC - There shall be no restr. ... except that
the area of any acc.space bounded by A or B ... must in no
case exceed 50 m**2 except in indiv. cases where C class
is req. Consideration may be given by the Administr.
to incr for public spaces.\

( wenn ( .main=main.add=add.firemethod == "IIIC" ) \ Method IIIC \
  dann ( fueralle .ga=ga.room \ alle Raeume \
    wenn ( ->^^.PartA.Reg3.x10 | ->^^.PartA.Reg3.x11 )
      \ accomodation or store \
    dann ( rechnen ( intglobal contclass ;
      ) \ definition fuer contclass \
      wenn ( ->.calc & contclass == 2 )
        \ continuous bulkhead typ A oder B \
      dann ( .groundarea <= 50 )
        \ ground area should not exceed 50 m**2 \

      wenn ( contclass == 1 ) \ cont class C \
      dann ( wenn ( .groundarea <= 50. )
        dann ( wahr )

        wenn ( default )
        dann ( wenn ( .solastype ==
          "PublicSpaceFurnishedRestrictedFireRisk " |
          .solastype ==
          "PublicSpaceFurnishedOtherThanRestrictedFireRisk" )
          \ Public space \
          dann ( ->.Frage )

          wenn ( default )
          dann ( falsch )
        )
      )
    )
  )
)

regel SOLASII2.PartC.Reg43.x4.calc fuer ship.ga.room
\ define if space bounded by continuous class "B" or "A" bulkheads (contclass = 2)
or bounded by continuous class "C" bulkheads (contclass = 1)\

( fueralle .surf
  rechnen ( intglobal contclass ;
    contclass = 2 ; ) \ contclass initialisieren \

```

```

wenn ( ^.surf=objekt.type == "bulkhead" ) \ contclass bestimmen \
dann ( wenn ( ^.surf=objekt.insclass >= "B0" )
      dann ( wahr )

      wenn ( ^.surf=objekt.insclass == "C" )
      dann ( rechnen ( contclass = min ( contclass, 1 ) ; ) )

      wenn ( ^.surf=objekt.insclass < "C" )
      dann ( rechnen ( contclass = 0 ; ) )
)
)

```

```

regel SOLASII2.PartC.Reg43.x4.Frage fuer ship.ga.room
\Consideration is given by the Administration for increasing
the Groundarea over 50 m**2\

```

```
( frage \ ok ? \ )
```

```

regel SOLASII2.PartC.Reg44 fuer ship
\Fire integrity for bulkheads and decks\

```

```
( dann ( ->.x2 & ->.x1 )
)
```

```

regel SOLASII2.PartC.Reg44.x1 fuer ship
\In addition to complying with the specific provisions
for fire integrity of bulkheads and decks mentioned
elsewhere in this part, the minimum fire integrity
of bulkheads and decks shall be as prescribed in
tables 44.1 and 44.2\

```

```
( fueralle .ga=ga.surf
  wenn ( .type == "bulkhead" ) \ fuer alle Schotte \
  dann ( ->^.Table44x1 ) \ Table 44.1 \

  wenn ( .type == "deck" ) \ fuer alle Decks \
  dann ( wahr ) \ & ->.Table44x2 ) # Table 44.2 \
)
```

```

regel SOLASII2.PartC.Reg44.x2 fuer ship
\ The following requirements shall govern application of the tables:
.1 Tables 44.1 and 44.2 shall apply respectively
to the bulkheads and decks separating adjacent spaces
.2 For determinig the appropiate fire integrity standards
to be appllied to divisions between adjacent spaces, such

```

spaces are classified according to their fire risk as shown in categories (1) to (11).

Classifying all spaces: \

```
( fueralle .ga=ga.room \for all rooms\
  wenn ( "tank" <= .type <= "superstructure" )
    dann ( wenn ( "EmergencySourcePower" <= .solastype
      <= "CentralizedEmergencyPublicAddressSystem" )
        dann ( rechnen ( .solascat = 1 ; ) \ category=1 \ )

    wenn ( ->^^^.PartA.Reg3.x10 )
      dann ( wenn ( "PassengerCorridor" <= .solastype <= "CrewLobby" )
        dann ( rechnen ( .solascat = 2 ; ) \ category = 2 \ )

        wenn ( default )
          dann ( rechnen ( .solascat = 3 ; ) \ category = 3 \ )
      )

    wenn ( "InteriorStairway" <= .solastype <= "Escalator" )
      dann ( rechnen ( .solascat = 4 ; ) \ category = 4 \ )

    wenn ( ->^^^.PartA.Reg3.x19 )
      dann ( rechnen ( .solascat = 6 ; ) \ category = 6 \ )

    wenn ( ->^^^.PartA.Reg3.x20 & ! ->^^^.PartA.Reg3.x19 )
      dann ( rechnen ( .solascat = 7 ; ) \ category = 7 \ )

    wenn ( .solastype >= "CargoOilTankOrServingTrunk"
      \ type >= Cargo Oil Tank \ &
      .solastype <= "CargoHatchwayOrServingTrunk"
      \ type <= Cargo Hatchway \ )
      dann ( rechnen ( .solascat = 8 ; ) \ category = 8 \ )

    wenn ( ->^^^.PartA.Reg3.x12 )
      dann ( wenn ( ( ( .solastype == "IsolatedLocker"
        || .solastype == "BaggageRoom"
        )
        & .groundarea < 2
        )
        || .solastype == "DryingRoom"
        || .solastype == "MainLaundry"
        )
      dann ( rechnen ( .solascat = 5 ; ) \ category = 5 \ )

    wenn ( default )
      dann ( rechnen ( .solascat = 9 ; ) \ category = 9 \ )
  )
```

```

wenn ( "LifeboatEmbarkationStation" <= .solastype <= "AirSpace" )
dann ( rechnen ( .solascat = 10 ; ) \ category = 10 \ )

wenn ( ->^^.PartA.Reg3.x14 )
dann ( rechnen ( .solascat = 11 ; ) \ category = 11 \ )

wenn ( default )
dann ( rechnen ( intlokal dummy ;
                dummy = .solascat ; )
        \Solas Category einlesen, weil nicht zuzuordnen\ )
)

wenn ( default ) \not a real room (subroom, hull or damage)\
dann ( rechnen ( .solascat = 0 ; ) \category = 0\ )
)

regel SOLASII2.PartC.Reg44.Table44x1 fuer ship.ga.surf
\ Table 44.1 \

( fueralle .panel
  dann ( fueralle .room \ alle Raeume, die an das Schott grenzen \
        rechnen ( intglobal c1, log1 ;
                  realglobal groundarea ;
                  c1 = ^^room=objekt.solascat ;
                  log1 = ^^room=objekt.solastype ;
                  groundarea = ^^room=objekt.groundarea ;
                ) \ Werte des ersten Raumes sichern \

  dann ( fueralle ^.room \ alle Raeume, die an das Schott grenzen \
        rechnen ( intglobal c2, log2 ;
                  c2 = ^^room=objekt.solascat ;
                  log2 = ^^room=objekt.solastype ;
                ) \ Werte des zweiten Raumes sichern \

  tabelle ( 12, c1 , 12, c2 ;

  (1,1) ^^insclass >= "A0" | ->^.noteE ( ^^ ) ;
        \required insulation A0 or note E\

  (1,2) ^^insclass >= "A0" ; \required insulation A0\
  (2,2) ^^insclass >= "C" ; \required insulation C\

  (1,3) ^^insclass >= "A60" ; \required insulation A60\
  (2,3) ^^insclass >= "B0" ; \required insulation B0\
  (3,3) ( ->^.noteA ( ^^ ) | ^^insclass >= "C" )
        & ->^.noteB ( ^^ ) ;

```

\required insulation C, note A and Note B\

```
(1,4) ^^\.insclass >= "A0" ; \required insulation A0\
(2,4) ->^.noteC ( ^^ ) ; \note C\
(3,4) ->^.noteC ( ^^ ) ; \note C\
(4,4) ->^.noteC ( ^^ ) ; \note C\
```

```
(1,5) ^^\.insclass >= "A15" ; \required insulation A15\
(2,5) ^^\.insclass >= "B0" ; \required insulation B0\
(3,5) ^^\.insclass >= "B0" ; \required insulation B0\
(4,5) ->^.noteC ( ^^ ) ; \note C\
(5,5) ^^\.insclass >= "C" ; \required insulation C\
```

```
(1,6) ^^\.insclass >= "A60" ; \required insulation A60\
(2,6) ^^\.insclass >= "A60" ; \required insulation A60\
(3,6) ^^\.insclass >= "A60" ; \required insulation A60\
(4,6) ^^\.insclass >= "A60" ; \required insulation A60\
(5,6) ^^\.insclass >= "A60" ; \required insulation A60\
(6,6) ->^.asterix ( ^^ ) ; \aterix\
```

```
(1,7) ^^\.insclass >= "A15" ; \required insulation A15\
(2,7) ^^\.insclass >= "A0" ; \required insulation A0\
(3,7) ^^\.insclass >= "A0" ; \required insulation A0\
(4,7) ^^\.insclass >= "A0" ; \required insulation A0\
(5,7) ^^\.insclass >= "A0" ; \required insulation A0\
(6,7) ^^\.insclass >= "A0" ; \required insulation A0\
(7,7) ^^\.insclass >= "A0" | ->^.noteD ( ^^ ) ;
      \required insulation A0 or note D\
```

```
(1,8) ^^\.insclass >= "A60" ; \required insulation A60\
(2,8) ^^\.insclass >= "A0" ; \required insulation A0\
(3,8) ^^\.insclass >= "A0" ; \required insulation A0\
(4,8) ^^\.insclass >= "A0" ; \required insulation A0\
(5,8) ^^\.insclass >= "A0" ; \required insulation A0\
(6,8) ->^.noteG ( ^^ ) ; \note G\
(7,8) ^^\.insclass >= "A0" ; \required insulation A0\
(8,8) ->^.asterix ( ^^ ) ; \asterix\
```

```
(1,9) ^^\.insclass >= "A60" ; \required insulation A60\
(2,9) ^^\.insclass >= "A0" ; \required insulation A0\
(3,9) ^^\.insclass >= "A0" ; \required insulation A0\
(4,9) ^^\.insclass >= "A0" ; \required insulation A0\
(5,9) ^^\.insclass >= "A0" ; \required insulation A0\
(6,9) ^^\.insclass >= "A60" ; \required insulation A60\
(7,9) ^^\.insclass >= "A0" ; \required insulation A0\
(8,9) ^^\.insclass >= "A0" ; \required insulation A0\
(9,9) ^^\.insclass >= "A0" | ->^.noteD ( ^^ ) ;
      \required insulation A0 or note D\
```

```

(1,10) ->^.asterix ( ^^ ) ; \asterix\
(2,10) ->^.asterix ( ^^ ) ; \asterix\
(3,10) ->^.asterix ( ^^ ) ; \asterix\
(4,10) ->^.asterix ( ^^ ) ; \asterix\
(5,10) ->^.asterix ( ^^ ) ; \asterix\
(6,10) ->^.asterix ( ^^ ) ; \asterix\
(7,10) ->^.asterix ( ^^ ) ; \asterix\
(8,10) ->^.asterix ( ^^ ) ; \asterix\
(9,10) ->^.asterix ( ^^ ) ; \asterix\

(1,11) ^^.insclass >= "A60" ; \required insulation A60\
(2,11) ^^.insclass >= "A30" ; \required insulation A30\
(3,11) ^^.insclass >= "A30" ; \required insulation A30\
(4,11) ^^.insclass >= "A30" ; \required insulation A30\
(5,11) ^^.insclass >= "A0" ; \required insulation A0\
(6,11) ^^.insclass >= "A60"
      | ( ^^.insclass >= "A0"
        & ->^.noteF ( ^^^.room=objekt ) ) ;
      \required insulation A60 or A0 and note F\
(7,11) ^^.insclass >= "A0" ; \required insulation A0\
(8,11) ^^.insclass >= "A0" ; \required insulation A0\
(9,11) ^^.insclass >= "A30" ; \required insulation A30\
(10,11) ^^.insclass >= "A0" ; \required insulation A0\
(11,11) ->^.asterix ( ^^ ) & ->^.noteH ( ^^ ) ;
        \asterix and note H\
      )
    )
  )
)

```

regel SOLASII2.PartC.Reg44.noteA fuer ship.ga.surf  
 \ No special requirements are imposed upon bulkheads in methods IIC and IIIC  
 fire protection\

```

( wenn ( "IIC" <= ^^.main=main.add=add.firemethod <= "IIIC" )
  dann ( wahr ) \no special requirements\
)

```

regel SOLASII2.PartC.Reg44.noteB fuer ship.ga.surf  
 \ In case of method IIIC "B" class bulkheads of "B-0" rating shall  
 be provided between spaces or groups of spaces of 50m<sup>2</sup> and over  
 in area \

```

( rechnen ( realglobal groundarea ; )

```

```

wenn ( ^^main=main.add=add.firemethod == "IIIC" &
      groundarea >= 50 ) \ WARNING: No check for spacegroups
                          with an area >= 50 m**2\
dann ( .insclass >= "B0" )
)

regel SOLASII2.PartC.Reg44.noteC fuer ship.ga.surf
\ For clarification as to which applies, see regulation 43 and 46 \

( dann ( wahr )
  \ Reg 43 and Reg 46 will be checked seperatly \
)

regel SOLASII2.PartC.Reg44.noteD fuer ship.ga.surf
\ Where spaces of the same numerical category and sperscript d appears,
a bulkhead or deck of the rating shown in the tables is only required
when the adjacent spaces are for a different purpose,
e.g. in category (9) a galley next to a galley does not require a bulkhead
but a galley next to a paint room requires an "A-0" bulkhead \

( rechnen ( intglobal log1, log2 ; )

  wenn ( log1 == log2 ) \ rooms for same purpose \
  dann ( wahr ) \ no requirements \

  wenn ( default ) \ rooms for different purpose \
  dann ( falsch ) \ required insulation acc. to table \
)

regel SOLASII2.PartC.Reg44.noteE fuer ship.ga.surf
\ Bulkheads separating the wheelhouse, chartroom and radio room from
each other any ve "B-0" rating \

( rechnen ( intglobal log1, log2 ; )

  wenn ( 2 <= log1 <= 4 & 2 <= log2 <= 4 )
  dann ( .insclass >= "B0" ) \ required insulation class B0\
)

regel SOLASII2.PartC.Reg44.noteF fuer ship.ga.room
\ A-0 rating may be used if no dangerous goods are intended to be
carried or if such goods are stowed not less than 3m horizontally
from such bulkhead \

```

```
( frage \ Are all dangerous goods stowed more than 3m
    away from the bulkhead? \
)
```

```
regel SOLASII2.PartC.Reg44.noteG fuer ship.ga.surf
\ For cargo spaces in which dangerous goods are intended to be
carried, regulation 54.2.8 applies. \
```

```
( dann ( wahr ) \ Reg 54.2.8 will be checked seperatly \
)
```

```
regel SOLASII2.PartC.Reg44.noteH fuer ship.ga.surf
\ Bulkheads and decks separating ro-ro cargo spaces shall capable
of being closed rsponsably gastight and such diivisions shall
have "A" class integrity in so far as is reasonable and practicable
in the opinion of the administration \
```

```
( wenn ( .gastight == "yes" & .insclass >= "A0" )
    dann ( wahr ) \ bulkhead is gastight and "A" class integrity \

    wenn ( default )
    dann ( ->.except )
)
```

```
regel SOLASII2.PartC.Reg44.noteH.except fuer ship.ga.surf
\ Bulkheads and decks separating ro-ro cargo spaces shall capable
of being closed rsponsably gastight and such diivisions shall
have "A" class integrity in so far as is reasonable and practicable
in the opinion of the administration \
```

```
( frage \ Exception ? \ )
```

```
regel SOLASII2.PartC.Reg44.noteI fuer ship.ga.room
\ Fire insulation need not be fitted if the machinery space in
category (7), in the opinion of the adiministration, has little or
no fire risk. \
```

```
( frage \ Exception ? \ )
```

```
regel SOLASII2.PartC.Reg44.asterix fuer ship.ga.surf
\ Where an asterix appears in the tables, the division is required
to be of steel or other equivalent material but is not required
to be of "A" class standard \
```

```
( dann ( .material == "steel" )  
)
```

## C.3.2 Datenmodell

Gekürztes Datenmodell fuer Regel SOLAS83II2

Datenmodelldatei : e:\uni\promo\zzzdat\ship Thu Mar 12 10:28:54 1992

Beginn : file=e:\uni\promo\zzzdat\solasii2

- k = startknoten

```

1 1      - ship , k = Ships Id (No. of Ship)
2 7      - regel , r = Regulations
2 6      - mach , k = Machinery (= mach)
          ... (gekuerzt)
2 5      - repro , k = Resistance and Propusion (= repro)
          ... (gekuerzt)
2 4      - weight , k = Leight Ship Weight Calculation (= weight)
          ... (gekuerzt)
2 3      - stab , k = Loading Cases and Stability (= stab)
          ... (gekuerzt)
2 2      - ga , k = General Arrangement (= ga) Definition of Decks,
          Bulkheads, Volumes, Supersrtructure, ...
3 8      - regel , r = regulations
3 7      - superstr , k = Superstructures, everything above Maindeck
          (Superstr. Id, defined Ids: Hatches, Back, Poop)
4 9      - cb , p = cb of Superstr. (cb = Vol / L / B / H) : - 1.000000
4 8      - Bredth , p = max. Bredth of Volume : m 1.000000
4 7      - Length , p = max. Length of Volume : m 1.000000
4 6      - HaBase , p = Height above Baseline : m 1.000000
4 5      - Height , p = max Height above Maindeck : m 1.000000
4 4      - ADeck , p = Total Groundarea of Superstr. : m^2 1.000000
4 3      - AGround , p = Gesamt Grundflaechе des Aufbaus :
          m^2 1.000000
4 2      - Vol , p = Total Volume of Superstr : m^3 1.000000
4 1      - deck , k = Decks or Parts of a Superstr. (Deck Id)
5 11     - DeckBelow , k = Deck Id of Deck below this, just allowed once
5 10     - cb , p = cb of Deck/Part Volume (e.g. = 1) : - 1.000000
5 9      - Breadth , p = max. Breadth of Deck/Part : m 1.000000
5 8      - Length , p = Max. Length of Deck/Part : m 1.000000
5 7      - DeckAbove , k = Name of Deck above this, just allowed once
5 6      - ADeck , p = Decksarea : m^2 1.000000
5 5      - AAbove , p = Area of Deck above : m^2 1.000000
5 4      - AGround , p = Groundarea of Deck : m^2 1.000000
5 3      - Vol , p = Volume of Deck/Part : m^3 1.000000
5 2      - Height , p = Height to next Deck : m 1.000000
5 1      - HaBase , p = Height of Deck above Baseline : m 1.000000
3 6      - fr , k = Building Frames (Frame Id)

```

```

4 1      - x , p = X-Koordinate : m 1.000000
3 5      - open , k = Opening Lists and Freeboard Lines (Open Id)
          Standard Ids are: side -> side of deck,
          marg -> Margin Line,
          watert -> watertight Openings,
          weathert -> weathertight Openings,
          prog -> Progressive Flooding will take place
4 3      - item , k = Item of List (Item Id)
5 5      - z , p = X-Koo of lowest Edge : m 1.000000
5 4      - y , p = X-Koo of lowest Edge : m 1.000000
5 3      - x , p = X-Koo of lowest Edge : m 1.000000
5 2      - connect , k = Rooms connected by this Opening
          (= sea or room Id)
4 2      - type , a = Type of Opening List/ Freeboard Line : 5
          SideofDeck
          MarginLine
          Watertight
          Weathertight
          ProgFlood
4 1      - des , s = Description
3 4      - line , k = Line to define panels (Line Id)
4 4      - surf , k = Reference to surf, which uses this line (Surf Id)
5 1      - panel , k = Reference to defined panel (Panel Id)
4 3      - x2 , v = Vector containing x,y,z for point, defining the line :
          m 1.000000
4 2      - x1 , v = Vector containing x,y,z for point, defining the line :
          m 1.000000
4 1      - des , s = Description
3 2      - surf , k = Surface as a List of Panels (Surf Id),
          surfaces should be of continuous material,
          isolation etc.
4 10     - gastight , a = gastightness : 2
          no
          yes
4 9      - material , a = material : 2
          steel
          aluminium
4 8      - regel , r = regulations
4 7      - insclass , a = division Class of surface : 8
          noiso
          C
          B0
          B15
          A0
          A15
          A30
          A60
4 6      - defpanel , k = Definition panel for generating the panels

```

```

                                of this surf (Defpanel Id)
5 7      - surf , k = surf, that limits this defpanel (Surf Id)
6 1      - pos , a = defpanel lies relativ to this surf : 2
          FwdSsAbove
          AftPsBelow
5 6      - x4 , v = point, defining this defpanel : m 1.000000
5 5      - x3 , v = point, defining this defpanel : m 1.000000
5 4      - x2 , v = point, defining this defpanel : m 1.000000
5 3      - x1 , v = point, defining this defpanel : m 1.000000
5 2      - type , a = Type of def Panel : 2
          solid
          hole
5 1      - des , s = Description
4 5      - boundary , k = all lines, building the boundary of the
          surface (Line Id)
4 4      - panel , k = panel, that decribes the surface (Panel Id)
5 5      - area , p = Area of panel : m^2 1.000000
5 2      - room , k = room that is defined by this panel (Room Id)
5 1      - line , k = Line, that defines the panel (Line Id)
4 3      - type , a = type of surface : 4
          shell
          bulkhead
          deck
          ceiling
4 2      - orient , a = main orientation of surface : 3
          transbh
          longbh
          deck
4 1      - des , s = Description
3 1      - room , k = Closed Rooms, Volumes (Room Id).
          For the room, forming the Bouancy for intact vessel
          -> Room Id = hull
4 13     - regel , r = regulations
4 12     - groundarea , p = groundarea of room : m^2 1.000000
4 11     - instpower , p = installed engine power in the room :
          kW 1.000000
4 10     - sub , k = Sub-Room, forming this room (Room Id)
5 1      - add , i = sub-room to be add (+1) or tob substracted (-1)
4 9      - surf , k = limiting Surface (surf Id)
5 2      - pos , a = room lies relative to surf : 2
          FwdSsAbove
          AftPsBelow
5 1      - panel , k = limiting panel of this surf (Panel Id)
4 8      - hydro , k = Hydrostatic Particulars as a table(Table Id)
5 18     - heel , v = Heeling Angles : Deg 57.300000
5 17     - trim , v = Trims corresponding to Heels : m 1.000000
5 16     - draught , v = Draughts corresponding to Heels : m 1.000000
5 15     - Surf , v = wetted Surface : m^2 1.000000

```

```

5 14      - Ixy , v = moment of Inertia xy : m^4 1.000000
5 13      - Iy , v = moment of Inertia y : m^4 1.000000
5 12      - Ix , v = moment of Inertia x : m^4 1.000000
5 11      - zw , v = z Waterline : m 1.000000
5 10      - yw , v = y Waterline : m 1.000000
5 9       - xw , v = x Waterline : m 1.000000
5 8       - Aw , v = Area Waterline : m^2 1.000000
5 7       - zb , v = z bouyancy : m 1.000000
5 6       - yb , v = y bouancy : m 1.000000
5 5       - xb , v = x bouancy : m 1.000000
5 4       - Vol , v = netto volume : m^3 1.000000
5 3       - ullage , v = actual ullage acc to filling : m 1.000000
5 2       - sound , v = actual sounding acc to filling : m 1.000000
4 6       - perm , p = Permeability for Damage Calculations
           (just for rooms type != subpart) : - 1.000000
4 5       - internal , p = Correction Factor for Internals
           (just for rooms type != subpart) : - 1.000000
4 4       - solascat , i = Category According to SOLAS II-2 Reg 26,44,58
4 3       - solastype , a = Type of Room acc. to SOLAS 3,26,44,58 : 106
           EmergencySourcePower
           EmergencySourceLight
           Wheelhouse
           Chartroom
           Radioroom
           FireExtinguishingRoom
           FireControlRoom
           FireRecordingStation
           ControlRoomPropulsionMachinery
           CentralizedFireAlarmEquipment
           CentralizedEmergencyPublicAddressSystem
           InteriorStairway
           Lift
           Escalator
           PassengerCorridor
           CrewCorridor
           PassengerLobby
           CrewLobby
           LifeboatEmbarkationStation
           LifeboatLoweringStation
           LiferaftEmbarkationStation
           LiferaftLoweringStation
           OpenDeckSpace
           EnclosedPromenade
           AirSpace
           CabinFurnishedRestrictedFireRisk
           PublicSpaceFurnishedRestrictedFireRisk
           CabinFurnishedOtherThanRestrictedFireRisk
           PublicSpaceFurnishedOtherThanRestrictedFireRisk

```

IsolatedLocker  
SmallStoreRoomInAccommodation  
SaleShop  
MotionPictureProjectionRoom  
FilmStowageRoom  
DietKitchenNoOpenFlame  
CleaningGearLockerNoFlammableLiquid  
LaboratoriesNoFlammableLiquid  
Pharmacie  
DryingRoom  
SpecieRoom  
BarberShop  
BeautyParlour  
CommunalSanitary  
Shower  
Bath  
WaterCloset  
SmallLaundryRoom  
IndoorSwimmingPool  
OperatingRoom  
IsolatedPantryNoCookingAppliances  
WaterTankFormingPartOfShipsStuktur  
VoidSpace  
Cofferdam  
VentilationRoom  
AirconditioningRoom  
WindlassRoom  
SteeringGearRoom  
StabilizerRoom  
ElectricalPropulsionMotorRoom  
SectionSwitchboardAndPurleyElectricalEquipmentNoOilFilled-  
TransformersAbove10kVA  
ShaftAlley  
PipeTunnel  
RefrigerationMachineryNotHandlingFlammableLiquids  
ClosedTrunk  
PipeTrunk  
CableTrunk  
CargoOilTankOrServingTrunk  
CargoHoldOrServingTrunk  
CargoHoldForDangerousGoodsOrServingTrunk  
CargoTrunkwayOrServingTrunk  
CargoHatchwayOrServingTrunk  
CargoHatchwayForDangerousGoodsOrServingTrunk  
OpenRoRoCargoSpace  
ClosedRoRoCargoSpace  
RefrigeratedChambersOrServingTrunk  
OilFuelTankOrServingTrunk

```

ShaftAlleyAndStorageOfCombustiblesOrServingTrunk
PipeTunnelAndStorageOfCombustiblesOrServingTrunk
OilFuelFillingStationOrServingTrunk
OilFilledElectricalTransformerAbove10kVAOrServingTrunk
TurbineDrivenAuxiliaryGeneratorRoomOrServingTrunk
ReciprocatingSteamEngineDrivenAuxiliaryGenRoomOrServingTrunk
InternalCombustionEngineDrivingEmergencyGenRoomOrServingTrunk
InternalCombustionEngineDrivingSprinklerPumpRoomOrServingTrunk
InternalCombustionEngineDrivingDrencherPumpRoomOrServingTrunk
InternalCombustionEngineDrivingFirePumpRoomOrServingTrunk
InternalCombustionEngineDrivingBilgePumpRoomOrServingTrunk
SpecialCategorySpaceOrServingTrunk
MainPropulsionMachineryRoom
MainPropulsionMachineryCasing
BoilerRoom
BoilerCasing
AuxiliaryMachineryRoom
MainGalley
MainGalleyAnnex
MainPantry
MainLaundry
MiscellaneousStore
MailRoom
BaggageRoom
GarbageRoom
Workshop
LampRoom
PaintRoom
StoreRoomContainingFlammableLiquids
Laboratories
4 2 - type , a = Type of Room : 11
      subroom
      tank
      cargo
      void
      cofferdam
      accomodation
      machinery
      outside
      superstructure
      hull
      damage
4 1 - des , s = Description
2 1 - main , k = Main Characteristics (=main)
      ... (gekuerzt)

```

## C.3.3 Daten

Wertedatei : e:\uni\promo\zzzdat\solasii2 Tue Jun 16 10:23:45 1992

Datenmodelldatei : e:\uni\promo\zzzdat\ship Tue Jun 16 09:12:23 1992

Beginn : file=e:\uni\promo\zzzdat\solasii2

```

- k = startknoten

1   - ship = B227
2   - main = main
3   - tsysys = 240 , 240
3   - brt = 999.900000 - , 999.900000 -
3   - typ = GeneralCargoContainer
3   - keeldate = 920303 , 920303
3   - add = add
4   - firemethod = IC
2   - ga = ga
3   - line = h1m6
4   - surf = lsm1
3   - line = h1s6
4   - surf = lsm1
3   - line = hms7
4   - surf = lsm2
3   - line = h1s7
4   - surf = lsm2
3   - line = sms1
4   - surf = qam2
4   - surf = qam1
4   - surf = lsm1
3   - line = sms2
4   - surf = lsm2
4   - surf = lsm1
4   - surf = qcm1
3   - line = sms3
4   - surf = lsm2
4   - surf = qcm2
3   - surf = lsm2
4   - type = bulkhead
4   - boundary = sms3
4   - boundary = hms7
4   - boundary = h1s7
4   - boundary = sms2
4   - insclass = A60
4   - panel = p1
5   - room = corridor
5   - room = staircase
3   - surf = lsm1

```

```
4      - material = steel
4      - type = bulkhead
4      - boundary = sms2
4      - boundary = hms6
4      - boundary = h1s6
4      - boundary = sms1
4      - insclass = A60
4      - panel = p1
5      - room = corridor
5      - room = enginecasing
3  - line = smp3
4      - surf = lpm2
4      - surf = qcm2
3  - line = smp2
4      - surf = qcm1
4      - surf = qpm1
4      - surf = lpm1
4      - surf = lpm2
3  - line = hms25
4      - surf = lpm2
3  - line = h1s25
4      - surf = lpm2
3  - line = hms10
4      - surf = lsm4
3  - surf = lsm3
4      - insclass = A0
4      - panel = p1
5      - room = corridor
5      - room = emergency
4      - type = bulkhead
3  - line = sms5
4      - surf = qsm1
4      - surf = lsm3
4      - surf = lsm4
3  - line = h1s15
4      - surf = qcm2
3  - line = hms15
4      - surf = qcm2
3  - line = h1s3
4      - surf = qcm2
3  - line = hms3
4      - surf = qcm2
3  - line = hmp23
4      - surf = qcm2
3  - line = h1p23
4      - surf = qcm2
3  - line = smp6
4      - surf = lpm
```

```
4      - surf = qcm2
3      - line = sms11
4      - surf = lsm
4      - surf = qcm2
3      - line = hms4
4      - surf = qcm3
3      - line = sms12
4      - surf = lsm
4      - surf = qcm3
3      - line = h1s4
4      - surf = qcm3
3      - line = smp7
4      - surf = lpm
4      - surf = qcm3
3      - room = opendeck
4      - solascat = 10 , 10
4      - groundarea = 0.000000 m^2 , 0.000000 m^2
4      - instpower = 0.000000 kW , 0.000000 kW
4      - solastype = OpenDeckSpace
4      - type = outside
3      - surf = lpm2
4      - boundary = h1s25
4      - boundary = hms25
4      - boundary = smp2
4      - boundary = smp3
4      - insclass = B15
4      - panel = p1
5      - room = staircase
5      - room = galley
4      - type = bulkhead
3      - surf = lpm1
4      - insclass = A0
4      - panel = p1
5      - room = refrigerated
5      - room = galley
4      - type = bulkhead
3      - surf = lsm
4      - insclass = noiso
4      - material = steel
4      - type = bulkhead
4      - panel = p5
5      - room = opendeck
5      - room = mess
4      - panel = p4
5      - room = opendeck
5      - room = corridor
4      - panel = p3
5      - room = opendeck
```

```
5         - room = crew
4         - panel = p2
5         - room = opendeck
5         - room = laundry
4         - panel = p1
5         - room = opendeck
5         - room = emergency
3     - surf = lpm
4         - insclass = noiso
4         - material = steel
4         - type = bulkhead
4         - panel = p4
5         - room = opendeck
5         - room = mess
4         - panel = p3
5         - room = opendeck
5         - room = corridor
4         - panel = p2
5         - room = opendeck
5         - room = galley
4         - panel = p1
5         - room = opendeck
5         - room = refrigerated
3     - surf = qam3
4         - insclass = noiso
4         - material = steel
4         - panel = p1
5         - room = opendeck
5         - room = refrigerated
4         - type = bulkhead
3     - surf = qam2
4         - insclass = noiso
4         - material = steel
4         - panel = p1
5         - room = opendeck
5         - room = enginecasing
4         - type = bulkhead
3     - surf = qam1
4         - insclass = noiso
4         - material = steel
4         - type = bulkhead
4         - panel = p2
5         - room = opendeck
5         - room = corridor
4         - panel = p1
5         - room = opendeck
5         - room = emergency
3     - surf = qfm
```

```
4      - insclass = noiso
4      - material = steel
4      - panel = p1
5      - room = opendeck
5      - room = mess
4      - type = bulkhead
3  - surf = qpm1
4      - insclass = A0
4      - panel = p1
5      - room = refrigerated
5      - room = galley
4      - type = bulkhead
3  - surf = qcm3
4      - boundary = smp7
4      - boundary = h1s4
4      - boundary = sms12
4      - boundary = hms4
4      - insclass = B0
4      - panel = p1
5      - room = corridor
5      - room = mess
4      - type = bulkhead
3  - surf = qcm2
4      - boundary = sms11
4      - boundary = smp6
4      - boundary = h1p23
4      - boundary = hmp23
4      - boundary = hms3
4      - boundary = h1s3
4      - boundary = hms15
4      - boundary = h1s15
4      - insclass = B0
4      - type = bulkhead
4      - panel = p3
5      - room = corridor
5      - room = crew
4      - panel = p2
5      - room = corridor
5      - room = staircase
4      - panel = p1
5      - room = corridor
5      - room = galley
3  - surf = qcm1
4      - material = steel
4      - insclass = A60
4      - panel = p1
5      - room = staircase
5      - room = enginecasing
```

```

4       - type = bulkhead
3     - surf = qsm1
4       - insclass = A60
4       - panel = p1
5         - room = laundry
5         - room = emergency
4       - type = bulkhead
3     - room = refrigerated
4       - solascat = 9 , 9
4       - groundarea = 8.000000 m^2 , 8.000000 m^2
4       - instpower = 0.000000 kW , 0.000000 kW
4       - solastype = RefrigeratedChambersOrServingTrunk
4       - type = machinery
3     - room = galley
4       - solascat = 9 , 9
4       - groundarea = 16.000000 m^2 , 16.000000 m^2
4       - instpower = 0.000000 kW , 0.000000 kW
4       - solastype = MainGalley
4       - type = accomodation
3     - room = staircase
4       - groundarea = 8.000000 m^2 , 8.000000 m^2
4       - solascat = 4 , 4
4       - instpower = 0.000000 kW , 0.000000 kW
4       - solastype = InteriorStairway
4       - type = accomodation
3     - room = mess
4       - solascat = 3 , 3
4       - groundarea = 30.000000 m^2 , 30.000000 m^2
4       - instpower = 0.000000 kW , 0.000000 kW
4       - solastype = PublicSpaceFurnishedRestrictedFireRisk
4       - type = accomodation
3     - room = enginecasing
4       - solascat = 6 , 6
4       - groundarea = 4.000000 m^2 , 4.000000 m^2
4       - instpower = 0.000000 kW , 0.000000 kW
4       - solastype = MainPropulsionMachineryCasing
4       - type = machinery
3     - room = emergency
4       - solascat = 1 , 1
4       - groundarea = 6.000000 m^2 , 6.000000 m^2
4       - instpower = 100.000000 kW , 100.000000 kW
4       - solastype = EmergencySourcePower
4       - type = machinery
3     - line = hms11
4       - surf = lsm4
3     - line = sms7
4       - surf = qcm2
4       - surf = lsm4

```

```
3      - line = h1s11
4      - surf = lsm4
3      - line = h1s10
4      - surf = lsm4
3      - line = hms14
4      - surf = qsm2
3      - line = sms6
4      - surf = lsm4
4      - surf = qsm2
3      - line = h1s14
4      - surf = qsm2
3      - line = sms10
4      - surf = lsm
4      - surf = qsm2
3      - surf = qsm2
4      - boundary = sms10
4      - boundary = h1s14
4      - boundary = sms6
4      - boundary = hms14
4      - insclass = B0
4      - panel = p1
5      - room = crew
5      - room = laundry
4      - type = bulkhead
3      - surf = lsm4
4      - boundary = sms5
4      - boundary = h1s10
4      - boundary = h1s11
4      - boundary = sms7
4      - boundary = hms11
4      - boundary = hms10
4      - insclass = B0
4      - panel = p2
5      - room = corridor
5      - room = crew
4      - panel = p1
5      - room = corridor
5      - room = laundry
4      - type = bulkhead
3      - room = corridor
4      - solascat = 2 , 2
4      - instpower = 0.000000 kW , 0.000000 kW
4      - groundarea = 17.000000 m^2 , 17.000000 m^2
4      - solastype = CrewCorridor
4      - type = accomodation
3      - room = crew
4      - solascat = 3 , 3
4      - instpower = 0.000000 kW , 0.000000 kW
```

```
4         - groundarea = 6.000000 m^2 , 6.000000 m^2
4         - solastype = CabinFurnishedRestrictedFireRisk
4         - type = accomodation
3     - room = laundry
4         - solascat = 5 , 5
4         - instpower = 0.000000 kW , 0.000000 kW
4         - groundarea = 6.000000 m^2 , 6.000000 m^2
4         - solastype = MainLaundry
4         - type = accomodation
```

## C.3.4 Prüfen, tabellarische Ausgabe, kurz

```
+-----+
| Pruefen |
+-----+
```

Wahrscheinlichkeit, dass aus  
'SOLASII-2'  
folgende Regel erfuehlt ist :

Regel 'SOLASII2'  
fuer 'ship=B227' aus e:\uni\promo\zzzdat\solasii2

| Ebene | Regel   | angewendet auf | er- gibt |
|-------|---|----------------|----------|
| 0     | SOLASII2<br>  SOLAS '74 incl. Ammendments up to '83<br>  Chapter II-2<br>  Construction - Fire Protection, Fire Detection and Fire Extinction   | ship=B227      | 0.000    |
| 1     | SOLASII2.PartA<br>  General   | ship=B227      | 0.000    |
| 2     | SOLASII2.PartA.Reg1<br>  Application  | ship=B227      | 0.000    |
| 3     | SOLASII2.PartA.Reg1.x1<br>  1.1 Unless expressedly provided otherwise, this chapter<br>  shall apply to ships the keel of which are laid or which<br>  are at a similar stage of construction on or after<br>  1.July 1986. | ship=B227      | 0.000    |
| 4     | SOLASII2.PartA.Reg1.appl<br>  Application of the Parts B, C, D  | ship=B227      | 0.000    |
| 5     | SOLASII2.PartC<br>  Fire Safety Measures for Cargo Ships  | ship=B227      | 0.000    |
| 6     | SOLASII2.PartC.Reg43<br>  Bulkheads within the accomodation and service spaces  | ship=B227      | 0.000    |
| 7     | SOLASII2.PartC.Reg43.x2<br>  Method IC - All bulkheads not required by this or<br>  other regulations of this part to be A or B class<br>  divisions, shall be of at least C class construction                             | ship=B227      | 0.000    |

|  |                                 |       |
|--|---------------------------------|-------|
| innere Regel 0   | ship=B227                       | 0.000 |
| -fuer alle Schleife  | .ga=ga.surf                     |       |
| 0. Teilbereich   | ship=B227.ga=ga.surf=lsm2       | 1.000 |
| 1. Teilbereich   | ship=B227.ga=ga.surf=lsm1       | 1.000 |
| 2. Teilbereich   | ship=B227.ga=ga.surf=lsm3       | 1.000 |
| 5. Teilbereich   | ship=B227.ga=ga.surf=lsm        | 0.000 |
| 6. Teilbereich   | ship=B227.ga=ga.surf=lpm        | 0.000 |
| 9. Teilbereich   | ship=B227.ga=ga.surf=qam1       | 0.000 |
| 10. Teilbereich  | ship=B227.ga=ga.surf=qfm        | 0.000 |
| 12. Teilbereich  | ship=B227.ga=ga.surf=qcm3       | 1.000 |
| 13. Teilbereich  | ship=B227.ga=ga.surf=qcm2       | 1.000 |
| 16. Teilbereich  | ship=B227.ga=ga.surf=qsm2       | 1.000 |
| 17. Teilbereich  | ship=B227.ga=ga.surf=lsm4       | 1.000 |
| -----  |                                 |       |
| 8 SOLASII2.PartC.Reg43.accosurf                                | ship=B227.ga=ga.surf=lpm2       | 0.000 |
| check, if surface meets accomodation or service space          |                                 |       |
| -einer von Schleife  | .panel                          |       |
| 0. Teilbereich   | p=B227.ga=ga.surf=lpm2.panel=p1 | 0.000 |
| -----  |                                 |       |
| 9 SOLASII2.PartC.Reg43.accopanel                               | p=B227.ga=ga.surf=lpm2.panel=p1 | 0.000 |
| check, if panel meets accomodation or service space            |                                 |       |
| -einer von Schleife  | .room                           |       |
| -----  |                                 |       |
| 10 SOLASII2.PartA.Reg3.x10                                     | ship=B227.ga=ga.room=staircase  | 0.000 |
| Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
| public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |
| cinemas, games and hobbies rooms, barber shops, pantries       |                                 |       |
| containing no cooking appliances and similar spaces            |                                 |       |
| -----  |                                 |       |
| 10 SOLASII2.PartA.Reg3.x11                                     | ship=B227.ga=ga.room=staircase  | 0.000 |
| Def. 11: Public spaces are those portions of the               |                                 |       |
| accomodation which are used for halls, dining rooms,           |                                 |       |
| lounges and similar permanently enclosed spaces.               |                                 |       |
| -----  |                                 |       |
| 10 SOLASII2.PartA.Reg3.x10                                     | ship=B227.ga=ga.room=galley     | 0.000 |
| Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
| public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |
| cinemas, games and hobbies rooms, barber shops, pantries       |                                 |       |
| containing no cooking appliances and similar spaces            |                                 |       |
| -----  |                                 |       |
| 10 SOLASII2.PartA.Reg3.x11                                     | ship=B227.ga=ga.room=galley     | 0.000 |
| Def. 11: Public spaces are those portions of the               |                                 |       |
| accomodation which are used for halls, dining rooms,           |                                 |       |
| lounges and similar permanently enclosed spaces.               |                                 |       |
| -----  |                                 |       |
| 8 SOLASII2.PartC.Reg43.accosurf                                | ship=B227.ga=ga.surf=lpm1       | 0.000 |
| check, if surface meets accomodation or service space          |                                 |       |
| -einer von Schleife  | .panel                          |       |

```

| 0. Teilbereich |p=B227.ga=ga.surf=lpm1.panel=p1|0.000|
-----
9|SOLASII2.PartC.Reg43.accopanel |p=B227.ga=ga.surf=lpm1.panel=p1|0.000|
| check, if panel meets accomodation or service space |
| -einer von Schleife |.room | |
-----
10|SOLASII2.PartA.Reg3.x10 |ip=B227.ga=ga.room=refrigerated|0.000|
| Def. 10: Accomodation spaces are those spaces used for |
| public spaces, corridors, lvatories,cabins,offices, hospitals, |
| cinemas, games and hobbies rooms, barber shops, pantries |
| containing no cooking appliances and similar spaces |
-----
10|SOLASII2.PartA.Reg3.x11 |ip=B227.ga=ga.room=refrigerated|0.000|
| Def. 11: Public spaces are those portions of the |
| accomodation which are used for halls, dining rooms, |
| lounges and similar permanently enclosed spaces. |
-----
10|SOLASII2.PartA.Reg3.x10 |ship=B227.ga=ga.room=galley |0.000|
| Def. 10: Accomodation spaces are those spaces used for |
| public spaces, corridors, lvatories,cabins,offices, hospitals, |
| cinemas, games and hobbies rooms, barber shops, pantries |
| containing no cooking appliances and similar spaces |
-----
10|SOLASII2.PartA.Reg3.x11 |ship=B227.ga=ga.room=galley |0.000|
| Def. 11: Public spaces are those portions of the |
| accomodation which are used for halls, dining rooms, |
| lounges and similar permanently enclosed spaces. |
-----
8|SOLASII2.PartC.Reg43.accosurf |ship=B227.ga=ga.surf=qam3 |0.000|
| check, if surface meets accomodation or service space |
| -einer von Schleife |.panel | |
| 0. Teilbereich |p=B227.ga=ga.surf=qam3.panel=p1|0.000|
-----
9|SOLASII2.PartC.Reg43.accopanel |p=B227.ga=ga.surf=qam3.panel=p1|0.000|
| check, if panel meets accomodation or service space |
| -einer von Schleife |.room | |
-----
10|SOLASII2.PartA.Reg3.x10 |ship=B227.ga=ga.room=opendeck |0.000|
| Def. 10: Accomodation spaces are those spaces used for |
| public spaces, corridors, lvatories,cabins,offices, hospitals, |
| cinemas, games and hobbies rooms, barber shops, pantries |
| containing no cooking appliances and similar spaces |
-----
10|SOLASII2.PartA.Reg3.x11 |ship=B227.ga=ga.room=opendeck |0.000|
| Def. 11: Public spaces are those portions of the |
| accomodation which are used for halls, dining rooms, |
| lounges and similar permanently enclosed spaces. |
-----

```

```

10|SOLASII2.PartA.Reg3.x10      |ip=B227.ga=ga.room=refrigerated|0.000|
|  Def. 10: Accomodation spaces are those spaces used for      |
|  public spaces, corridors, lvatories,cabins,offices, hospitals, |
|  cinemas, games and hobbies rooms, barber shops, pantries   |
|  containing no cooking appliances and similar spaces         |
-----
10|SOLASII2.PartA.Reg3.x11      |ip=B227.ga=ga.room=refrigerated|0.000|
|  Def. 11: Public spaces are those portions of the            |
|  accomodation which are used for halls, dining rooms,        |
|  lounges and similar permanently enclosed spaces.            |
-----
8|SOLASII2.PartC.Reg43.accosurf |ship=B227.ga=ga.surf=qam2      |0.000|
|  check, if surface meets accomodation or service space      |
|  -einer von Schleife          |.panel                          |      |
|  0. Teilbereich              |p=B227.ga=ga.surf=qam2.panel=p1|0.000|
-----
9|SOLASII2.PartC.Reg43.accopanel |p=B227.ga=ga.surf=qam2.panel=p1|0.000|
|  check, if panel meets accomodation or service space        |
|  -einer von Schleife          |.room                            |      |
-----
10|SOLASII2.PartA.Reg3.x10      |ship=B227.ga=ga.room=opendeck  |0.000|
|  Def. 10: Accomodation spaces are those spaces used for      |
|  public spaces, corridors, lvatories,cabins,offices, hospitals, |
|  cinemas, games and hobbies rooms, barber shops, pantries   |
|  containing no cooking appliances and similar spaces         |
-----
10|SOLASII2.PartA.Reg3.x11      |ship=B227.ga=ga.room=opendeck  |0.000|
|  Def. 11: Public spaces are those portions of the            |
|  accomodation which are used for halls, dining rooms,        |
|  lounges and similar permanently enclosed spaces.            |
-----
10|SOLASII2.PartA.Reg3.x10      |ip=B227.ga=ga.room=enginecasing|0.000|
|  Def. 10: Accomodation spaces are those spaces used for      |
|  public spaces, corridors, lvatories,cabins,offices, hospitals, |
|  cinemas, games and hobbies rooms, barber shops, pantries   |
|  containing no cooking appliances and similar spaces         |
-----
10|SOLASII2.PartA.Reg3.x11      |ip=B227.ga=ga.room=enginecasing|0.000|
|  Def. 11: Public spaces are those portions of the            |
|  accomodation which are used for halls, dining rooms,        |
|  lounges and similar permanently enclosed spaces.            |
-----
8|SOLASII2.PartC.Reg43.accosurf |ship=B227.ga=ga.surf=qpm1      |0.000|
|  check, if surface meets accomodation or service space      |
|  -einer von Schleife          |.panel                          |      |
|  0. Teilbereich              |p=B227.ga=ga.surf=qpm1.panel=p1|0.000|
-----
9|SOLASII2.PartC.Reg43.accopanel |p=B227.ga=ga.surf=qpm1.panel=p1|0.000|

```

|       |  |                                 |       |
|-------|--|---------------------------------|-------|
|       | check, if panel meets accomodation or service space            |                                 |       |
|       | -einer von Schleife  | .room                           |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x10  | ip=B227.ga=ga.room=refrigerated | 0.000 |
|       | Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
|       | public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |
|       | cinemas, games and hobbies rooms, barber shops, pantries       |                                 |       |
|       | containing no cooking appliances and similar spaces            |                                 |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x11  | ip=B227.ga=ga.room=refrigerated | 0.000 |
|       | Def. 11: Public spaces are those portions of the               |                                 |       |
|       | accomodation which are used for halls, dining rooms,           |                                 |       |
|       | lounges and similar permanently enclosed spaces.               |                                 |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x10  | ship=B227.ga=ga.room=galley     | 0.000 |
|       | Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
|       | public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |
|       | cinemas, games and hobbies rooms, barber shops, pantries       |                                 |       |
|       | containing no cooking appliances and similar spaces            |                                 |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x11  | ship=B227.ga=ga.room=galley     | 0.000 |
|       | Def. 11: Public spaces are those portions of the               |                                 |       |
|       | accomodation which are used for halls, dining rooms,           |                                 |       |
|       | lounges and similar permanently enclosed spaces.               |                                 |       |
| ----- |  |                                 |       |
| 8     | SOLASII2.PartC.Reg43.accosurf                                  | ship=B227.ga=ga.surf=qcm1       | 0.000 |
|       | check, if surface meets accomodation or service space          |                                 |       |
|       | -einer von Schleife  | .panel                          |       |
|       | 0. Teilbereich   | p=B227.ga=ga.surf=qcm1.panel=p1 | 0.000 |
| ----- |  |                                 |       |
| 9     | SOLASII2.PartC.Reg43.accopanel                                 | p=B227.ga=ga.surf=qcm1.panel=p1 | 0.000 |
|       | check, if panel meets accomodation or service space            |                                 |       |
|       | -einer von Schleife  | .room                           |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x10  | ship=B227.ga=ga.room=staircase  | 0.000 |
|       | Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
|       | public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |
|       | cinemas, games and hobbies rooms, barber shops, pantries       |                                 |       |
|       | containing no cooking appliances and similar spaces            |                                 |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x11  | ship=B227.ga=ga.room=staircase  | 0.000 |
|       | Def. 11: Public spaces are those portions of the               |                                 |       |
|       | accomodation which are used for halls, dining rooms,           |                                 |       |
|       | lounges and similar permanently enclosed spaces.               |                                 |       |
| ----- |  |                                 |       |
| 10    | SOLASII2.PartA.Reg3.x10  | ip=B227.ga=ga.room=enginecasing | 0.000 |
|       | Def. 10: Accomodation spaces are those spaces used for         |                                 |       |
|       | public spaces, corridors, lvatories,cabins,offices, hospitals, |                                 |       |

|       |  |   |       |
|-------|--|---|-------|
|       |  | cinemas, games and hobbies rooms, barber shops, pantries        |       |
|       |  | containing no cooking appliances and similar spaces             |       |
| ----- |  |   |       |
| 10    |  | SOLASII2.PartA.Reg3.x11  ship=B227.ga=ga.room=enginecasing      | 0.000 |
|       |  | Def. 11: Public spaces are those portions of the                |       |
|       |  | accomodation which are used for halls, dining rooms,            |       |
|       |  | lounges and similar permanently enclosed spaces.                |       |
| ----- |  |   |       |
| 8     |  | SOLASII2.PartC.Reg43.accosurf  ship=B227.ga=ga.surf=qsm1        | 0.000 |
|       |  | check, if surface meets accomodation or service space           |       |
|       |  | -einer von Schleife  .panel                                     |       |
|       |  | 0. Teilbereich  p=B227.ga=ga.surf=qsm1.panel=p1                 | 0.000 |
| ----- |  |   |       |
| 9     |  | SOLASII2.PartC.Reg43.accopanel  p=B227.ga=ga.surf=qsm1.panel=p1 | 0.000 |
|       |  | check, if panel meets accomodation or service space             |       |
|       |  | -einer von Schleife  .room                                      |       |
| ----- |  |   |       |
| 10    |  | SOLASII2.PartA.Reg3.x10  ship=B227.ga=ga.room=laundry           | 0.000 |
|       |  | Def. 10: Accomodation spaces are those spaces used for          |       |
|       |  | public spaces, corridors, lvatories,cabins,offices, hospitals,  |       |
|       |  | cinemas, games and hobbies rooms, barber shops, pantries        |       |
|       |  | containing no cooking appliances and similar spaces             |       |
| ----- |  |   |       |
| 10    |  | SOLASII2.PartA.Reg3.x11  ship=B227.ga=ga.room=laundry           | 0.000 |
|       |  | Def. 11: Public spaces are those portions of the                |       |
|       |  | accomodation which are used for halls, dining rooms,            |       |
|       |  | lounges and similar permanently enclosed spaces.                |       |
| ----- |  |   |       |
| 10    |  | SOLASII2.PartA.Reg3.x10  ship=B227.ga=ga.room=emergency         | 0.000 |
|       |  | Def. 10: Accomodation spaces are those spaces used for          |       |
|       |  | public spaces, corridors, lvatories,cabins,offices, hospitals,  |       |
|       |  | cinemas, games and hobbies rooms, barber shops, pantries        |       |
|       |  | containing no cooking appliances and similar spaces             |       |
| ----- |  |   |       |
| 10    |  | SOLASII2.PartA.Reg3.x11  ship=B227.ga=ga.room=emergency         | 0.000 |
|       |  | Def. 11: Public spaces are those portions of the                |       |
|       |  | accomodation which are used for halls, dining rooms,            |       |
|       |  | lounges and similar permanently enclosed spaces.                |       |
| ----- |  |   |       |
| 6     |  | SOLASII2.PartC.Reg44  ship=B227                                 | 0.000 |
|       |  | Fire integrity for bulkheads and decks                          |       |
| ----- |  |   |       |
| 7     |  | SOLASII2.PartC.Reg44.x1  ship=B227                              | 0.000 |
|       |  | In addition to complying with the specific provisions           |       |
|       |  | for fire integrity of bulkheads and decks mentioned             |       |
|       |  | elsewhere in this part, the minimum fire integrity              |       |
|       |  | of bulkheads and decks shall be as prescribed in                |       |
|       |  | tables 44.1 and 44.2  |       |

|  |                                 |       |  |
|--|---------------------------------|-------|--|
| -fuer alle Schleife                    | .ga=ga.surf                     |       |  |
| 0. Teilbereich                         | ship=B227.ga=ga.surf=lsm2       | 1.000 |  |
| 1. Teilbereich                         | ship=B227.ga=ga.surf=lsm1       | 1.000 |  |
| 2. Teilbereich                         | ship=B227.ga=ga.surf=lsm3       | 1.000 |  |
| 3. Teilbereich                         | ship=B227.ga=ga.surf=lpm2       | 0.000 |  |
| 4. Teilbereich                         | ship=B227.ga=ga.surf=lpm1       | 1.000 |  |
| 5. Teilbereich                         | ship=B227.ga=ga.surf=lsm        | 0.000 |  |
| 6. Teilbereich                         | ship=B227.ga=ga.surf=lpm        | 0.000 |  |
| 7. Teilbereich                         | ship=B227.ga=ga.surf=qam3       | 1.000 |  |
| 8. Teilbereich                         | ship=B227.ga=ga.surf=qam2       | 1.000 |  |
| 9. Teilbereich                         | ship=B227.ga=ga.surf=qam1       | 0.000 |  |
| 10. Teilbereich                        | ship=B227.ga=ga.surf=qfm        | 1.000 |  |
| 11. Teilbereich                        | ship=B227.ga=ga.surf=qpm1       | 1.000 |  |
| 12. Teilbereich                        | ship=B227.ga=ga.surf=qcm3       | 1.000 |  |
| 13. Teilbereich                        | ship=B227.ga=ga.surf=qcm2       | 0.000 |  |
| 14. Teilbereich                        | ship=B227.ga=ga.surf=qcm1       | 1.000 |  |
| 15. Teilbereich                        | ship=B227.ga=ga.surf=qsm1       | 1.000 |  |
| 16. Teilbereich                        | ship=B227.ga=ga.surf=qsm2       | 1.000 |  |
| 17. Teilbereich                        | ship=B227.ga=ga.surf=lsm4       | 1.000 |  |
| -----                                  |                                 |       |  |
| 8 SOLASII2.PartC.Reg44.Table44x1       | ship=B227.ga=ga.surf=lpm2       | 0.000 |  |
| Table 44.1                             |                                 |       |  |
| -fuer alle Schleife                    | .panel                          |       |  |
| 0. Teilbereich                         | p=B227.ga=ga.surf=lpm2.panel=p1 | 0.000 |  |
| -----                                  |                                 |       |  |
| innere Regel 1                         | p=B227.ga=ga.surf=lpm2.panel=p1 | 0.000 |  |
| -fuer alle Schleife                    | .room                           |       |  |
| alle Raeume, die an das Schott grenzen |                                 |       |  |
| 0. Teilbereich                         | rf=lpm2.panel=p1.room=staircase | 0.000 |  |
| 1. Teilbereich                         | .surf=lpm2.panel=p1.room=galley | 1.000 |  |
| -----                                  |                                 |       |  |
| innere Regel 0                         | rf=lpm2.panel=p1.room=staircase | 0.000 |  |
| -fuer alle Schleife                    | ^.room                          |       |  |
| alle Raeume, die an das Schott grenzen |                                 |       |  |
| 0. Teilbereich                         | rf=lpm2.panel=p1.room=staircase | 1.000 |  |
| 1. Teilbereich                         | .surf=lpm2.panel=p1.room=galley | 0.000 |  |
| -----                                  |                                 |       |  |
| 8 SOLASII2.PartC.Reg44.Table44x1       | ship=B227.ga=ga.surf=lsm        | 0.000 |  |
| Table 44.1                             |                                 |       |  |
| -fuer alle Schleife                    | .panel                          |       |  |
| 0. Teilbereich                         | ip=B227.ga=ga.surf=lsm.panel=p5 | 1.000 |  |
| 1. Teilbereich                         | ip=B227.ga=ga.surf=lsm.panel=p4 | 0.000 |  |
| 2. Teilbereich                         | ip=B227.ga=ga.surf=lsm.panel=p3 | 1.000 |  |
| 3. Teilbereich                         | ip=B227.ga=ga.surf=lsm.panel=p2 | 0.000 |  |
| 4. Teilbereich                         | ip=B227.ga=ga.surf=lsm.panel=p1 | 1.000 |  |
| -----                                  |                                 |       |  |
| innere Regel 1                         | ip=B227.ga=ga.surf=lsm.panel=p4 | 0.000 |  |
| -fuer alle Schleife                    | .room                           |       |  |

```

|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lsm.panel=p4.room=opendeck|1.000|
|   1. Teilbereich           |surf=lsm.panel=p4.room=corridor|0.000|
-----
|innere Regel 0           |surf=lsm.panel=p4.room=corridor|0.000|
| -fuer alle Schleife     |^.room           |      |
|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lsm.panel=p4.room=opendeck|1.000|
|   1. Teilbereich           |surf=lsm.panel=p4.room=corridor|0.000|
-----
|innere Regel 1           |ip=B227.ga=ga.surf=lsm.panel=p2|0.000|
| -fuer alle Schleife     |.room           |      |
|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lsm.panel=p2.room=opendeck|1.000|
|   1. Teilbereich           |.surf=lsm.panel=p2.room=laundry|0.000|
-----
|innere Regel 0           |.surf=lsm.panel=p2.room=laundry|0.000|
| -fuer alle Schleife     |^.room           |      |
|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lsm.panel=p2.room=opendeck|1.000|
|   1. Teilbereich           |.surf=lsm.panel=p2.room=laundry|0.000|
-----
8|SOLASII2.PartC.Reg44.Table44x1 |ship=B227.ga=ga.surf=lpm           |0.000|
| Table 44.1           |           |           |
| -fuer alle Schleife     |.panel           |      |
|   0. Teilbereich           |ip=B227.ga=ga.surf=lpm.panel=p4|1.000|
|   1. Teilbereich           |ip=B227.ga=ga.surf=lpm.panel=p3|0.000|
|   2. Teilbereich           |ip=B227.ga=ga.surf=lpm.panel=p2|1.000|
|   3. Teilbereich           |ip=B227.ga=ga.surf=lpm.panel=p1|1.000|
-----
|innere Regel 1           |ip=B227.ga=ga.surf=lpm.panel=p3|0.000|
| -fuer alle Schleife     |.room           |      |
|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lpm.panel=p3.room=opendeck|1.000|
|   1. Teilbereich           |surf=lpm.panel=p3.room=corridor|0.000|
-----
|innere Regel 0           |surf=lpm.panel=p3.room=corridor|0.000|
| -fuer alle Schleife     |^.room           |      |
|   alle Raeume, die an das Schott grenzen           |
|   0. Teilbereich           |surf=lpm.panel=p3.room=opendeck|1.000|
|   1. Teilbereich           |surf=lpm.panel=p3.room=corridor|0.000|
-----
8|SOLASII2.PartC.Reg44.Table44x1 |ship=B227.ga=ga.surf=qam1           |0.000|
| Table 44.1           |           |           |
| -fuer alle Schleife     |.panel           |      |
|   0. Teilbereich           |p=B227.ga=ga.surf=qam1.panel=p2|0.000|
|   1. Teilbereich           |p=B227.ga=ga.surf=qam1.panel=p1|1.000|
-----

```

```

|innere Regel 1                |p=B227.ga=ga.surf=qam1.panel=p2|0.000|
| -fuer alle Schleife          |.room                            |      |
|   alle Raeume, die an das Schott grenzen          |                                |
|   0. Teilbereich             |urf=qam1.panel=p2.room=opendeck|1.000|
|   1. Teilbereich             |urf=qam1.panel=p2.room=corridor|0.000|
-----
|innere Regel 0                |urf=qam1.panel=p2.room=corridor|0.000|
| -fuer alle Schleife          |^.room                            |      |
|   alle Raeume, die an das Schott grenzen          |                                |
|   0. Teilbereich             |urf=qam1.panel=p2.room=opendeck|1.000|
|   1. Teilbereich             |urf=qam1.panel=p2.room=corridor|0.000|
-----
8|SOLASII2.PartC.Reg44.Table44x1 |ship=B227.ga=ga.surf=qcm2        |0.000|
| Table 44.1                    |                                |
| -fuer alle Schleife          |.panel                            |      |
|   0. Teilbereich             |p=B227.ga=ga.surf=qcm2.panel=p3|1.000|
|   1. Teilbereich             |p=B227.ga=ga.surf=qcm2.panel=p2|1.000|
|   2. Teilbereich             |p=B227.ga=ga.surf=qcm2.panel=p1|0.000|
-----
|innere Regel 1                |p=B227.ga=ga.surf=qcm2.panel=p1|0.000|
| -fuer alle Schleife          |.room                            |      |
|   alle Raeume, die an das Schott grenzen          |                                |
|   0. Teilbereich             |urf=qcm2.panel=p1.room=corridor|0.000|
|   1. Teilbereich             |.surf=qcm2.panel=p1.room=galley|1.000|
-----
|innere Regel 0                |urf=qcm2.panel=p1.room=corridor|0.000|
| -fuer alle Schleife          |^.room                            |      |
|   alle Raeume, die an das Schott grenzen          |                                |
|   0. Teilbereich             |urf=qcm2.panel=p1.room=corridor|1.000|
|   1. Teilbereich             |.surf=qcm2.panel=p1.room=galley|0.000|
-----

```

Schalter

-----

Unbekannte wurden erfragt bzw. über die DB berechnet

Integration mittels Trapezregel

Anzahl Stützstellen : 10

## C.3.5 Dimensionieren, tabellarische Ausgabe, kurz

```
+-----+
| Dimensionieren |
+-----+
```

Wahrscheinlichkeit, dass aus  
'SOLASII-2'  
folgende Regel erfuehlt ist :

Regel 'SOLASII2.PartC.Reg44.Table44x1'  
fuer 'ship=B227.ga=ga.surf=lpm2' aus e:\uni\promo\zzzdat\solasii2

| Ebene | Regel   | angewendet auf   | er-<br>gibt                |
|-------|---|--|----------------------------|
| 0     | SOLASII2.PartC.Reg44.Table44x1<br>  Table 44.1<br>  -fuer alle Schleife<br>  0. Teilbereich | ship=B227.ga=ga.surf=lpm2<br> .panel<br> p=B227.ga=ga.surf=lpm2.panel=p1 | 1.000<br> <br> <br>  1.000 |

Dimensionierung  
-----

Optimierungsmethode : Zufallssuche mit Nebenbedingung N1  
Optimierungskriterium : Abstand der Grenzen  
= 0.6654

| Name                               | Unter Gr. | Ober Gr. |
|------------------------------------|-----------|----------|
| ship=B227.ga=ga.surf=lpm2.insclass | 4.0039    | 6.0000   |

Schalter  
-----

Unbekannte wurden erfragt bzw. über die DB berechnet  
Integration mittels Trapezregel  
Anzahl Stützstellen : 10

## D Programmsteuerung

### D.1 Compiler

COMP MY-TOOLS Benutzer Hinweise COMP

#### KOMMANDO

comp - Regelsyntax Compiler

#### FORMAT

comp [ -sxxxx ] [ -n ] [ -r ] [ -p ] source regeldb

#### BESCHREIBUNG

Die in der Datei source in der Regel-Syntax beschriebenen Regeln werden analysiert und die Strukturbäume in die Datenbank regeldb geschrieben. Dateiname ohne Extensions angeben. Bei source wird .reg erwartet.

Es wird immer eine Protokolldatei unter dem Namen source.log angelegt.

Bei Fehlern oder Warnungen während der Analyse wird der Wert 1 zurückgegeben, sonst 0.

Optionen:

- sxxxx Reservierter Speicher fuer den Strukturbaum einer Regel in kB  
Voreinstellung: -s64
- n Anlegen einer neuen Regel-Datenbank
- r Ueberschreiben vorhandener Regeln in Regel-Datenbank
- p Ausgabe des Strukturbaumes in Protokolldatei nach erfolgreicher Analyse

## D.2 Bearbeiten

REGEL MY-TOOLS Benutzer Hinweise REGEL

### KOMMANDO

regel - Bearbeiten von Regelwerken

### FORMAT

```
regel [ -sxxx ] [ -n ] [ -r ] [ -u ] [ -x ] [ -m(m|t) ]
      [ -amxx ] [ -atxx ] [ -ox ] [ -kx ]
      protokoll regelddb regelwerk regelname objektodb objekt
      ergdb modus [ optwerte ]
```

### BESCHREIBUNG

Das Programm führt die unter modus angegebene Aufgabe für die Regel regelname aus regelwerk, deren Strukturbaum in regelddb abgelegt ist, für objekt aus der Objektdatenbank objddb aus.

Die Ergebnisstruktur wird in ergdb abgelegt.

modus kann folgende Werte annehmen:

|   |                |
|---|----------------|
| 0 | Inhalt         |
| 1 | Relevanz       |
| 2 | Pruefen        |
| 3 | Dimensionieren |

Wenn modus = 4 ist, muß in optwerte eine Liste von zu optimierenden Werten mit den max. und min

Während der Bearbeitung werden Fehlermeldungen in die Datei protokoll geschrieben.

Bei Fehlern wird der Wert 1 zurückgegeben, sonst 0.

Optionen:

```
-sxxxx  Reservierter Speicher fuer Ergebnisse einer
        Regel in kB
        Voreinstellung: -s8

-n      Anlegen einer neuen Ergebnis-Datenbank

-r      Ueberschreiben vorhandener Ergebnisse in Ergebnis-
```

Datenbank

-u       Unbekannte Werte nicht Rechnen oder Lesen

-x       rechnen nicht ausfuehren bei Relevanz

-m(m|t) Methode zum numerischen Berechnen der Integrale und  
Werte  
-mm     Monte-Carlo-Methode  
-mt     Trapezregel  
Voreinstellung: -mt

-amxxx   Anzahl der Stuetzstellen für die Monte-Carlo-Method  
Voreinstellung: -am50

-atxxx   Anzahl der Stuetzstellen für die Trapezregel  
Voreinstellung: -am50

-o(0:4) Optimierungsmethode  
-o0     von Hand  
-o1     Methode OPT, Ungleichungen für  
          Obergrenzen >= Untergrenzen  
-o2     Methode OPT, Ungleichungen für  
          Obergrenzen >= Untergrenzen und  
          Erfülltsein der Regel >= 1.0  
-o3     Zufallssuche, Ungleichungen für  
          Obergrenzen >= Untergrenzen  
-o4     Zufallssuche, Ungleichungen für  
          Obergrenzen >= Untergrenzen und  
          Erfülltsein der Regel >= 1.0  
Voreinstellung: -o0

-k(0:3) Optimierungskriterium  
-k0     Erfülltsein der Regel  
-k1     Erfülltsein + Breite des Bereichs \* Gewicht  
-k2     Breite des Bereichs  
-k3     Erfülltsein, wenn Erfülltsein <= 1.0,  
          sonst 1.0 + Breite des Bereichs  
Voreinstellung: -k0

### D.3 Ausgabe

POSTP

MY-TOOLS Benutzer Hinweise

POSTP

#### KOMMANDO

```
postp - Regelergebnis Ausgabe
```

#### FORMAT

```
postp [ -l ] [ -b ] [ -r ] [ -v ] [ -k ] [ -s ] [ -l ]  
      [ -t ] [ -i ] [ -a ] [ -u ] [ -f ]  
      ergebnis ergdb regelwerk regelname objekt db objekt  
      modus
```

#### BESCHREIBUNG

Das Programm liest die Ergebnisstruktur für die Regel regelname aus regelwerk für die Aufgabe modus für das Objekt objekt aus objdb und schreibt die Ausgabe formatiert in die Datei ergebnis.

modus kann folgende Werte annehmen:

|   |                |
|---|----------------|
| 0 | Inhalt         |
| 1 | Relevanz       |
| 2 | Pruefen        |
| 3 | Dimensionieren |

Bei Fehlern wird der Wert 1 zurückgegeben, sonst 0.

Optionen:

- l lineare Ausgabe
- b tabellarische Ausgabe (Blockweise)  
Voreingestellt
- r rekursiv, d.h. Regelaufrufe in Regeln ausgeben
- v Kriterien, deren Voraussetzung 0 sind ausgeben
- k Kriterien ausgeben
- s Regelaufrufe mit erg = 1
- c Kommentare ausgeben

- t Teilbereiche, die nicht anzuwenden sind ausgeben
- i innere Regelaufrufe ausgeben
- u Ausgabe aller verwendeter Werte
- f gleichbedeutend mit -rvksctiu
- a Ausgabe an existierende Ausgabedatei anhängen



```

Postprozessor :                STARTEN
                  : letzte Ergebnisse ANSEHEN

Compiler       :                STARTEN
                  : Log-Datei        ANSEHEN
                  : Regelcode Datei  EDITIEREN
                  : Regel und Log-Datei EDITIEREN

Obj Datenbank  :                STARTEN

```

Den aufgerufenen Programmen werden die Parameter aus der folgenden Liste übergeben:

```

Quit           :
  - Konfiguration speichern (0/1)
  - Objekt Datenbank sichern (0/1)

Shell         :
  - Konfigurationsdatei
  - Pfad fuer Scratch-Dateien

Allg. Parameter :
  - Regel Source Datei
  - Regel Datenbank
  - Vorschrift
  - Regel
  - Objektdatenbank
  - Objekt
  - Ergebnisdatenbank oder SC für Scratch
  - Ausgabedatei oder SC für Scratch-Datei

Regelmodul    :
  - Unbekannte lesen (0/1)
  - Rechenmethode:
    0 = Trapez Regel
    1 = Monte Carlo
  - Anz.Stuetzstellen Trapez Regel
  - Anz.Stuetzstellen Monte Carlo
  - 'rechnen' ausfuehren bei RELEVANZ (0,1)
  - Optimierungs Kriterium
    0 = Erfülltsein der Regel
    1 = Erfülltsein + Breite des Bereichs * Gewicht
    2 = Breite des Bereichs
    3 = Erfülltsein, wenn Erfülltsein <= 1.0,

```

- sonst 1.0 + Breite des Bereichs
- Optimierungsmethode
  - 0 = von Hand
  - 1 = Methode OPT, Ungleichungen für  
Obergrenzen  $\geq$  Untergrenzen
  - 2 = Methode OPT, Ungleichungen für  
Obergrenzen  $\geq$  Untergrenzen und  
Erfülltsein der Regel  $\geq$  1.0
  - 3 = Zufallssuche, Ungleichungen für  
Obergrenzen  $\geq$  Untergrenzen
  - 4 = Zufallssuche, Ungleichungen für  
Obergrenzen  $\geq$  Untergrenzen und  
Erfülltsein der Regel  $\geq$  1.0
- Speicherplatz fuer ein Regelergebnis in kB
- Ergebnisdatenbank:
  - 0 = neuanlegen
  - 1 = ergänzen, überschreiben

## Postprozessor :

- automatisch starten (0/1)
- Ausgabe Modus
  - 0 = tabellarisch
  - 1 = chronologisch
- rekursive Regelaufrufe (0/1)
- Kriterien, deren Voraussetzung 0 sind (0/1)
- Kriterien ausgeben (0/1)
- Regelaufrufe mit P = 1 (0/1)
- Kommentare ausgeben (0/1)
- Teilbereiche, die nicht anzuwenden sind (0/1)
- innere Regelaufrufe (0/1)
- benutze Werte ausgeben
- Aufgabe:
  - 0 = Inhalt
  - 1 = Relevanz
  - 2 = Prüfen
  - 3 = Dimensionieren
- Ausgabedatei:
  - 0 = neuanlegen
  - 1 = ergänzen

## Compiler :

- Regel Datenbank:
  - 0 = neuanlegen
  - 1 = ergänzen/ueberschreiben
- Strukturbaum in Protokoll-Datei

- Speicherplatz in kB fuer Analyse
- nach Aufruf .erg und .log Datei ansehen