

Heft 102

Peter Loos

**Konzeption einer graphischen
Rezeptverwaltung und deren Integration
in eine CIP-Umgebung - Teil 1**

Juni 1993

Peter Loos

Konzeption einer graphischen Rezeptverwaltung und deren Integration in eine CIP-Umgebung - Teil 1

Inhaltsübersicht

1. Einleitung	1
2. Charakterisierung der verfahrenstechnischen Industrie	2
3. Arten von Rezepten.....	3
4. Ereignisse für den Anstoß der Rezepterstellung	4
5. Datenmodell der Rezeptstrukturen.....	7
6. Funktionen und Benutzeroberfläche.....	14
6.1. Materialflußsicht.....	16
6.2. Anlagensicht.....	18
6.3. Phasensicht.....	18
7. Ausblick.....	22
Literaturverzeichnis.....	23

1. Einleitung

Die seit einigen Jahren viel diskutierten Konzepte des Computer Integrated Manufacturing (CIM) reflektieren vorwiegend die Sicht der diskreten Industrie. So stehen bei CIM Fragen der Integration von CAD und CAM, von CAD und Stücklisten oder von CAM und Fertigungssteuerung im Vordergrund. Prototypische Anwendungen liegen in den Bereichen des Automobilbaus, der Zulieferindustrie und dem Maschinenbau. Die Anforderungen der verfahrenstechnischen Industrie werden durch die CIM-Konzepte dagegen nur unvollständig abgedeckt. Nur vereinzelt findet man unter dem Begriff Computer Integrated Processing (CIP) Konzepte für diese Industrie (vgl. Polke 89). Ein umfassender konzeptioneller Rahmen für CIP, wie ihn beispielweise Scheer für CIM gegeben hat (vgl. Scheer 90a), fehlt bisher.

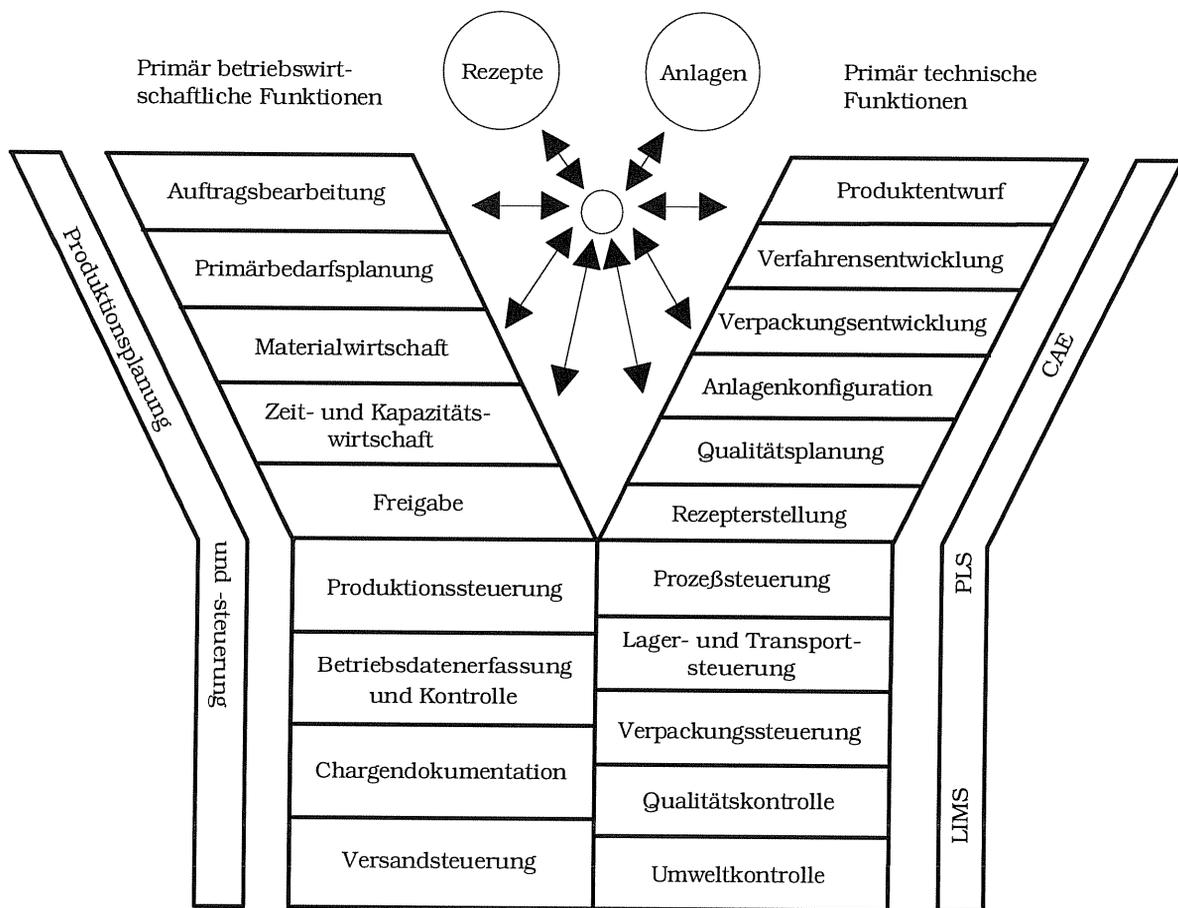


Abb. 1: Informationssysteme für die Produktion in der verfahrenstechnischen Industrie (CIP)

In Abbildung 1 werden die Informationssysteme für die Produktion in der verfahrenstechnischen Industrie in Anlehnung an das Y-Modell von Scheer gezeigt. Die Darstellung der betriebswirtschaftlich orientierten Systeme (linke Seite des Y-Modells) weicht nur wenig von

der des CIM-Konzeptes ab. Der Grund hierfür ist, daß es aus betriebswirtschaftlicher Sicht prinzipiell keine unterschiedlichen Konzepte für die verfahrenstechnische Industrie gibt, was sich in der größtenteils gleichen Funktionalität der Systeme widerspiegelt, die hauptsächlich durch interne Adaptionen an die einzelnen Bedürfnisse der verfahrenstechnischen Industrie angepaßt werden (vgl. *Hofmann 92, Luber 92*). Dagegen gibt es starke Abweichungen bei den technisch orientierten Systemen, was sich aus den unterschiedlichen Produktionsprozessen ergibt und zu unterschiedlichen Anwendungssystemen führt.

Im folgenden soll ein Konzept für die Rezepterstellung und ihre Integration in das Computer Integrated Processing aufgestellt werden. Dazu wird eine Charakterisierung der verfahrenstechnischen Industrie vorgenommen, der Rezeptbegriff eingeordnet sowie die auslösenden Ereignisse für die Rezepterstellung definiert. Anschließend werden die Datenstrukturen sowie die Funktionen und Benutzeroberflächen einer Rezeptverwaltung beschrieben.

2. Charakterisierung der verfahrenstechnischen Industrie

Die verfahrenstechnische Industrie ist geprägt durch die Herstellung von Gütern, die sich während des Produktionsprozesses zumindest teilweise in einem nicht-festen Aggregatzustand befinden oder deren Ausmaß zu klein für die Mengeneinheit 'Stück' ist, z.B. Granulat oder Mehl. Aus diesem Grund erfolgt die Produktion meist chargenweise, wobei die Größe einer Charge mit Maßeinheiten wie Kilogramm, Tonnen, Liter oder Kubikmeter gemessen wird. Verfahrenstechnische Prozesse findet man beispielsweise in der chemischen und pharmazeutischen Industrie, in der Nahrungs- und Genußmittelindustrie, in der Gummiindustrie oder in der papiererzeugenden Industrie. Die unterschiedlichen Produktionsprozeß werden in den Herstellungsvorschriften dokumentiert. So spricht man hier, im Gegensatz zu den Arbeitsplänen der diskreten Fertigung, von Rezepten oder Rezepturen.

Abbildung 2 zeigt eine Typologie für die industrielle Produktion, bei der die Intensität der Schattierung der Merkmalsausprägungen die Relevanz für die verfahrenstechnische Industrie wiedergibt (zur Typologie der industrielle Produktion vgl. *Jost 92*). Insbesondere die Merkmale Kontinuität, Kuppelproduktion, Beherrschbarkeit und Materialverbrauch haben direkten Einfluß auf die Herstellvorschriften und somit auf die Art der Informationsdarstellung für Rezepte.

Merkmale	Merkmalsausprägungen			
Güterform	Fließgüter		Stückgüter	
Stoffverwertung	Analytisch	Durchlaufend		Synthetisch
Fertigungsart	Einzel-	Kleinserien-	Großserien-	Massen-
Fertigungsorganisation	Werkstatt-	Gruppen-	Linien-	Fließ-
Produktionsprozesse	Herstellung	Montage		Verpackung
Kontinuität	Kontinuierlich		Diskontinuierlich	
Teile der Erzeugnisstruktur	Komplex	Einfach	Einteilige Erzeugnisse	
Kuppelproduktion	Keine	Geringe	Starke	
Beherrschbarkeit	Einfach, deterministisch		Komplex, nicht-deterministisch	
Materialverbrauch	Linear		Nicht-linear	

Abb. 2: Typologie für die verfahrenstechnische Produktion

3. Arten von Rezepten

Für eine Beschreibung von Verfahrensabläufen ist es notwendig, allgemeingültige Strukturen zu definieren, die auf unterschiedliche Arten verfahrenstechnischer Herstellungsprozesse angewandt werden können. Von verschiedenen Gremien werden deshalb Begriffe der verfahrenstechnischen Produktion definiert und klassifiziert, so z.B. von dem SP88-Gremium der Instrument Society of America (ISA) und dem Arbeitskreis 2.3 der Normenarbeitsgemeinschaft für Meß- und Regelungstechnik in der Chemischen Industrie

(NAMUR). Die folgenden Begriffe und Grundstrukturen lehnen sich deshalb weitgehend an die Empfehlungen der genannten Gremien an (vgl. *NE33* und *SP88*).

Rezepte lassen sich in verschiedene Bestandteile zerlegen (vgl. *Uhlig 87*). Eine **Operation** ist die kleinste selbständig ausführbare Einheit innerhalb eines Rezeptes. Sie setzt sich aus verschiedenen Funktionen, auch **Phasen** genannt, zusammen. Werden Funktionen automatisiert durchgeführt, so können ihnen Programme in Form von Steuercodes zugeordnet werden. Ein Steuercode enthält beispielsweise den Befehl zum Öffnen eines Ventils. Verschiedene Operationen eines Rezeptes, die einen Verfahrensabschnitt (z.B. Hydrieren) beschreiben, lassen sich zu einem **Teilrezept** zusammenfassen.

Neben dieser hierarchischen Gliederung kann man verschiedene Konkretisierungszustände von Rezepten unterscheiden. Ein **Verfahren** beschreibt die allgemeine Vorgehensweise zur Herstellung eines bestimmten Stoffes. Die Verfahren dienen zur Generierung von **Urrezepten** (englisch: **general recipe**), die um Informationen bezüglich Rohstoffe, Zwischenprodukte, Erzeugnisse und deren Mengenverhältnisse zueinander sowie um allgemeine Anforderungen an die zur Produktion notwendigen Anlagen ergänzt werden. Urrezepte bestehen aus Teilurzepten und chemisch-technischen Grundoperationen. Sie werden weiter zu **Grundrezepten** (**site recipe** und **master recipe**) detailliert, die um Informationen bezüglich der konkret einzusetzenden Anlagen ergänzt werden. Während die bisherigen Rezepttypen Stammdatencharakter besitzen, werden **Steuerrezepte** (**control recipe**) für Produktionsaufträge generiert und beziehen sich immer auf konkrete Chargen. Sie enthalten Termine, Mengen und Qualitäten des geplanten Produktionsauftrages bzw. realisierte Werte für abgeschlossene Aufträge (vgl. auch *Loos 93*).

4. Ereignisse für den Anstoß der Rezepterstellung

Die Rezept der verfahrenstechnischen Industrie enthalten die Bestandteile der herzustellenden Produkte sowie die Herstellenweisungen. Sie weisen somit Informationen auf, die bei der diskreten Fertigung in Stücklisten und Arbeitsplänen hinterlegt sind. Stücklisten werden üblicherweise in den Materialwirtschaftssystemen gepflegt. Wichtige Funktionen, wie die Materialbedarfsauflösung, benötigen die Stücklisten als Inputdaten. Arbeitspläne werden in den Zeit- und Kapazitätswirtschaftsmoduln verwaltet und dienen als Grundlage für Terminierung und Kapazitätsbedarfsrechnung. Obwohl die genannten Produktionsplanungs- und -steuerungsmoduln die Daten verwalten, werden die Eingangsinformationen in den technischen CIM-Systemen erzeugt, die Stücklisten beispielsweise im CAD-System und die Arbeitspläne im Arbeitsplanungssystem. Eine wesentliche Aufgabe von CIM besteht deshalb in der Integration der Generierungsfunktionen von CAD und Arbeitsplanung mit den Verwaltungsfunktionen des PPS-Systems. Dies gilt ebenso für die Erstellung von Rezepten.

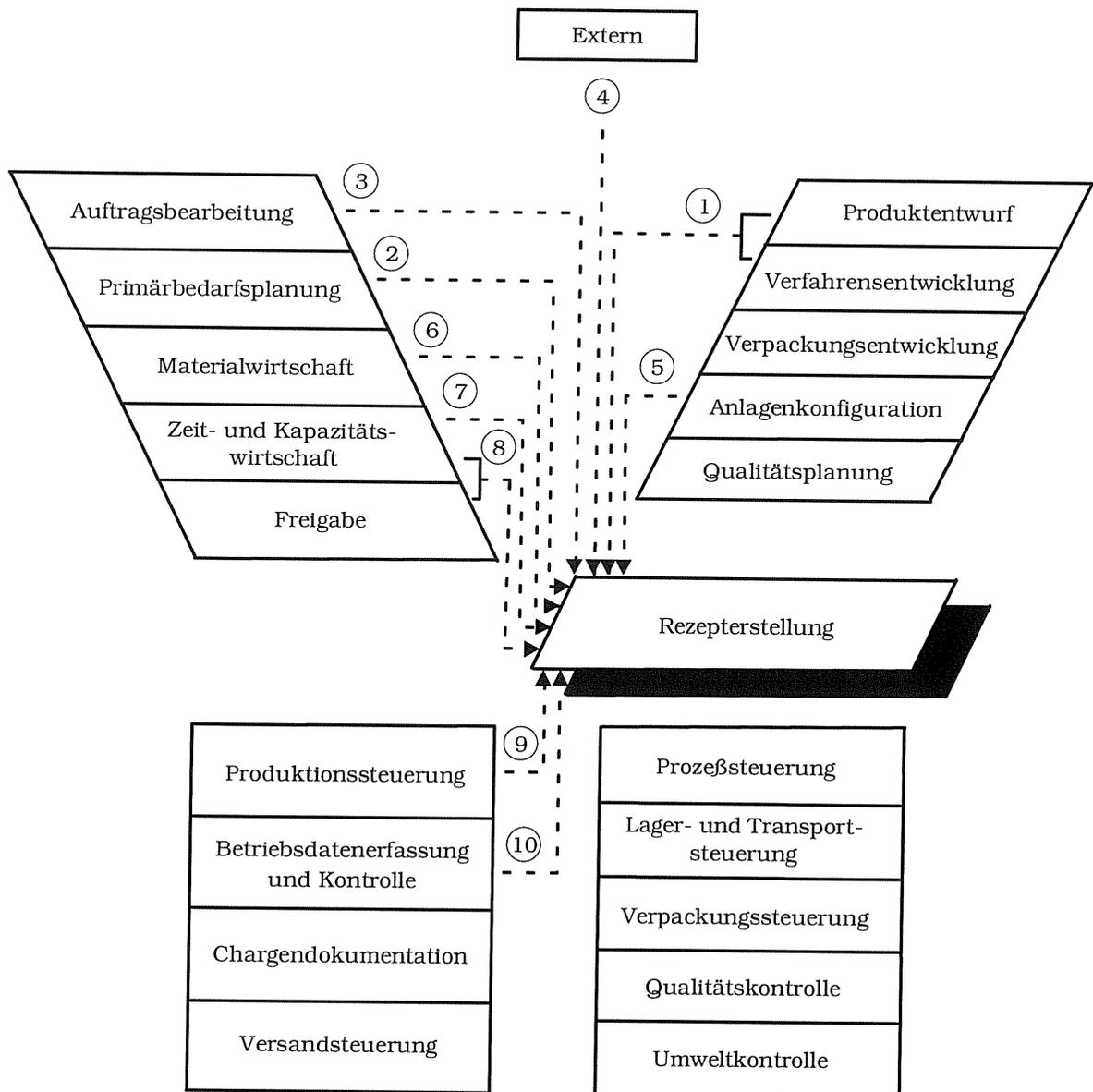


Abb. 3: Trigger für die Rezepterstellung im CIP-Modell

Obwohl die Information der Rezepte im PPS-System gespeichert wird (sei es in Stücklisten und Arbeitsplänen oder als Rezept), wird für deren Erzeugung ein eigenes System eingesetzt, das die technischen Randbedingungen berücksichtigen kann. In Abbildung 3 sind von anderen CIP-Komponenten ausgehende Trigger dargestellt, die eine Erstellung oder Modifikation von Rezepten auslösen. In Abbildung 4 werden die Trigger benannt und angegeben, auf welche Konkretisierungsebene von Rezepten sich diese Trigger auswirken können (die Angabe der Konkretisierungsebene erfolgt in Anlehnung an die SP88- und die NAMUR-Begriffe). Die Intensität der Schattierung gibt die Relevanz des Triggers für die Rezeptebene an.

SP88		General Recipe	Site Recipe	Master Recipe	Control Recipe
NAMUR		Urrezept		Grundrezept	Steuerrezept
1	R&D Order				
2	Feasibility for Long Time Planning				
3	Technical Feasibility				
4	Change of Legal Regulations				
5	Change of Unit Configuration				
6	Changes of Materials				
7	Changes of Resources				
8	MRP-Requirements / Production Order				
9	Problems During Production				
10	Evaluation of Historical Data				

Abb. 4: Zusammenhang zwischen Rezepterstellungstrigger und Rezeptkonkretisierung

So wird beispielsweise bei Trigger 1, der als Forschungs- und Entwicklungsauftrag aus dem Produktentwurf oder der Verfahrensentwicklung kommt, die Erstellung von Urrezepten angestoßen. Dies ist notwendig, um herauszufinden, ob sich ein in Entwicklung befindliches Produkt oder Verfahren mit den im Unternehmen zur Verfügung stehenden Anlagen und Materialien prinzipiell produziert werden kann. Trigger 8 wird dagegen durch die Freigabe von Produktionsaufträgen im PPS-Systems ausgelöst und bewirkt die Erstellung eines Steuerrezeptes. Eventuell muß bei einem neuen Produkt erst das Grundrezept erzeugt werden.

Neben der Integration mit den übrigen CIP-Komponenten sollten bei der Gestaltung eines Rezepterstellungsystems weitere Kriterien erfüllt werden:

- **Konsistenz bei der Rezeptrepräsentation**
Rezepte sollten immer in der gleichen Art dargestellt werden. Die Konsistenz gilt bezüglich Benutzeroberfläche, Begriffswelt und Rezeptstrukturen, bezüglich der unterschiedlichen Konkretisierungsebenen von Rezepten und bezüglich der unterschiedlichen Anwendungssysteme (z.B. Rezepterstellung, Produktionssteuerung oder Kontroll- und Auswertungsfunktionen).
- **Visualisierung von Rezeptstrukturen**
Die zum Teil komplexen Rezeptstrukturen und -abhängigkeiten sollten in einer einfach verständlichen Form visualisiert werden. Für den Fertigungsbereich wird dies im Rahmen von Leitstandskonzepten bereits erfolgreich eingesetzt, z.B. Gantt-Diagramme für die Darstellung von zeitlichen Abhängigkeiten (vgl. hierzu *Scheer/Loos 93*).
- **Einfache Rezepterstellung und -pflege**
Die Visualisierung sollte sich nicht nur auf die Wiedergabe von Rezeptstrukturen beziehen, sondern auch bei der Erstellung und Pflege genutzt werden, wie dies z.B. bei Projektsteuerungssystemen, CASE-Tools oder bei Anlagenkonfigurationssystemen geschieht. (vgl. auch *Reinhardt/Schrieber/Strozyk 93*).
- **Wiederverwendung von Rezepten**
Informationen bereits erstellter Rezepte sollten bei Anlage von neuen Rezepten genutzt werden können. Dies bedeutet nicht nur das Kopieren von Rezepten. Vielmehr sollten Unterstützungsfunktionen für die Nutzung von Bibliotheken für Operationen, Phasen oder SteuerCodes angeboten werden, auf die aus Rezepten referenziert werden kann. Neben der Erstellung erleichtert dies auch die Anpassung von Rezepten, z.B. bei Anlagenkonfigurationsänderung.
- **Regelunterstützung**
Das Erstellungssystem sollte die Konsistenzprüfung von Rezepten unterstützen. Dies erstreckt sich von der Überprüfung der Widerspruchsfreiheit von Materialverknüpfungen oder Vorgabezeiten bis zur Generierung von Vorgabezeiten aufgrund technischer Regeln.

5. Datenmodell der Rezeptstrukturen

Die Rezeptdatenstrukturen als Grundlage für die Funktionen einer Rezeptverwaltung sollten mindestens folgenden Anforderungen genügen:

- In einem Rezept sollen verschiedene Rezeptalternativen definiert werden können. Dies kann eine einzelne Operation, aber auch eine Folge verschiedener Operationen betreffen.

Dabei soll nicht nur die Möglichkeit gegeben werden, ein zweites Rezept zu definieren, sondern verschiedene Alternativen sollen in einem Rezept hinterlegt werden können. Dadurch müssen jeweils nur die alternativen Schritte gepflegt werden, die gemeinsamen Operationen werden nur einmal definiert. Eine Alternative innerhalb eines Rezeptes wird als Pfad bezeichnet.

- Die Operationen innerhalb eines Rezeptes sollen in unterschiedlicher Reihenfolge abarbeitbar sein. Obwohl diese Anforderung eher bei flexiblen Fertigungssystemen in der diskreten Fertigung anzutreffen ist (vgl. *Zörmlein 88*), soll sie auch für die Strukturierung von Rezepten aufgestellt werden, da häufig eine Mischform der Produktion anzutreffen ist.
- Innerhalb eines Rezeptes müssen neben sequentiellen auch parallele Bearbeitungen zugelassen werden. Dies ist beispielsweise bei der zeitweisen Aufsplittung von Chargen oder bei der parallelen Bearbeitung von Mischungskomponenten notwendig.
- Die einzelnen Prozeßschritte eines Rezeptes, die durch Operationen und Phasen repräsentiert werden, sollen auch unabhängig von einem konkreten Rezept für den Aufbau von Operations- bzw. Phasenbibliotheken verwaltet werden können.
- Das Eingangs- und Ausgangsmaterial muß exakt den einzelnen Prozeßschritten zugeordnet werden können. Dies ist u.a. für eine genaue Materialbedarfsermittlung sowie für die Materialflußsteuerung notwendig.
- Es muß möglich sein, Variationen der Eingangsmaterialien zu definieren. Dies gilt zum einen für die komplette Substitution von Materialien, andererseits für Änderungen von Mischungsverhältnissen.

Adäquate Datenstrukturen für Ur- und Grundrezepte, die den gestellten Anforderung genügen, sind in Abbildung 5 dargestellt. Als Beschreibungssprache wird das Expanded Entity-Relationship-Modell (PERM) genutzt, das neben den allgemein gebräuchlichen Konstrukten des Entity-Relationship-Modells mit Hilfe der RC-Diagramme auch die Formulierung zusätzlicher Integritätsbedingungen gestattet (vgl. *Loos 92*).

Das Urrezept besteht aus einem Rezeptkopf (Typ *Urrezept*) und den notwendigen Prozeßschritten (Typ *Prozeßschritt*). Prozeßschritte enthalten eine grobe Beschreibung des Verfahrens und sind einem Urrezept eindeutig zugeordnet. Die Reihenfolge der Prozeßschritte ist durch die rekursive Beziehung *Prozeßfolge* festgelegt. Die Integritätsbedingung <2> schließt Zyklen in den Prozeßschritten aus (<2.1>) und stellt sicher, daß nur Prozeßschritte des gleichen Urrezeptes in eine Prozeßfolge gebracht werden (<2.2>). Des weiteren werden für ein Urrezept die notwendigen Eingangs- und Ausgangsstoffe benötigt. Dies wird durch die Beziehung *UrRepMat* zu dem Entitytyp *Input-Output-Kombination* sichergestellt. Die Struktur der Input-Output-Kombinationen wird weiter unten erläutert. Über die Beziehung *UrRepAnlage* werden die für die Prozeßdurchführung notwendigen Anlagentypen zugeordnet.

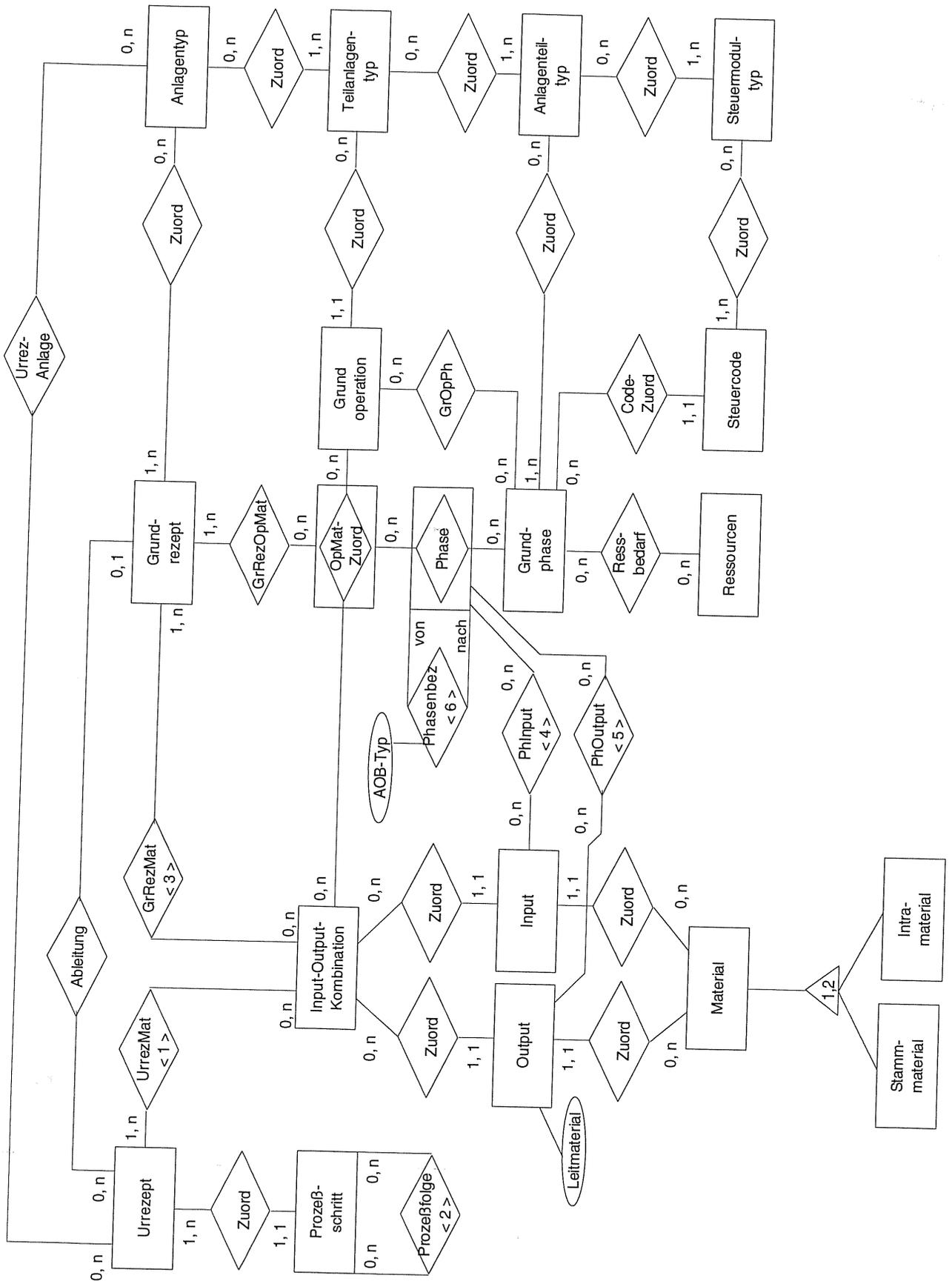
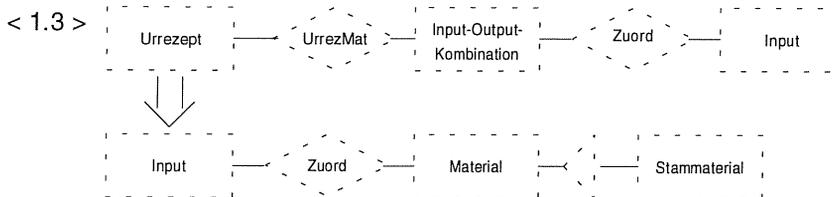
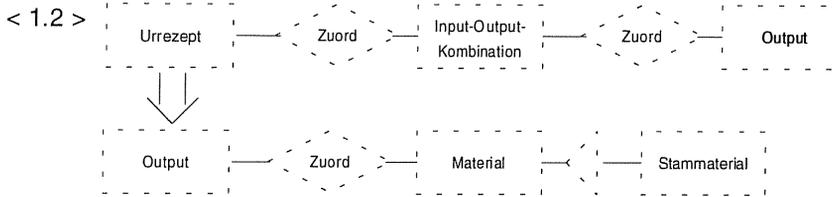
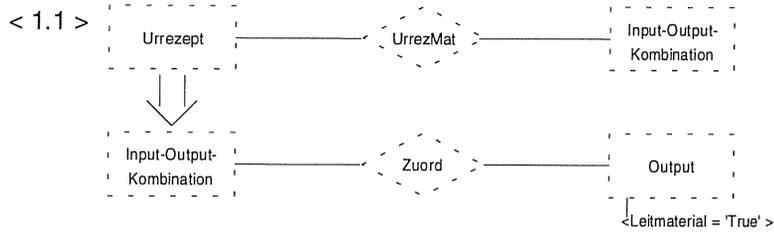
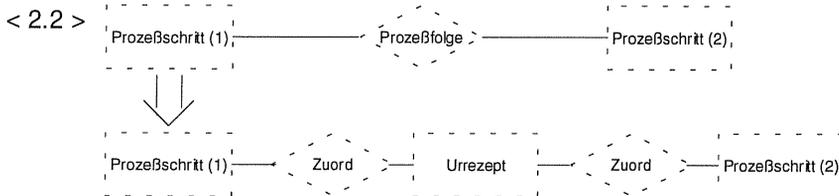
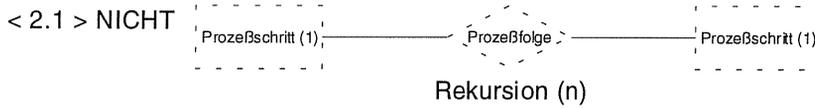


Abb. 5-1: Datenstrukturen der Rezeptverwaltung (PERM)

< 1 > UrrezMat

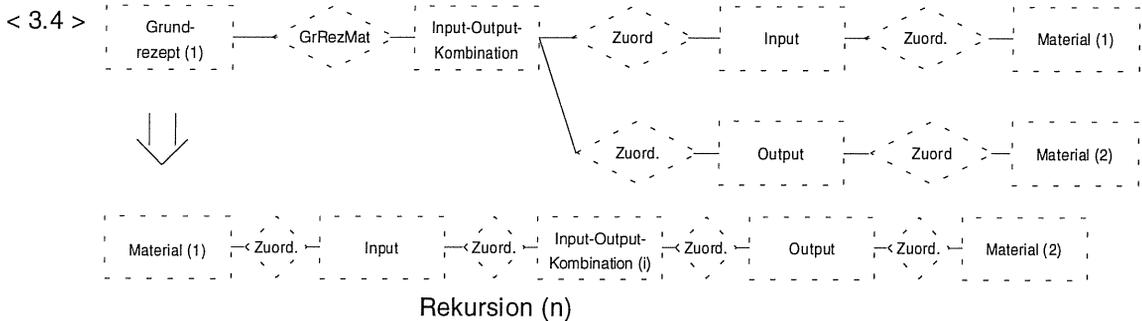


< 2 > Prozeßfolge



< 3 > GrRezMat

< 3.1 > - < 3.3 > analog < 1.1 > - < 1.3 >



UND

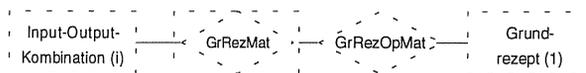
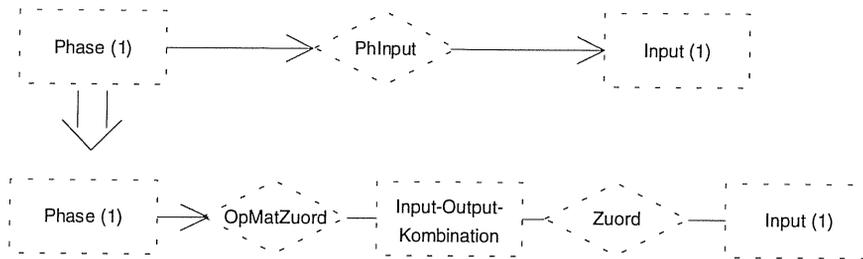


Abb. 5-2: Datenstrukturen der Rezeptverwaltung (RC-Diagramm)

< 4 > PhInput

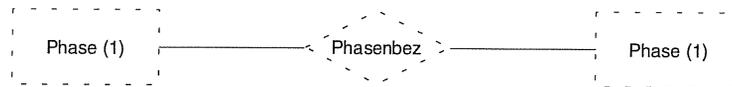


< 5 > PhOutput analog < 4 >

< 6 > Phasenbez

< 6.1 > AOB-Typ, von, nach $\rightarrow \emptyset$

< 6.2 > NICHT



< 6.3 >

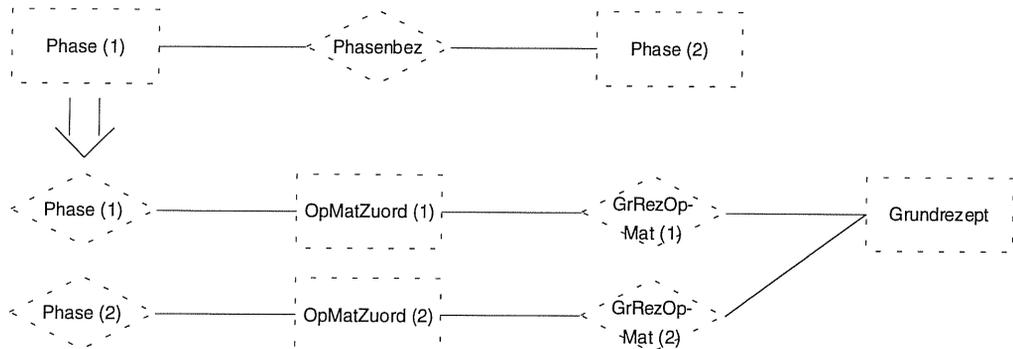


Abb. 5-3: Datenstrukturen der Rezeptverwaltung (RC-Diagramm)

Aus einem Urrezept können mehrere Grundrezepte abgeleitet werden, ein Grundrezept wird im allgemeinen aus einem Urrezept abgeleitet. Dies wird durch die Beziehung *Ableitung* dargestellt. Die Prozeßschritte eines Grundrezeptes werden durch Grundoperationen und Grundphasen abgebildet. Diese werden in eigenen Entitytypen beschrieben und können unabhängig von einem konkreten Grundrezept existieren. Ein Grundrezept referenziert auf bestehende Grundoperation, die ihrerseits auf Grundphasen referenzieren.

Um sowohl alternative Materialien als auch Netzstrukturen innerhalb eines Rezeptes zu realisieren, wird die Struktur *Input-Output-Kombination* eingeführt. Sie verbindet über die Entitytypen *Input* und *Output* alle Materialien eines Prozesses, sowohl die Eingangs- als auch die

Ausbringungsmaterialien. Bei einem solchen Prozeß kann es sich um ein Urrezept, ein Grundrezept oder eine Phase handeln. Einer Input-Output-Kombination werden über den Entitytyp *Input* alle Eingangsmaterialien (z.B. Rohstoffe und Katalysatoren) und über den Typ *Output* alle Ausbringungsmaterialien (z.B. Produkte und Katalysatoren, aber auch Abfallstoffe) zugeordnet. Damit erfüllt die Input-Output-Kombination die Aufgaben einer Baukastenstücklistenstruktur (vgl. *Scheer 90b*, S. 87). Ein Material kann sowohl ein Stammaterial (Typ *Stammaterial*) als auch ein Intraprozeßmaterial sein, für das kein eigener Materialstamm angelegt wird, da es nur zwischen zwei Prozeßschritten existiert und beispielsweise nicht gelagert wird (Typ *Intramaterial*). Sowohl einem Urrezept als auch einem Grundrezept werden über die Beziehungen *UrrezMat* bzw. *GrRezMat* jeweils eine oder mehrere Materialkombinationen zugeordnet. Sind mehrere Kombinationen einem Rezept zugeordnet, so sind diese als Alternativen anzusehen. In der Regel werden die Eingangsmaterialien variieren, z.B. bei Materialsubstitution. Eventuell können auch die Ausbringungsmaterialien variieren, insbesondere die Abfallstoffe oder die Kuppelprodukte in Abhängigkeit der eingesetzten Eingangsstoffe. Von der theoretischen Möglichkeit, die kompletten Ausbringungsmaterialien in einem Rezept zu variieren oder sogar alle Baukastenstücklisten eines Betriebs in einem Rezept zu definieren, wird man im allgemeinen absehen. Vielmehr wird man nur die Materialkombinationen mit gleichem Leitmaterial in einem Rezept zusammenfassen (Attribut *Leitmaterial* an Entitytyp *Output*). Die Integritätsbedingung <1> legt fest, daß eine einem Urrezept zugeordnete Materialkombination auch ein Leitmaterial als Ausbringung definiert hat <1.1> sowie daß jedes Ausbringungsmaterial <1.2> und jedes Eingangsmaterial <1.3> als Stammaterial definiert ist. Analoge Integritätsbedingungen gelten für die Materialzuordnung zu dem Grundrezept (<3.1> bis <3.3> der Beziehung *GrrezMat*).

Grundoperationen werden auch Materialkombinationen zugeordnet (Beziehung *OpMatZuord*). Im Gegensatz zu den Materialzuordnungen auf Rezeptebene können hierbei auch Intraprozeßmaterialien auftreten, sowohl als Eingangs- als auch als Ausbringungsmaterial. Häufig werden nur die letzten Operationen eines Rezeptes Stammaterialien hervorbringen und die ersten Operationen eines Rezeptes Stammaterialien verbrauchen. Eine bestimmte Materialkombination, d.h. die Transformation der Eingangs- in Ausbringungsmaterialien, kann durch verschiedene Grundoperationen geleistet werden. Sie sind als Operationsalternativen zu interpretieren und können sich innerhalb eines Rezeptes substituieren. Eine Operation kann aber auch verschiedenen Materialkombinationen zugeordnet werden. Dies ist immer dann der Fall, wenn der gleiche Prozeß auf unterschiedliche Materialien angewandt werden kann. Ein Grundrezept besteht aus einer Anzahl von Grundoperationen, die in Abhängigkeit von einer konkreten Materialtransformation einem Grundrezept zugeordnet werden. Deshalb wird die Rezept-Operationszuordnung *GrRezOpZuord* nicht an den Entitytyp *Grundoperation*, sondern an den uminterpretierten Beziehungstyp *OpMatZuord* geführt.

Die Verbindung und Reihenfolge der Operationen eines Grundrezeptes ergeben sich aus den jeweiligen Input- bzw. Outputmaterialien. Deckt sich das Outputmaterial einer Operation mit

dem Inputmaterial einer zweiten Operation, so können die beiden Operationen in einer Sequenz nacheinander abgearbeitet werden. Auf diese Weise kann die Folge der Operationen in einem Rezept traversiert werden. Des Weiteren können teilweise unabhängige Bearbeitungspfade definiert werden, was für Alternativen oder Parallelbearbeitung notwendig ist. In jedem Fall muß sichergestellt werden, daß die Input- und Outputmaterialien der Materialkombination des Grundrezeptes auch über eine beliebig lange Folge von Input-Output-Kombinationen der zugeordneten Operationen abgedeckt sind. Nur in diesem Fall ist ein Rezept auch auf Operationsebene konsistent. Dies wird durch die Integritätsbedingung <3.4> sichergestellt.

Den Urrezepten entsprechend werden auch den Grundrezepten Anlagentypen zugeordnet, auf denen sie bearbeitet werden. Im Gegensatz zu den Urrezepten muß ein Grundrezept aber mindestens einem Anlagentyp zugeordnet werden (Beziehung *Zuord* zwischen *Grundrezept* und *Anlagentyp*). Die Grundoperationen werden entsprechend der Anlagenstrukturierung auf genau einem Teilanlagentyp produziert (Beziehung *Zuord* zwischen *Grundoperation* und *Teilanlagentyp*).

Die Grundphasen sind über die Beziehung *GrOpPh* den Grundoperationen zugeordnet. Die Beziehung bestimmt, welche Grundphasen prinzipiell für die Bearbeitung einer Grundoperation geeignet sind. Welche Grundphasen in einem konkreten Grundrezept innerhalb einer Operation notwendig sind, wird durch die Beziehung *Phase* zwischen den Typen *Grundphase* und *OpMatZuord* festgelegt. Die Beziehung *Phase* wird nicht der Grundoperation zugeordnet, da diese in einem konkreten Grundrezept mehrmals vertreten sein kann. Die auszuführenden Phasen einer Operation eines Grundrezeptes sind sequentiell abzuarbeiten. Die Ordnung wird implizit durch eine laufende Nummer definiert. Darüber hinaus bestehen zeitliche Abhängigkeiten zwischen Phasen unterschiedlicher Operationen des gleichen Grundrezeptes. Diese Abhängigkeiten ergeben sich zum einen aus dem Materialfluß zwischen den Operationen. Beispielsweise kann es notwendig sein, das Entladen einer Teilanlage als letzte Phase einer Operation parallel zu dem Beladen als erste Phase der folgenden Operation durchzuführen. Aus der Verknüpfung der Operationen über die Input-Output-Kombinationen lassen sich nur die Reihenfolge der Operationen ableiten, nicht jedoch das zeitliche Verhältnis zueinander. Deshalb wird die Beziehung *Phasenbez* eingeführt, die verschiedene Phasen rekursiv verbindet. Diese Verknüpfungen stellen Anordnungsbeziehungen in Sinne der Netzplantechnik dar und können mit minimalen und maximalen Übergangszeiten spezifiziert werden. Entsprechend der Netzplantechnik können die Typen Ende-Start, Start-Start, Ende-Ende und Start-Ende unterschieden werden (Attribut *AOB-Typ*). So kann man für die Verknüpfung der Entlade- mit der Beladephase eine Start-Start- und eine Ende-Ende-Beziehung, jeweils mit einer minimalen und einer maximalen Übergangszeit von null definieren. Zum anderen kann es notwendig sein, zusätzliche zeitliche Bedingungen ohne direkten Materialfluß zu formulieren. So kann beispielsweise bei einem sauerstoffempfindlichen Stoff eine maximale Dauer von der Entnahme der Rohstoffe aus der Verpackung bis zu einem bestimmten Prozeßschritt vorgeben sein. Dies läßt

sich durch eine Start-Ende-Beziehung mit maximaler Übergangszeit zwischen der Einfüllphase und der Prozeßphase über mehrere Operationen hinweg formulieren.

Die oben bereits erwähnten Sequenzen der Phasen in einer Operation stellen Anordnungsbeziehungen vom Typ Ende-Start mit maximalen Übergangszeiten von null dar. Auf ihre explizite Formulierung kann verzichtet werden.

Die Integritätsbedingung der Beziehung *Phasebez* (<6>) gestattet die Formulierung unterschiedlicher Typen von Anordnungsbeziehungen zwischen den gleichen Phasen (<6.1>), schließt Zyklen in der Bearbeitung der Phasen aus (<6.2>) und stellt sicher, daß nur zwischen den Phasen zeitliche Bedingungen definiert werden, die zum gleichen Grundrezept gehören (<6.3>).

Den einzelnen Phasen können neben den Anlagenteiltypen weitere Ressourcen wie Personal, Transportmittel oder Vorrichtungen zugeordnet werden (Zuordnung *RessBedarf* zum Typ *Ressourcen*). Durch die Zuordnung auf Phasenebene kann im Rahmen der Terminierung und Prozeßsteuerung eine zeitgerechte Ressourcenzusteuering erfolgen. Die zeitgenaue Zusteuering ist allerdings auch für Material notwendig, weshalb die Phasen über die Beziehungen *PhInput* bzw. *PhOutput* mit den jeweiligen Eingangs- und Ausbringungstoffen verbunden werden. Die Bedingungen <4> und <5> stellen sicher, daß nur solche Materialien referenziert werden, die bereits über die Input-Output-Kombination der Operation festgelegt sind.

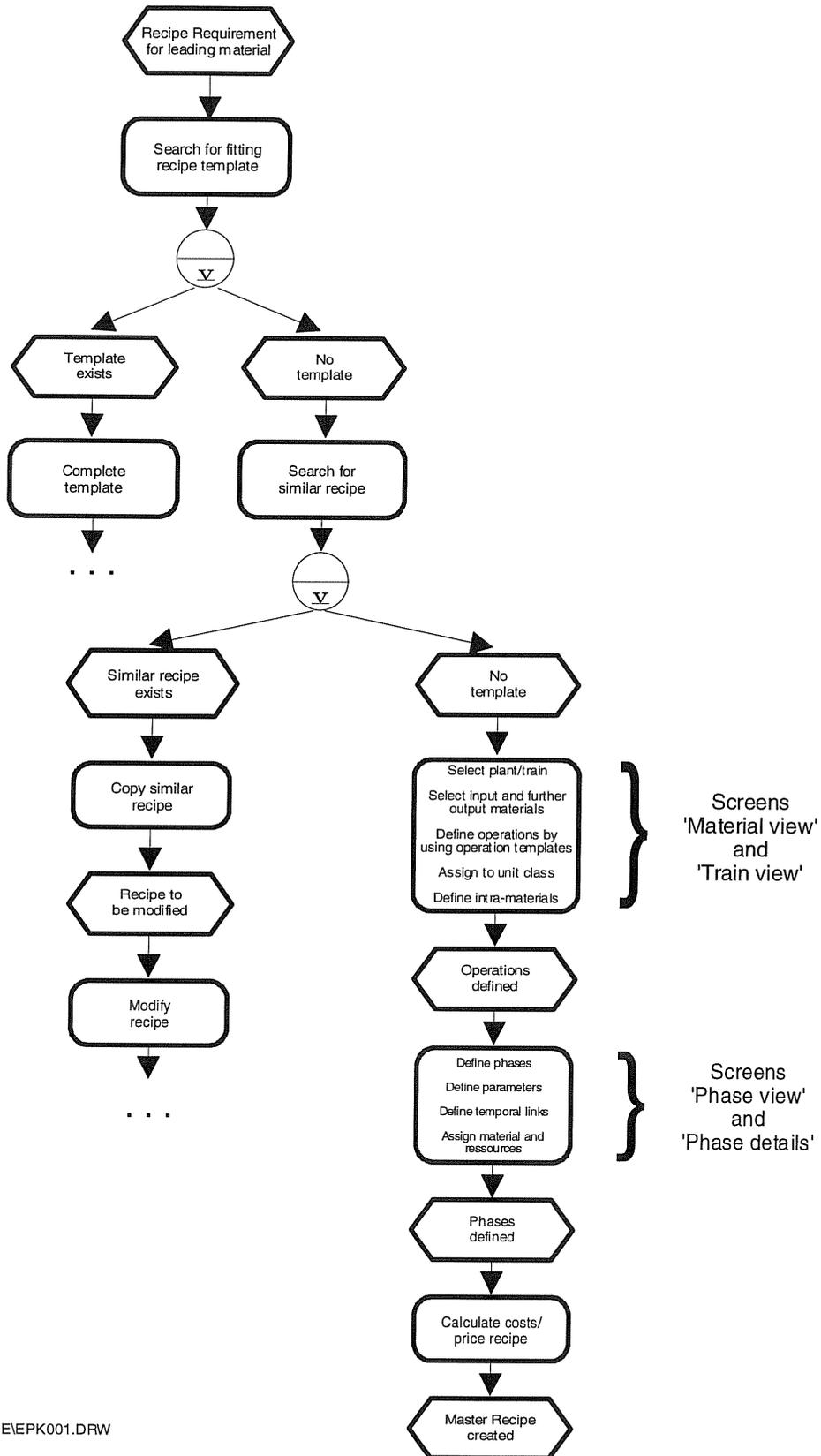
Bei einer automatisierten Produktion können Phasen durch Steuercodes weiter beschrieben werden, die die Anweisungen für das Fahren von Anlagen beinhalten. Die Steuercodes sind einem oder mehreren Steuermoduln, den kleinsten Bausteinen der Anlagenhierarchie, zugeordnet (Zuordnung *Zuord* zwischen den Typen *Steuercode* und *Steuermodultyp*).

6. Funktionen und Benutzeroberfläche

Nach der Definition der Datenstrukturen sollen beispielhaft die wichtigsten Funktionen und Masken der graphischen Benutzerschnittstellen (GUI, Graphical User Interface) gezeigt werden. Dabei wird sich auf die Vorgehensweise bei der Neudefinition eines Grundrezeptes beschränkt. Diese Funktion kann beispielsweise durch einen Produktionsauftrag (Trigger 8 in Abbildung 3) ausgelöst werden. Eine entsprechende Prozeßkette zur Definition ist in Abbildung 6 dargestellt (zur Methode der Prozeßkette vgl. *Keller/Nüttgens/Scheer 92* und *Scheer 93*).

Ein Grundrezept wird in der Regel durch Konkretisierung eines Urrezeptes erzeugt. Der Einstieg für die Erstellung ist die Definition des herzustellenden Materials, des Leitmaterials. Zu diesem können vorhandene Urrezepte gesucht werden, die als Rahmen für die weitere Verfeinerung dienen.

Definition of Master Recipe



H:\CAPISCE\GRACE\EPK001.DRW

Abb. 6: Prozeßkette zur Grundrezeptverwaltung

Dies wird durch den linken Ast der Prozeßkette dargestellt. Die Funktion wird als *Search for fitting recipe template* bezeichnet, da anstelle von Urrezepten auch unvollständige Grundrezepte als Rahmen dienen können. Sie erfüllen damit die gleichen Funktionen wie die Urrezepte. Gemeinsam werden sie als Rezepttemplates bezeichnet. Eine andere Möglichkeit besteht darin, ein Rezept aus einem bestehenden Grundrezept zu erzeugen. In diesem Fall wird eine Kopie erzeugt, die anschließend geändert wird (linker unterer Ast der Prozeßkette). Schließlich kann ein Grundrezept ohne Template komplett neu definiert werden. Dieser Fall ist im rechten Ast der Prozeßkette abgebildet.

6.1. Materialflußsicht

Die Definition der Materialien kann an einer Maske entsprechend der Abbildung 7 erfolgen. Nach der Eingabe des Leitmaterials kann die Festlegung der Eingangs- und weiterer Ausbringungsmaterialien erfolgen. Die Operationen werden innerhalb der gleichen Übersicht eingefügt. Dazu kann auf die bestehenden Grundoperationen referenziert werden. Diese werden, in Analogie zu der Verallgemeinerung auf Rezeptebene, in der Prozeßkette als *operation template* bezeichnet und können Operationen anderer Rezepte beinhalten. Daneben können auch neue Grundoperationen definiert werden.

Die Materialien und Operationen werden durch Materialflüsse miteinander verbunden, die durch gerichtete Pfeile dargestellt werden. In Abbildung 7 ist ein Rezept rec001 zur Produktion des Materials M5 dargestellt, in dem zwei parallele Bearbeitungspfade existieren. Im ersten Pfad wird aus dem Material M1 durch Operation Op1 das Material M2 und anschließend durch Operation Op2 das Material M4 hergestellt. M4 ist ein Intraprozeßmaterial und erscheint nicht im Materialstamm. Parallel dazu wird das Eingangsmaterial M3 durch Operation Op4 bearbeitet. Die Operation erzeugt ein Kuppel- oder Abfallprodukt M6 sowie ein nicht benanntes Zwischenprodukt, das zusammen mit M4 von der Operation Op3 zum Endprodukt M5 verarbeitet wird. Aus diesem Sachverhalt lassen sich die genannten Materialien M1, M2, M3, M4, M5, M6 sowie das nicht benannte Zwischenmaterial (im folgenden M8) ableiten. Auch das Material M8 wird als nichtanzuzeigendes Intramaterial in die Entitytypen *Material* bzw. *Intramaterial* (vgl. Abbildung 5) aufgenommen, um damit die Materialkombinationen M1-M2, M2-M3, M3-M5M8 und M4M8-M5 ableiten zu können (Entitytyp *Input-Output-Kombination*). Aus den abgebildeten Operation und deren Position innerhalb des Materialflusses lassen sich die Typen *Grundoperation* (soweit nicht bereit vorhanden) sowie *OpMatZuord* ableiten. Weiterhin kann die Materialkombination M1M3-M5M6 als Input-Output-Kombination des Grundrezeptes generiert und über die Beziehung *GrRezMat* dem Grundrezept zugeordnet werden.

Daneben sind in der graphischen Übersicht Alternativen angegeben. Die Operation Op5 substituiert die Operation Op2. Dies führt zu weiteren Objekten der Typen *OpMatZuord* und gegebenenfalls *Grundoperation*. Eine neue Materialkombination ist nicht erforderlich.

Operation Op6 kann Operation Op4 ersetzen. Im Gegensatz zu der vorherigen Alternative müssen hier allerdings auch zusätzliche Materialkombinationen eingeführt werden, da das Material M7 das Material M3 ersetzt. Daraus folgt eine neue Materialkombination, wobei das Ausbringungsmaterial nicht benannt ist. Da es wie das bereits erzeugte, nicht-benannte Intramaterial M8 als Eingangsmaterial für Operation Op3 dient, kann auch der Materialfluß zwischen Op6 und Op3 mit Material M8 bezeichnet werden. Die neue Materialkombination für die Operationsebene lautet deshalb M7-M8. Zusätzlich muß für das Grundrezept eine neue Materialkombination eingeführt werden, da bei diesem alternativen Pfad ein Eingangsmaterial ersetzt und als Konsequenz das Kuppel- oder Abfallprodukt M6 entfällt. Dies führt zur Kombination M1M7-M5.

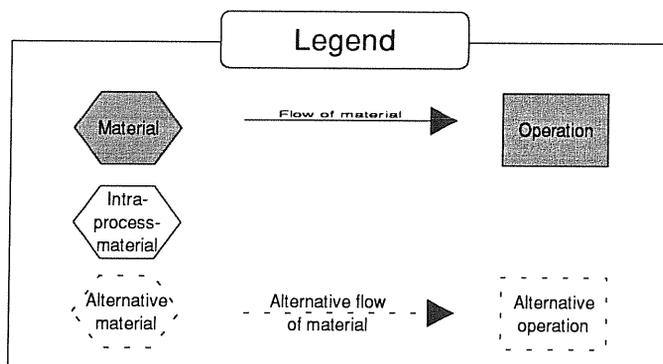
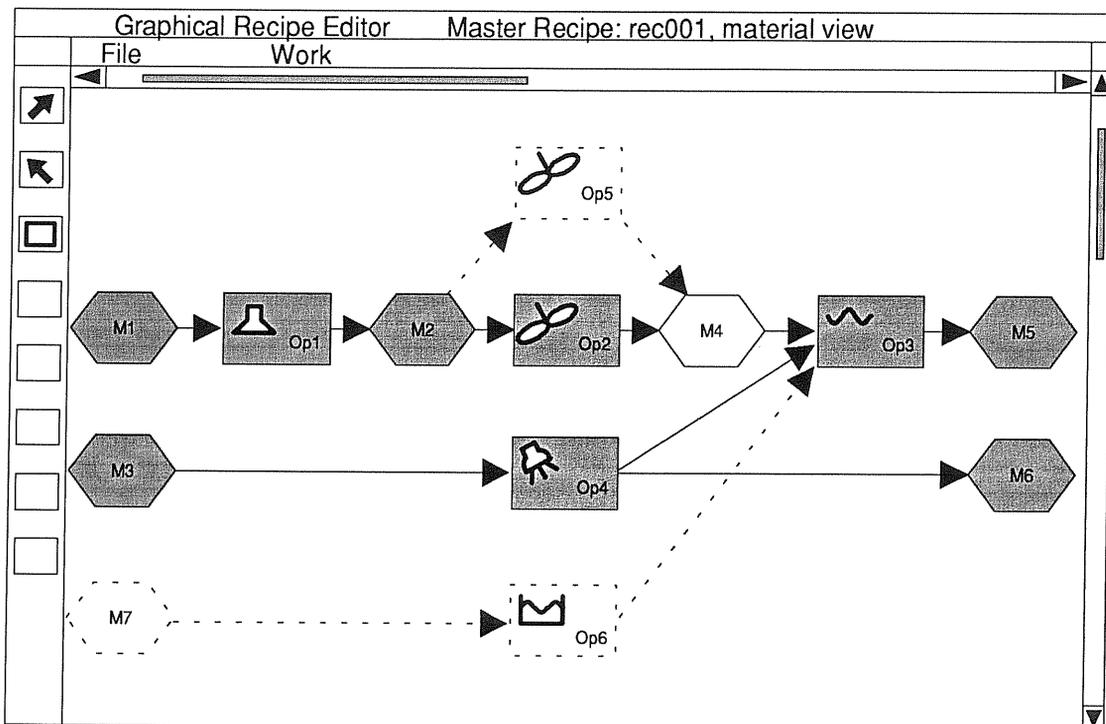


Abb. 7: Bildschirmmaske zum Materialfluß

6.2. Anlagensicht

Die Zuordnung zu den Anlagen kann mit Hilfe einer Anlagenübersicht erfolgen. Die Anlagenübersicht ist entsprechend der Anlagenstrukturen hierarchisch aufgebaut. Auf Rezeptebene erfolgt die Zuordnung auf Anlagentypenebene. In diesem Fall werden in der Anlagenübersicht alle Anlagentypen dargestellt. Durch Mausselektion erfolgt die Zuordnung zu dem Rezept (Beziehung *Zuord* zwischen den Typen *Grundrezept* und *Anlagentyp*). Nach der Selektion des Anlagentyps werden alle Teilanlagentypen, aus denen der Anlagentyp besteht, angezeigt. Eine solche Übersicht ist in Abbildung 8 dargestellt. Hier erfolgt eine Zuordnung zu den einzelnen Operationen .

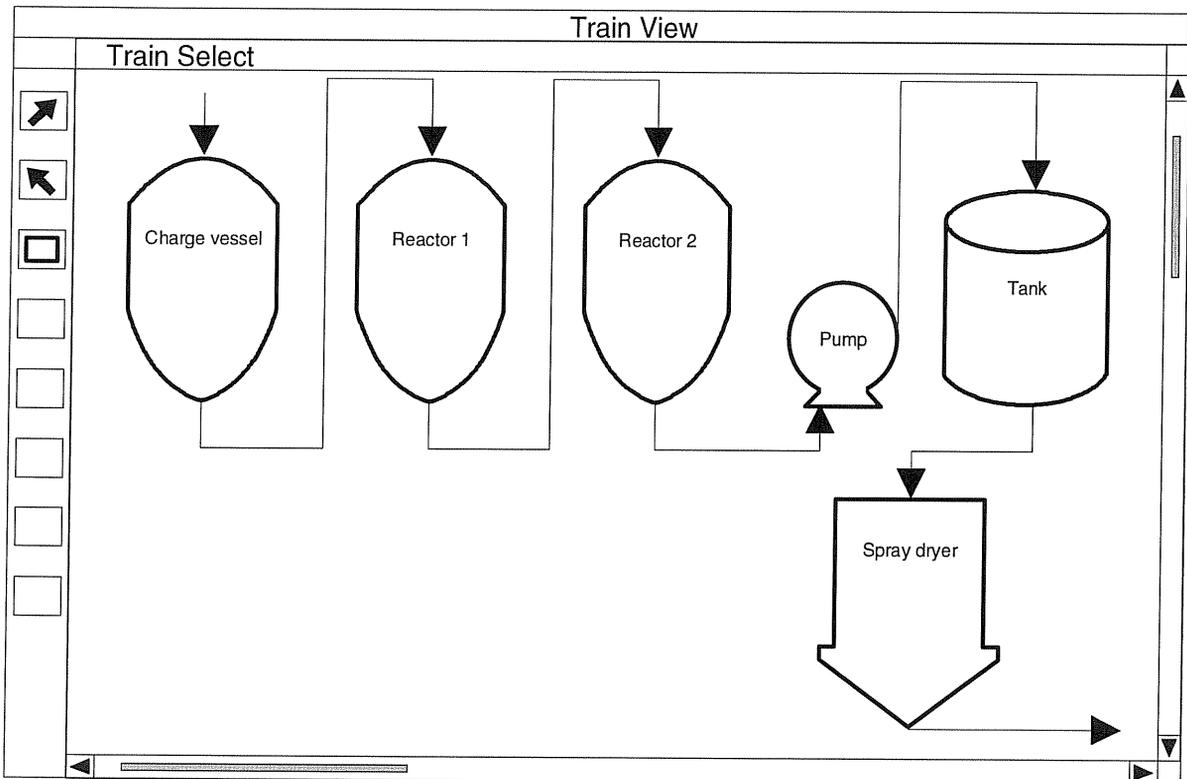


Abb. 8: Bildschirmmaske zum Anlagenlayout

6.3. Phasensicht

Nach dieser Grobdefinition erfolgt die Verfeinerung des Rezepts in der Maske Phasenübersicht (vgl. Abbildung 9). In dieser Übersicht werden nur die Operationen eines möglichen Pfades des Rezepts gezeigt, in diesem Beispiel die in Abbildung 7 dunkel dargestellten Operationen. Die Phasenübersicht zeigt pro Operation alle beteiligten Phasen. Die Darstellung der Phasen kann graphisch nach bestimmten Typen unterschieden werden, hier die sonstigen Phasen, die Rüstphasen (set-up) und die Wartephasen (waiting point), die eine Unterbrechung des Produktionsprozesses gestatten. Die Darstellung der Phasenlängen erfolgt zeitproportional zu der am

oberen Bildschirmrand dargestellten Zeitachse. Als Phasenlänge können minimale, maximale oder durchschnittliche Durchlaufzeiten gewählt werden. Diese Darstellungsart lehnt sich an die Gantt-Darstellung von Fertigungsleitständen an. Neben den Phasen werden in dieser Übersicht auch die zeitlichen Abhängigkeiten zwischen den Phasen (temporal links) definiert. Sie repräsentieren die Beziehungen *Phasenbez* des Datenmodells in Abbildung 5. Die Beziehungen zwischen der letzten Phase von Operation Op1 und der dritten Phase von Operation Op2 sind typisch für die bereits oben erläuterten parallelen Entlade-Belade-Beziehungen ohne Zeitdauer.

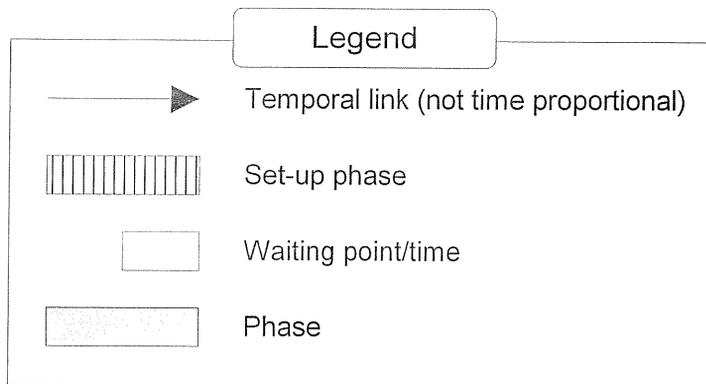
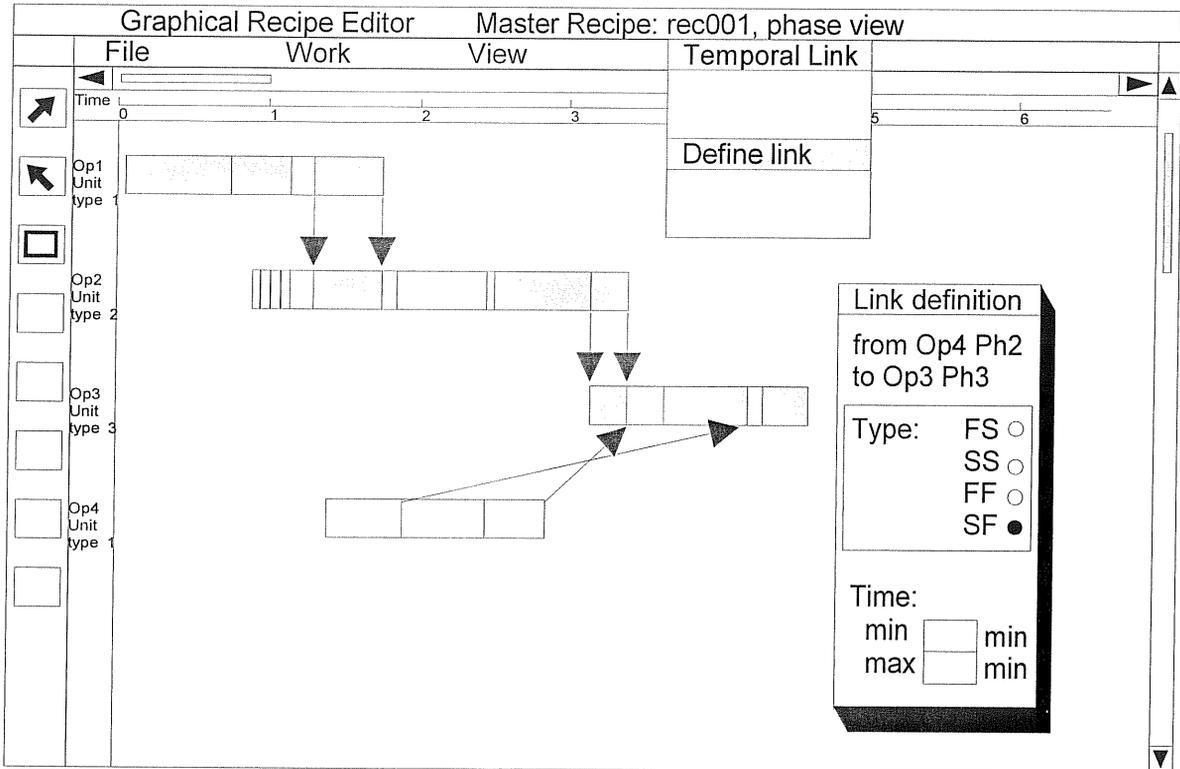


Abb. 9: Bildschirmmaske zu zeitlichen Beziehungen

Das Gleiche gilt für die beiden Beziehungen zwischen den Operationen Op2 und Op3. Auch die Beziehung zwischen der dritten Phase von Operation Op4 und der zweiten Phase von Operation Op3 repräsentieren wie die vorhergenannten Beziehungen die Materialflüsse des Rezepts, wobei hier allerdings keine Parallelität gefordert wird. Die Beziehung zwischen der zweiten Phase von Operation Op4 und der dritten Phase von Operation Op3 stellt keinen direkten Materialfluß dar, sondern eine zusätzliche zeitliche Restriktion bezüglich der maximalen Zeitdauer der dazwischenliegenden Phasen, deren aufsummierte maximale Durchlaufzeiten durchaus größer sein können. Die Beziehung ist vom Anordnungsbeziehungstyp Start-Ende.

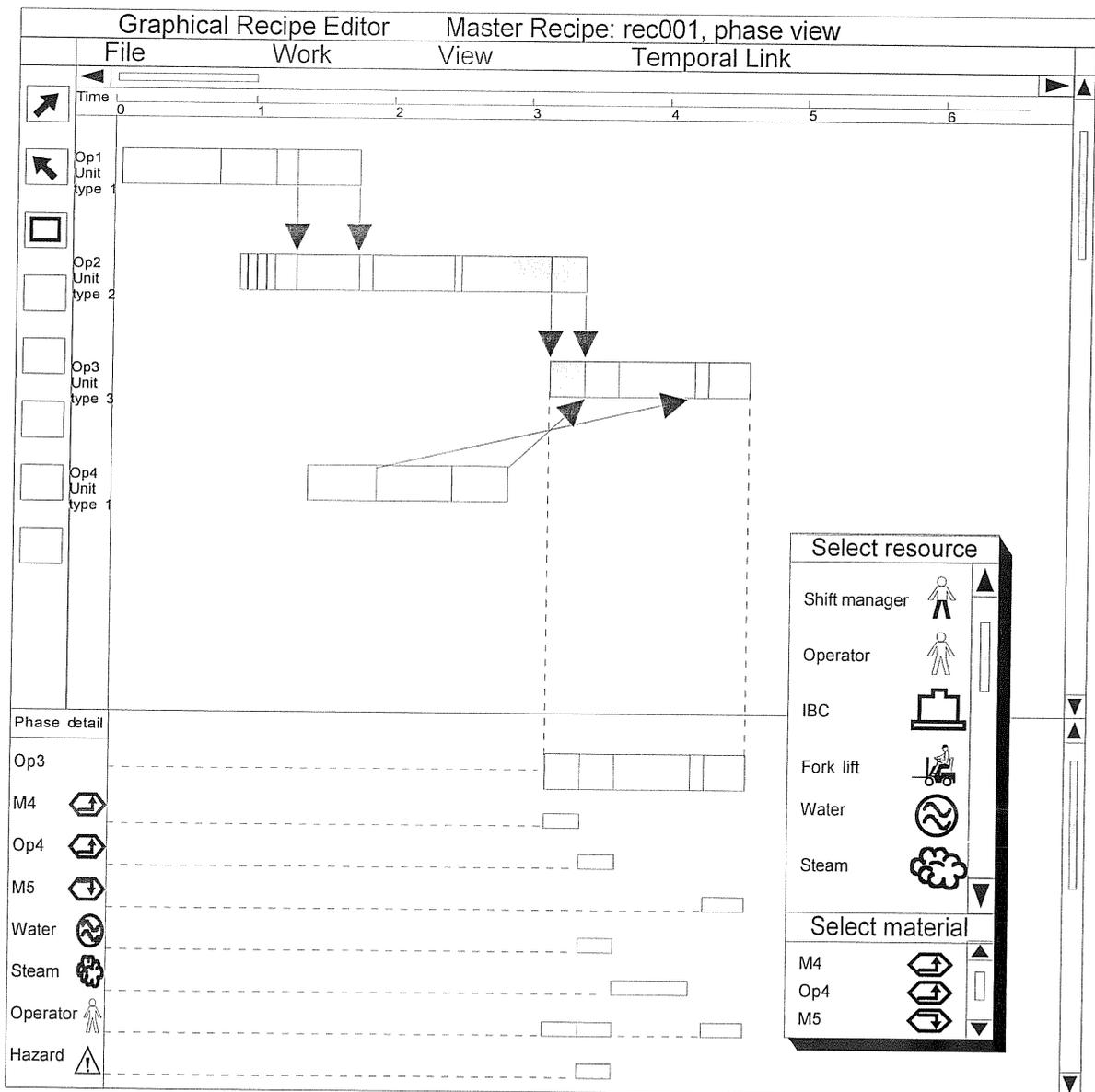


Abb. 10: Bildschirmmaske zum Phasendetail

Die Phasen werden anschließend weiterspezifiziert. Falls nicht auf bestehende Grundphasen referenziert werden kann, müssen diese, einschließlich der Steuercodes und der Zuordnungen zu Anlagenteilen bzw. Steuermodultypen, angelegt werden. Den Phasen werden die benötigten Ressourcen sowie die Materialien der der Operationen zugeordneten Input-Output-Kombinationen zugeordnet. Dazu wird eine Scrollbox aufgeklappt (vgl. Abbildung 10), die zum einen alle in dem Anlagentyp verfügbaren Ressourcen (obere Teil der Scrollbox) enthält sowie alle Materialien der Input-Output-Kombination. In Fall der Operation Op3 sind dies die Eingangsmaterialien M4 und das nicht-benannte Material aus Operation Op4, das intern als M8 bezeichnet wurde sowie das Ausgangsmaterial M5. Die Zuordnung kann durch eine einfache Drag and Drop-Funktion realisiert werden. Das Material M4 wird kontinuierlich der ersten Phase zugeführt, z.B. ein permanenter Materialfluß durch langsames Einpumpen. Das Material der Operation Op4 wird dagegen dem Start der zweiten Phase zugeordnet, was darauf hinweist, daß das Material komplett zu Beginn der Phase bereitzustellen ist oder der genaue Zeitpunkt nicht bekannt ist.

7. Ausblick

Das aufgezeigte Konzept verbindet Rezeptdatenstrukturen mit einer graphischen Benutzeroberfläche, wie sie heute von modernen Workstations bereitgestellt werden. Die Ideen des Konzeptes lassen sich auch auf andere Bereiche von operativen Systemen übertragen, beispielsweise auf die Verwaltung von Anlagedaten (Maschinen und Kostenstellen) oder auf die Arbeitsplanerstellung für die diskrete Fertigung. Durch die Informationsrepräsentation kann die Integration zu anderen CIP-Komponenten unterstützt werden. So wird beispielsweise durch die Phasendarstellung ein fließender Übergang zu der Gantt-Darstellung der Produktionssteuerung geschaffen.

Weiterentwicklungspotentiale der Konzeption liegen beispielsweise in der

- Integration von regelbasiertem Wissen zur Unterstützung der Vorgabezeitermittlung und der Plausibilitätsprüfungen,
- Integration von Kostenrechnungsinformationen, z.B. für eine begleitende Kalkulation,
- Integration von multimedialen Funktionen, z.B. von Anlagedaten.

Literaturverzeichnis

Hofmann 92

Hofmann, M.: PPS - nichts für die chemische Industrie?, in: *io Management*, 61(1992)1, S. 30-33.

Keller/Nüttgens/Scheer 92

Keller, G.; Nüttgens, M.; Scheer, A.-W.: Semantische Prozeßmodellierung auf der Grundlage Ereignis-gesteuerter Prozeßketten (EPK), in: Scheer, A.-W. (Hrsg.), *Veröffentlichungen des Institutes für Wirtschaftsinformatik*, Heft 89, Saarbrücken Januar 1992.

Jost 92

Jost, W.: *Rechnergestützte CIM-Rahmenplanung - Konzeption und Realisierung eines Werkzeuges zur Analyse und Planung von CIM-Systemen*, Dissertation Universität des Saarlandes 1992.

Loos 92

Loos, P.: *Datenstrukturierung in der Fertigung*, München-Wien 1992.

Loos 93

Loos, P.: Planungshierarchien für die dezentrale Auftragsabwicklung in der Prozeßindustrie, in: *Vernetzte Dezentralisierung - Neue Ansätze für die Auftragsabwicklung* (Tagungsband, Saarbrücken, 26.-27. April 1993).

Luber 92

Luber, A.: How to Identify a True Process Industry Solution, in: *Production and Inventory Management* 12(1992)2, S.16-17.

NE33

NAMUR Empfehlung NE33, Anforderungen an Systeme zur Rezeptfahrweise, Mai 1992.

Polke 89

Polke, M.: CIP in der Verfahrensindustrie, in: *CIM Management*, 5(1989)5, S. 34-35.

Reinhardt/Schrieber/ Strozyk 93

Reinhardt, A.; Schrieber, R.; Strozyk, W.: Objektorientiertes grafisches Konfigurieren von Automatisierungssystemen mit PROGRAF AS, in: *atp - Automatisierungstechnische Praxis* 35(1993)4, S. 237-241.

Scheer 90a

Scheer, A.-W.: *CIM - Der computergesteuerte Industriebetrieb*, 4.A., Berlin et al. 1990.

Scheer 90b

Scheer, A.-W.: *Wirtschaftsinformatik - Informationssysteme im Industriebetrieb*, 3.A., Berlin et al. 1990.

Scheer 93

Scheer, A.-W.: *ARIS Methodenbuch - Fachkonzept*, Saarbrücken 1993.

Scheer/Loos 93

Scheer, A.-W.; Loos, P.: *Advanced Architectures for Shop-Floor Control Systems*, in: *Ingénierie des Systèmes d'Information*, 1993, (in Vorbereitung).

SP88

ISA-dS88.01, *Batch Control Systems - Models and Terminology*, Draft 5, Dezember 1992.

Uhlig 87

Uhlig, R. J.: Erstellen von Ablaufsteuerungen für Chargenprozesse mit wechselnden Rezepturen, in: *atp - Automatisierungstechnische Praxis*, 29(1987)1, S. 17-23.

Zörntlein 88

Zörntlein, G.: *Flexible Fertigungssysteme: Belegung, Steuerung und Datenorganisation*, München-Wien 1988.

Die Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi) im Institut für empirische Wirtschaftsforschung an der Universität des Saarlandes erscheinen in unregelmäßiger Folge.

* Die Hefte 1 - 31 werden nicht mehr verlegt.

- Heft 32: A.-W. Scheer: Einfluß neuer Informationstechnologien auf Methoden und Konzepte der Unternehmensplanung, März 1982, Vortrag anläßlich des Anwendergespräches "Unternehmensplanung und Steuerung in den 80er Jahren in Hamburg vom 24. - 25.11.1981
- Heft 33: A.-W. Scheer: Disposition- und Bestellwesen als Baustein zu integrierten Warenwirtschaftssystemen, März 1982, Vortrag anläßlich des gdi-Seminars "Integrierte Warenwirtschafts-Systeme" in Zürich vom 10. - 12. Dezember 1981
- Heft 34: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS - Ein Ansatz zur Entwicklung prüfungsgerechter Software-Systeme, Mai 1982
- Heft 35: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Konzept einer computergestützten Prüfungsumgebung, Juli 1982
- Heft 36: A.-W. Scheer: Rationalisierungserfolge durch Einsatz der EDV - Ziel und Wirklichkeit, August 1982, Vortrag anläßlich der 3. Saarbrücker Arbeitstagung "Rationalisierung" in Saarbrücken vom 04. - 06. 10.1982
- Heft 37: A.-W. Scheer: DV-gestützte Planungs- und Informationssysteme im Produktionsbereich, September 1982
- Heft 38: A.-W. Scheer: Interaktive Methodenbanken: Benutzerfreundliche Datenanalyse in der Marktforschung, Mai 1983
- Heft 39: A.-W. Scheer: Personal Computing - EDV-Einsatz in Fachabteilungen, Juni 1983
- Heft 40: A.-W. Scheer: Strategische Entscheidungen bei der Gestaltung EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anläßlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 41: H. Krcmar: Schnittstellenprobleme EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anläßlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 42: A.-W. Scheer: Factory of the Future, Vorträge im Fachausschuß "Informatik in Produktion und Materialwirtschaft" der Gesellschaft für Informatik e. V., Dezember 1983
- Heft 43: A.-W. Scheer: Einführungsstrategie für ein betriebliches Personal-Computer-Konzept, März 1984
- Heft 44: A.-W. Scheer: Schnittstellen zwischen betriebswirtschaftlicher und technische Datenverarbeitung in der Fabrik der Zukunft, Juli 1984
- Heft 45: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Ein Werkzeug zur Messung der Qualität von Software-Systemen, August 1984
- Heft 46: H. Krcmar: Die Gestaltung von Computer am-Arbeitsplatz-Systemen - ablauforientierte Planung durch Simulation, August 1984
- Heft 47: A.-W. Scheer: Integration des Personal Computers in EDV-Systeme zur Kostenrechnung, August 1984

- Heft 48: A.-W. Scheer: Kriterien für die Aufgabenverteilung in Mikro-Mainframe Anwendungssystemen, April 1985
- Heft 49: A.-W. Scheer: Wirtschaftlichkeitsfaktoren EDV-orientierter betriebswirtschaftlicher Problemlösungen, Juni 1985
- Heft 50: A.-W. Scheer: Konstruktionsbegleitende Kalkulation in CIM-Systemen, August 1985
- Heft 51: A.-W. Scheer: Strategie zur Entwicklung eines CIM-Konzeptes - Organisatorische Entscheidungen bei der CIM-Implementierung, Mai 1986
- Heft 52: P. Loos, T. Ruffing: Verteilte Produktionsplanung und -steuerung unter Einsatz von Mikrocomputern, Juni 1986
- Heft 53: A.-W. Scheer: Neue Architektur für EDV-Systeme zur Produktionsplanung und -steuerung, Juli 1986
- Heft 54: U. Leismann, E. Sick: Konzeption eines Bildschirmtext-gestützten Warenwirtschaftssystems zur Kommunikation in verzweigten Handelsunternehmen, August 1986
- Heft 55: D. Steinmann: Expertensysteme (ES) in der Produktionsplanung und -steuerung (PPS) unter CIM-Aspekten, November 1987, Vortrag anlässlich der Fachtagung "Expertensysteme in der Produktion" am 16. und 17.11.1987 in München
- Heft 56: A.-W. Scheer: Enterprise wide Data Model (EDM) as a Basis for Integrated Information Systems, Juli 1988
- Heft 57: A.-W. Scheer: Present Trends of the CIM Implementation (A qualitative Survey) Juli 1988
- Heft 58: A.-W. Scheer: CIM in den USA - Stand der Forschung, Entwicklung und Anwendung, November 1988
- Heft 59: R. Herterich, M. Zell: Interaktive Fertigungssteuerung teilautonomer Bereiche, November 1988
- Heft 60: A.-W. Scheer, W. Kraemer: Konzeption und Realisierung eines Expertenunterstützungssystems im Controlling, Januar 1989
- Heft 61: A.-W. Scheer, G. Keller, R. Bartels: Organisatorische Konsequenzen des Einsatzes von Computer Aided Design (CAD) im Rahmen von CIM, Januar 1989
- Heft 62: M. Zell, A.-W. Scheer: Simulation als Entscheidungsunterstützungsinstrument in CIM, September 1989
- Heft 63: A.-W. Scheer: Unternehmens-Datenbanken - Der Weg zu bereichsübergreifenden Datenstrukturen, September 1989
- Heft 64: C. Berkau, W. Kraemer, A.-W. Scheer: Strategische CIM-Konzeption durch Eigenentwicklung von CIM-Modulen und Einsatz von Standardsoftware, Dezember 1989
- Heft 65: A. Hars, A.-W. Scheer: Entwicklungsstand von Leitständen^[1], Dezember 1989
- Heft 66: W. Jost, G. Keller, A.-W. Scheer: CIMAN - Konzeption eines DV-Tools zur Gestaltung einer CIM-orientierten Unternehmensarchitektur, März 1990

- Heft 67: A.-W. Scheer: Modellierung betriebswirtschaftlicher Informationssysteme (Teil 1: Logisches Informationsmodell), März 1990
- Heft 68: W. Kraemer: Einsatzmöglichkeiten von Expertensystemen in betriebswirtschaftlichen Anwendungsgebieten, März 1990
- Heft 69: A.-W. Scheer, R. Bartels, G. Keller: Konzeption zur personalorientierten CIM-Einführung, April 1990
- Heft 70: St. Spang, K. Ibach: Zum Entwicklungsstand von Marketing-Informationssystemen in der Bundesrepublik Deutschland, September 1990
- Heft 71: D. Aue, M. Baresch, G. Keller: **URMEL**, Ein **U**nte**R**nehmens**M**odellierungsansatz, Oktober 1990
- Heft 72: M. Zell: Datenmanagement simulationsgestützter Entscheidungsprozesse am Beispiel der Fertigungssteuerung, November 1990
- Heft 73: A.-W. Scheer, M. Bock, R. Bock: Expertensystem zur konstruktionsbegleitenden Kalkulation, November 1990
- Heft 74: R. Bartels, A.-W. Scheer: Ein Gruppenkonzept zur CIM-Einführung, Januar 1991
- Heft 75: M. Nüttgens, St. Eichacker, A.-W. Scheer: CIM-Qualifizierungskonzept für Klein- und Mittelunternehmen (KMU), Januar 1991
- Heft 76: Ch. Houy, J. Klein: Die Vernetzungsstrategie des Instituts für Wirtschaftsinformatik - Migration vom PC-Netzwerk zum Wide Area Network (noch nicht veröffentlicht)
- Heft 77: W. Kraemer: Ausgewählte Aspekte zum Stand der EDV-Unterstützung für das Kostenmanagement: Modellierung benutzerindividueller Auswertungssichten in einem wissensbasierten Controlling-Leitstand, Mai 1991
- Heft 78: H. Heß: Vergleich von Methoden zum objektorientierten Design von Softwaresystemen, August 1991
- Heft 79: A.-W. Scheer: Konsequenzen für die Betriebswirtschaftslehre aus der Entwicklung der Informations- und Kommunikationstechnologien, Mai 1991
- Heft 80: G. Keller, J. Kirsch, M. Nüttgens, A.-W. Scheer: Informationsmodellierung in der Fertigungssteuerung, August 1991
- Heft 81: A.-W. Scheer: Papierlose Beratung - Werkzeugunterstützung bei der DV-Beratung, August 1991
- Heft 82: C. Berkau: VOKAL (System zur Vorgangskettendarstellung und -analyse), Teil 1: Struktur der Modellierungsmethode - Dezember 1991 (wird nicht verlegt)
- Heft 83: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - Theoretical Foundations - 1991
- Heft 84: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - A Survey - 1991
- Heft 85: W. Hoffmann, M. Nüttgens, A.-W. Scheer, St. Scholz: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 1: Produktionsplanung), Oktober 1991

- Heft 86: A.-W. Scheer: Koordinierte Planunginseln: Ein neuer Lösungsansatz für die Produktionsplanung, November 1991
- Heft 87: M. Nüttgens, G. Keller, S. Stehle: Konzeption hyperbasierter Informationssysteme, Dezember 1991
- Heft 88: W. Hoffmann, B. Maldener, M. Nüttgens, A.-W. Scheer: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 2: Produktionssteuerung), Januar 1992
- Heft 89: G. Keller, M. Nüttgens, A.-W. Scheer: Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)", Januar 1992 (noch nicht veröffentlicht)
- Heft 90: C. Berkau, A.-W. Scheer: VOKAL (System zur Vorgangskettendarstellung), Teil 2: VKD-Modellierung mit Vokal, Dezember 1991
- Heft 91: C. Berkau: Konzept eines controllingbasierten Prozeßmanagers als intelligentes Multi-Agent-System, Januar 1992
- Heft 92: A. Hars, R. Heib, Chr. Kruse, J. Michely, A.-W. Scheer: Approach to classification for information engineering - methodology and tool specification, August 1992
- Heft 93: M. Nüttgens, A.-W. Scheer, M. Schwab: Integrierte Entsorgungssicherung als Bestandteil des betrieblichen Informationsmanagements, August 1992
- Heft 94: Chr. Kruse, A.-W. Scheer: Modellierung und Analyse dynamischen Systemverhaltens, Oktober 1992
- Heft 95: R. Backes, W. Hoffmann, A.-W. Scheer: Konzeption eines Ereignisklassifikationssystems in Prozeßketten, November 1992
- Heft 96: P. Loos: Die Semantik eines erweiterten Entity-Relationship-Modells und die Überführung in SQL-Datenbanken, November 1992
- Heft 97: Chr. Kruse, M. Gregor: Integrierte Simulationsmodellierung in der Fertigungssteuerung am Beispiel des CIM-TTZ Saarbrücken, Dezember 1992
- Heft 98: R. Heib: Konzeption für ein computergestütztes IS-Controlling, Dezember 1992
- Heft 99: H. Heß: Gestaltungsrichtlinien zur objektorientierten Modellierung, Dezember 1992
- Heft 100: P. Loos: Representation of Data Structures Using the Entity Relationship Model and the Transformation in Relational Databases, January 1993
- Heft 101: W. Hoffmann, J. Kirsch, A.-W. Scheer: Modellierung mit Ereignisgesteuerten Prozeßketten (Methodenbuch, Stand: Dezember 1992), Januar 1993
- Heft 102: P. Loos: Konzeption einer graphischen Rezeptverwaltung und deren Integration in eine CIP-Umgebung - Teil 1, Juni 1993

