

EPOS

Produktbeschreibung

Die leistungsfähige Entwicklungsumgebung für die Softwareentwicklung

Version 03.00



GPP AG

Kolpingring 18 a
D-82039 Oberhaching
Telefon: 089 - 613 04 - 1
Telefax: 089 613 04 - 207
e-mail: info@gppag.de
Internet: <http://www.gppag.de>

Ihr Partner für
Kommunikation und Information



EPOS

Copyright © 2002 GPP AG
Kolpingring 18 a
D-82039 Oberhaching bei München

+ 49 (0) 89 61 30 41

www.gppag.de

EPOS-TB-11003

Diese Technische Beschreibung ist für die EPOS Version 9.x.x gültig.

Alle Rechte vorbehalten. Kein Teil dieser Unterlage darf ohne vorherige schriftliche Genehmigung der GPP AG kopiert, gespeichert oder in anderer Weise abgeändert oder verbreitet werden.

Diese Unterlage kann ohne vorherige Ankündigung im Zuge der technischen Weiterentwicklung geändert werden. Sie stellt keine Zusicherung von Leistungsmerkmalen von EPOS dar.

Inhalt

1	Übersicht: Was ist EPOS.....	5
1.1	Zielsetzung	5
1.2	Aufbau und Komponenten	6
1.3	Erfahrungen beim praktischen Einsatz.....	7
2	Die methodische Grundkonzeption von EPOS.....	9
2.1	Durchgängige Rechnerunterstützung	9
2.2	EPOS und Vorgehensmodell	11
3	Leistungsumfang von EPOS	13
3.1	Das EPOS-Projektmodell	13
3.2	Unterstützung für unterschiedliche Entwurfsmethoden	16
3.3	Die Vorgehensweise bei der Anwendung von EPOS.....	18
3.4	Komponenten von EPOS	20
4	Die EPOS-Spezifikationsprachen	21
4.1	Vorbemerkungen.....	21
4.2	EPOS-R für die Anforderungsanalyse.....	21
4.2.1	Übersicht	21
4.2.2	Dokumentenaufbau	22
4.2.3	Anforderungen und Randbedingungen	22
4.2.4	Das Begriffslexikon	24
4.3	EPOS-S für den Entwurf.....	25
4.3.1	Übersicht	25
4.3.2	Grundsätzlicher Aufbau eines Entwurfsobjekts	26
4.3.3	Die Entwurfsobjekte.....	27
4.4	EPOS-P für das Projektmanagement	33
4.4.1	Übersicht	33
4.4.2	Beschreibungsmittel für Projektplanung und -steuerung.....	34
4.4.3	Beschreibungsmittel für das Änderungswesen	35
5	Die EPOS-Benutzeroberfläche.....	37
5.1	Übersicht.....	37
5.2	Das Dialogsystem.....	37

EPOS

5.3	Eingabe von EPOS-Spezifikationen.....	38
6	Die EPOS-Werkzeuge	39
6.1	Unterstützung von Entwurfsmethoden	39
6.2	Fehleranalysen	41
6.2.1	Fehlervermeidung und frühzeitige Fehlerentdeckung	41
6.2.2	Analyse von Anforderungen.....	42
6.2.3	Feststellung von Entwurfsfehlern	42
6.3	Metriken.....	43
6.4	Dokumentation	45
6.4.1	Variante 1 – Standarddokumentation (Anforderungen)	46
6.4.2	Variante 2 - Standarddokumentation (Entwurfsinformationen) ..	47
6.4.3	Der Dokumentengenerator.....	57
6.5	Codegenerierung	57
6.5.1	Möglichkeiten der Codegenerierung	57
6.5.2	Coderückführung.....	59
6.5.3	Einsatz von EPOS in Programmierumgebungen	59
6.6	Projektplanung	60
6.7	Projektkontrolle	61
7	Verfügbarkeit von EPOS	67

1 Übersicht: Was ist EPOS

1.1 Zielsetzung

EPOS (**E**ntwicklungs- und **P**rojektmanagement-**O**rientiertes **S**pezifikationssystem) ist ein Software-Werkzeugsystem zur Rechnerunterstützung für Entwicklungs- und Projektmanagementtätigkeiten. Es zielt darauf ab, den an einem IT¹-Projekt beteiligten Fachleuten – Technologen, Systemanalytikern, Projektleitern, Softwareentwicklern, Programmierern usw. – lästige Routinetätigkeiten abzunehmen und ihnen so mehr Freiräume für kreative Tätigkeiten zu eröffnen.

Im Einzelnen werden mit dem EPOS-System die folgenden Zielsetzungen verfolgt:

- Homogene Rechnerunterstützung für alle Phasen eines IT-Projekts
- Integrierte Rechnerunterstützung für Entwicklung, Projektmanagement und Produktverwaltung
- Rechnerunterstützung für unterschiedliche Entwurfsmethoden
- Rechnerunterstützung vor allem mit graphischen Darstellungen
- Rechnerunterstützung für Software- und Hardware-Entwicklung
- Unfassende statische und dynamische Prüfungen
- Automatische und schritthaltende Generierung der Dokumentation, Einhaltung von Entwicklungsstandards
- Automatische Codegenerierung mit Code-Rückführung (Tracing)
- Rechnerunterstützung für Pflege und Wartung

Ein Software-Werkzeugsystem, das in der Art des EPOS-Systems umfassende Rechnerunterstützung für die Software- und gegebenenfalls für die Hardwareentwicklung bietet, wird als CASE²-Toolsystem bezeichnet. Andere gebräuchliche Benennungen sind: Software-/Hardware-Entwicklungsumgebung oder Software-Produktionsumgebung (SPU, SDE³).

¹ Informationstechnik

² Computer Aided Software Engineering, Computer Aided System Engineering

³ Software Development Environment

Was ist EPOS?

1.2 Aufbau und Komponenten

EPOS umfasst im wesentlichen:

- Drei Spezifikationssprachen: für das Requirements-Engineering EPOS-R, für Entwürfe EPOS-S und für das Projektmanagement EPOS-P
- Funktionen zum Übertragen von EPOS-R, -S oder -P-Spezifikationen in die EPOS-Projektbibliothek
- Die EPOS-Projektbibliothek, d.h. eine Datenbank, in der alle projektrelevanten Informationen abgelegt werden und aus der alle Analysen und Dokumentationen erzeugt werden.
- Die EPOS-Werkzeuge, d.h. in EPOS integrierte Funktionen zur Auswertung der EPOS-Projektbibliothek und zur Ausgabe von Ergebnissen an die EPOS-Benutzer: Analysewerkzeuge zur Durchführung von statischen und dynamischen Prüfungen und zur Ausgabe von Fehlerhinweisen, Methodenwerkzeuge zur gezielten Führung und Hilfestellung für die vom Benutzer ausgewählte Entwurfsmethode, Dokumentationswerkzeuge für die Generierung von projektspezifischen Dokumenten und Codegenerierungswerkzeuge zur automatischen Erzeugung von Programm(uell)code für eine Reihe von Programmiersprachen. Zu den Codegenerierungswerkzeugen zählt auch die Code-Rückführung („Code-Feedback“). Das Projektmanagement-Unterstützungspaket umfasst alle Programme für die Projektplanung und Projektsteuerung.

1.3 Erfahrungen beim praktischen Einsatz

EPOS ist ein erprobtes Entwicklungssystem. Es wird seit 1980 in der Industrie, bei Behörden, in Forschungsinstituten und an Hochschulen eingesetzt. Seit seiner Markteinführung wurden über 10.000 Arbeitsplätze von EPOS verkauft.

Der Einsatzbereich von EPOS bei diesen Installationen reicht von der Entwicklung kleiner Assemblerprogramme für Produkte mit hoher Stückzahl bis hin zu großen technischen Systemen, die von verschiedenen Unternehmen mit unterschiedlichen Entwicklungsphilosophien und Entwicklungsplattformen in unterschiedlichen Programmiersprachen realisiert werden.

Das Einsatzspektrum reicht von komplexen sicherheitskritischen Realzeitanwendungen bis hin zu kommerziellen Programmen, von reinen Softwaresystemen bis hin zur Entwicklung von verteilten Software/Hardware-Systemen.

Beispiele von Anwendungssystemen, in denen EPOS in den letzten Jahren eingesetzt wurde, sind:

- Regelungssysteme und Navigationssysteme für Luftfahrzeuge
- Verschiedene technische Anwendungen
- Kraftfahrzeugsysteme
- Betriebsleitsysteme im Nahverkehr
- Werkzeugmaschinensteuerungen
- Vermittlungssysteme
- Entwicklung hardwarenaher Software
- Kommerzielle Anwendungssysteme für Bausparkassen
- Simulation und Rapid Prototyping bei Realzeitanwendungen
- Sanierung und Nutzungsdauerverlängerung von Legacy-Systemen

Der Nutzen von EPOS in Projekten ist ein wichtiges Ziel. Auf der Grundlage eines Fragebogens wurden Interviews mit Projektleitern bei einer Reihe von Unternehmen durchgeführt, um Erfahrungswerte für den tatsächlichen Nutzen von EPOS zu erhalten. Insgesamt wurden die Erfahrungen bei 22 Projekten in die Auswertung einbezogen.

Was ist EPOS?

Der Grundtenor der Erfahrungsberichte war: EPOS hat sich – trotz mancher bemängelter Schwächen – im praktischen Einsatz bewährt. Es trägt wesentlich zu einem systematischen, strukturierten Vorgehen bei und unterstützt wirkungsvoll die Entwicklung und die Projektleitung. Die Ergebnisse der Auswertung lassen sich wie folgt zusammenfassen:

- Durch den Einsatz von EPOS wurden die Kosten der Entwicklung gegenüber einem nicht rechnergestützten Vorgehen im Mittel um 9 % verringert.
- Bei der darauf folgenden Pflege und Wartung betrug die (mittlere) Kosteneinsparung jedoch 69 %. Je nach dem Anteil der Pflege- und Wartungskosten an den Gesamtkosten der Projekte ergab sich damit durch den Einsatz von EPOS ein beträchtlicher Kostenvorteil.
- Die Qualität der Entwicklungsergebnisse wurde in allen Phasen wesentlich verbessert. Dies wurde darauf zurückgeführt, dass EPOS ein methodisches Vorgehen und einen strukturierten Entwurf erzwingt.
- Es ergab sich eine Verringerung des Projekt-Risikos, d.h. einer größeren Sicherheit bezüglich der technischen Lösung und der termin- und kostengerechten Abwicklung der Projekte.
- Die Unterstützung von Entwicklungsstandards durch EPOS (z.B. des Vorgehensmodells, vgl. Abschnitt 2.2) wurde als wichtiger Aspekt begrüßt.

Von allen Benutzern wurde bestätigt, dass EPOS zwar im Großen und Ganzen leicht zu erlernen ist, dass aber eine gründliche Schulung Voraussetzung für den wirkungsvollen Einsatz ist (entsprechende Kurse werden von der GPP AG angeboten).

2 Die methodische Grundkonzeption von EPOS

2.1 Durchgängige Rechnerunterstützung

Bei der Durchführung eines technischen Projekts lassen sich, wie in Abbildung 1 gezeigt, drei Tätigkeitsbereiche unterscheiden:



Abbildung 1 – Ingenieurstätigkeiten in einem Entwicklungsprojekt

- **Entwicklungstätigkeiten**, beginnend mit dem „Requirements Engineering“ (Definition der fachlichen und technischen Anforderungen), Erarbeitung einer Systemarchitektur und anschließend der Entwurf von Software- und Hardwarekomponenten, sowie deren Realisierung. Die Realisierung kann zum Beispiel in Form eines Programms in einer bestimmten Programmiersprache erfolgen. Weitere Unterstützung bietet EPOS für die Entwicklung von Tests und für die Abnahme des Systems.
- **Projektmanagementtätigkeiten**, d.h. die Projektplanung mit Festlegung des zeitlichen Ablaufs und der dabei zu erarbeitenden Ergebnisse („Deliverables“), Zuweisung von Arbeitspaketen an die Projektmitarbeiter, Überwachung des Projektablaufs mit Termin- und Kostenkontrolle und Entscheidungen über zulässige Änderungen.
- **Produktmanagementtätigkeiten und Qualitätssicherung**. Dazu gehört die Verwaltung aller bei der Entwicklung anfallenden Zwischen- und Endergebnisse, wie zum Beispiel Pflichtenheft, verschiedene Versio-



Grundkonzeption von EPOS

nen und Varianten, Konfigurationen von Spezifikationen etc., sowie die Überwachung der Einhaltung qualitätsrelevanter Vorgaben.

Diese Tätigkeitsbereiche sind jedoch nicht unabhängig voneinander, sondern es bestehen Wechselwirkungen und Beziehungen. Die Grundkonzeption von EPOS beruht deshalb auf einer integrierten Rechnerunterstützung, bei der diese Beziehungen in Form von Relationen zwischen den Beschreibungsanteilen in einer integrierten Projektdatenbank („EPOS-Datenbank“) berücksichtigt werden.

2.2 EPOS und Vorgehensmodell

EPOS ist ein Softwarewerkzeug für die Anforderungsanalyse und den Entwurf auf den unterschiedlichen Ebenen und ist voll kompatibel mit dem Vorgehensmodell⁴.

Das Vorgehensmodell beschreibt in seinem Submodell SE die Aktivitäten SE 1 bis SE 9. Diese werden durch EPOS wie hier dargestellt unterstützt:

Aktivitäten des Vorgehensmodells	Anwendbare EPOS-Komponente
SE 1 – System-Anforderungsanalyse	EPOS-R (Anforderungsdefinition),
SE 2 – System-Entwurf	EPOS-S (Systemarchitektur mit ACTION MODULE-Objekten)
SE 3 – SW-/HW-Anforderungsanalyse	EPOS-R (Konzeption)
SE 4 – SW-Grobentwurf (Software)	EPOS-S
SE 5 – SW-Feinentwurf	EPOS-S
SE 6 – SW-Implementierung	EPOS Codegenerator(en)
SE 7 – SW-Integration	-
SE 8 – System-Integration	-
SE 9 – Überleitung in die Nutzung	-

Tabelle 1 - Vorgehensmodell und EPOS (SE-Aktivitäten)

Neben den Aktivitäten aus dem Submodell SE, in denen konstruktive Methoden zur Gewährleistung des geforderten Qualitätsniveaus von EPOS angeboten werden, bietet EPOS noch eine Reihe von Analysen und Auswertungen von Spezifikationen an, die die Qualitätssicherung bei der Wahrnehmung von analytischen QS-Maßnahmen unterstützen.

⁴ Entwicklungsstandard für IT-Systeme des Bundes (Vorgehensmodell), Juni 1997

Grundkonzeption von EPOS

Die nachstehende Tabelle beschreibt die Leistungsmerkmale, die EPOS für eine Reihe von V-Modell-Aktivitäten im Rahmen der Qualitätssicherung bietet:

Aktivitäten des Vorgehensmodells	Anwendbare EPOS-Komponente
QS 2 - Prüfungsvorbereitung	EPOS-R und EPOS-S zur Beschreibung von Prüfspezifikationen
QS 3 – Prozessprüfung von Aktivitäten	EPOS-A Auswertungen zur Einhaltung von methodischen Vorgaben
QS 4 - Produktprüfung	EPOS-A Auswertungen zum Nachweis der Vollständigkeit und Konsistenz von Spezifikationen (Prüfungen zu Entwicklungsergebnissen der Aktivitäten SE 1 bis SE 5)

Tabelle 2 - Vorgehensmodell und EPOS (QS-Aktivitäten)

Mit seinen Projektmanagementfunktionen unterstützt EPOS eine Reihe von Aktivitäten des Submodells PM des Vorgehensmodells:

Aktivitäten des Vorgehensmodells	Anwendbare EPOS-Komponente
PM 1.5 – Grobplan erstellen	EPOS-P Funktionen zur Erstellung von Balkenplänen, Meilensteinplänen sowie zur Ressourcenplanung
PM 4 – Feinplanung	EPOS-P Funktionen zur Erstellung von Balkenplänen, Meilensteinplänen sowie zur Ressourcenplanung
PM 8 – Projektkontrolle und -steuerung	EPOS-Analysen zur Ermittlung des Projektfortschritts sowie Projektmanagementfunktionen von EPOS für den Soll/Ist-Vergleich von geplanten Terminen und erreichten Terminen

Tabelle 3 - Vorgehensmodell und EPOS (PM-Aktivitäten)

3 Leistungsumfang von EPOS

3.1 Das EPOS-Projektmodell

Das EPOS-Projektmodell orientiert sich am Entwicklungsstandard für IT-Systeme des Bundes (Vorgehensmodell). Hierbei werden die folgenden grundsätzlichen Tätigkeiten im Rahmen des Systementwicklungsprozesses unterstützt:

- Analysetätigkeiten mit dem Ziel, Anforderungen aus fachlicher Sicht an das Gesamtsystem zu erstellen,
- Entwurfstätigkeiten mit dem Ziel, den grundsätzlichen Aufbau eines Systems in seine wesentlichen Elemente und deren Schnittstellen festzulegen,
- Analysetätigkeiten mit dem Ziel, technische Anforderungen für die Hauptelemente innerhalb einer Systemarchitektur zu formulieren,
- Entwurfstätigkeiten (Grob- und Feinentwurf) für die Hauptelemente der Systemarchitektur,
- Implementierung der Feinentwurfs im Code,
- Planung aller Aktivitäten des projektspezifischen Entwicklungsprozesses,
- Projektsteuerung und –kontrolle (insbesondere Unterstützung der Fortschrittsüberwachung),
- Maßnahmen der Qualitätssicherung im Rahmen von Analyse und Entwurfstätigkeiten.

In Abbildung 2 sind die Analysetätigkeiten gelb, die Entwurfstätigkeiten grün und die Unterstützung der Implementierung rot dargestellt. Die projektbegleitenden Aktivitäten zum Projektmanagement und zur Qualitätssicherung sind in der Abbildung blau hinterlegt.

Zu jeder Art von Tätigkeiten bietet EPOS speziell geeignete Darstellungsmittel an (Spezifikationssprachen):

- Für die Formulierung fachlicher und technischer Anforderungen wird **EPOS-R** (R für Requirements Engineering) verwendet (gelb).

Leistungsumfang von EPOS

- Für die Darstellung von Entwurfsinformationen auf den unterschiedlichen Ebenen wird **EPOS-S** (S für Systementwurf) verwendet (grün).
- Die Beschreibung von Projekten mit Aktivitäten, Ressourcen, Mitarbeitern sowie die Erfassung des Projektfortschritts erfolgt mit Mitteln von **EPOS-P** (P für Projektmanagement, in der Abbildung blau).

Zur jedem Aufgabenbereich bietet EPOS spezielle Dokumentations- und Analysefunktionen an („EPOS Werkzeuge“).

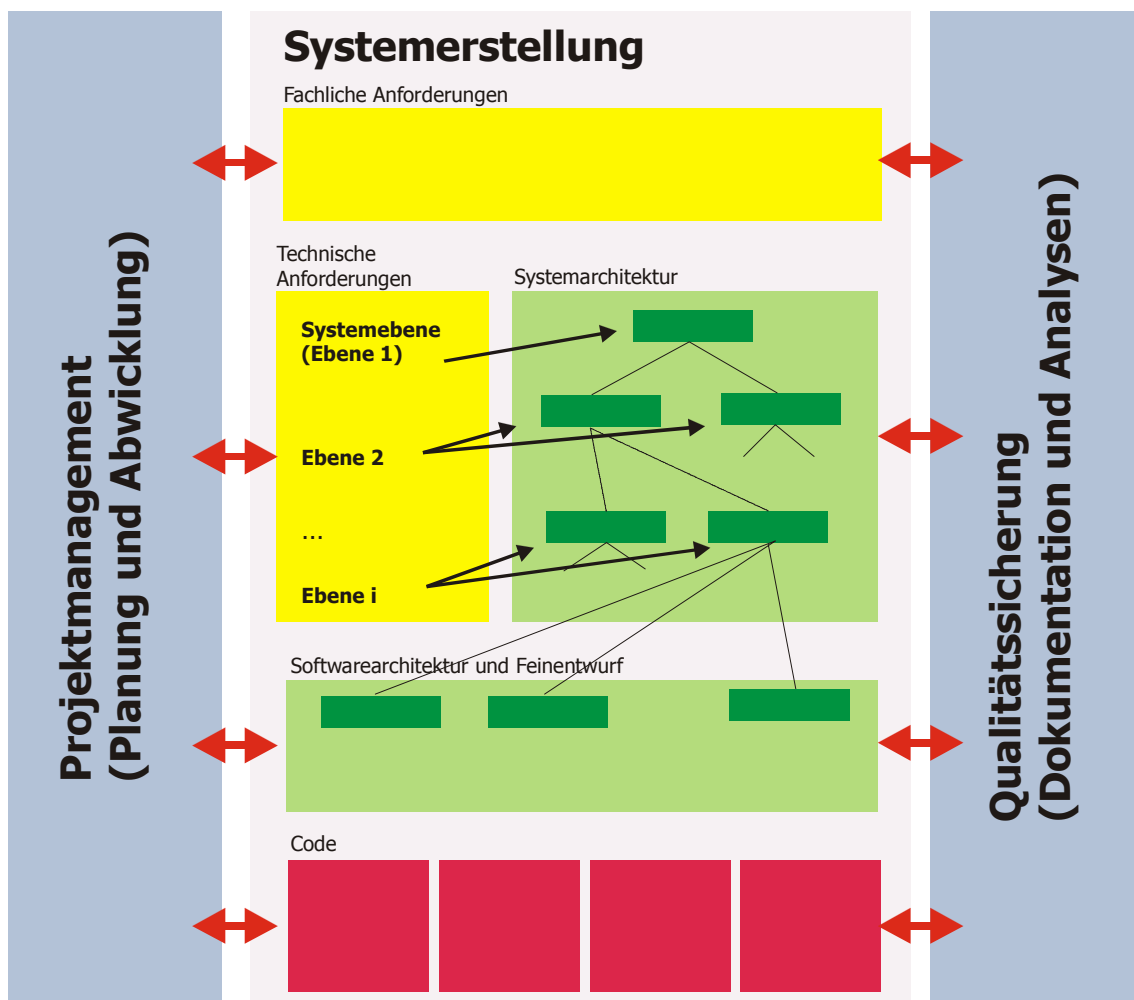


Abbildung 2 - Das EPOS-Projektmodell

Obwohl für die einzelnen Aufgaben unterschiedliche Darstellungsmittel verwendet werden, liegt der Schwerpunkt innerhalb von EPOS auf der durchgängigen Verknüpfung zwischen den Aufgabenbereichen:

Leistungsumfang von EPOS

- Die fachlichen Anforderungen bilden die Grundlage des Projekts.
- Bereits in der Projektplanung (Grobplanung) werden fachliche Anforderungen auf Arbeitspakete verteilt.
- Während der Durchführung von Analyse- und Entwurfstätigkeiten erfolgt eine Bezugnahme auf die jeweiligen Arbeitspakete, so dass zu jeder Zeit festgestellt werden kann, welche Entwicklungsergebnisse als Teil welchen Arbeitspakets entstanden sind.
- In der Systemarchitektur erfolgt eine Zuordnung von fachlichen Anforderungen zu den Hauptelementen des Systems als Grundlage für die Anforderungsverfolgung („Requirements Tracing“).
- Die Systemarchitektur bildet die Grundlage für die Strukturierung der technischen Anforderungen (zu jedem Element der Architektur wird ein Kapitel in den Technischen Anforderungen erstellt).
- Die Technischen Anforderungen stellen die Vorgabe für Grob- und Feinentwurf dar, die Nachweisführung der korrekten Umsetzung erfolgt wiederum durch eine Anforderungszuordnung.
- Die Implementierung wird durch Codegeneratoren für eine Reihe von Zielsprachen oder durch Codeselektion unterstützt. Der automatisch erzeugte Code ist so gekennzeichnet, dass Ergänzungen, die im weiteren Verlauf der Entwicklung durchgeführt werden, sofort den entsprechenden Teilen im Entwurf zugeordnet werden können („Code-Rückführung“ oder „Code Feedback“).
- Das Code-Feedback stellt die Konvergenz von Code und Entwürfen weitgehend automatisch sicher.

Leistungsumfang von EPOS

3.2 Unterstützung für unterschiedliche Entwurfsmethoden

Entsprechend den unterschiedlichen Gesichtspunkten, von denen man beim Entwurf von Software/Hardware-Systemen ausgehen kann, lassen sich beim Systementwurf verschiedenartige methodische Entwurfs-Ansätze (sog. Entwurfsmethoden) unterscheiden:

- Beim **Funktions-orientierten Entwurf** (auch als Aktions-orientierter Entwurf bezeichnet) wird die auszuführende Gesamtfunktion in die wichtigsten Teilfunktionen (Aktionen) aufgeteilt, diese Aktionen werden wieder verfeinert usw. So gelangt man zu einer baumartigen Struktur von Aktionen. Für jede Aktion werden dann – sozusagen im Nachzug – die jeweiligen Eingangs- und Ausgangsdaten festgelegt.
- Beim **Datenfluss-orientierten Entwurf** (analog zu der Anforderungsanalysemethode „Structured Analysis“) geht man schon beim „Requirements Engineering“ von den zu erfassenden, zu verknüpfenden und zu beeinflussenden Informationen (Daten) aus. Aus der Beschreibung der erforderlichen Datentransformationen (die hier im allg. als „Prozesse“ bezeichnet werden) ergeben sich dann – im Nachzug – die erforderlichen Aktionen und Abläufe. Die Datenflüsse und „Prozesse“ werden in EPOS mit Hilfe von EPOS-S beschrieben und in der für die jeweiligen Methoden typischen graphischen Notation dargestellt. Nach den dabei verwendeten Symbolen und Vorgehenschritten unterscheidet man die Methoden nach DeMarco, Gane-Sarson oder Yourdon.
- Beim **Datenstruktur-orientierten Entwurf** (bekannt als Michael-Jackson-Methode) werden als Erstes die Strukturen der Eingangs- und Ausgangsdaten des zu lösenden Problems analysiert und festgelegt. Daraus wird dann ein Programm abgeleitet, das aus der Eingangsdatenstruktur die Ausgangsdatenstruktur erzeugt.
- Beim **Objekt-orientierten**⁵ Entwurf (OOD von object-oriented design) werden – im Gegensatz zu den Funktions- und den Daten-orientierten

⁵ EPOS unterstützt speziell die HOOD-Methode. Die Definition des Begriffs „objekt-orientiert“ weicht hierbei von der Definition nach Booch, Rumbaugh oder Jacobson ab.

Ansätzen, bei denen ja Funktionen bzw. Daten zunächst getrennt betrachtet werden – Funktionen und Daten zu einer neuen Betrachtungseinheit, den sog. Objekten, zusammengefasst. Die Kommunikation zwischen den Objekten erfolgt über Nachrichten (messages). Eine Nachricht beschreibt, welche Operationen ein Objekt auszuführen hat. Eine spezielle Ausprägung des objekt-orientierten Entwurfsansatzes ist die sog. HOOD-Methode (von hierarchical object-oriented design), bei der Objekte hierarchisch in weitere Objekte zerlegt werden können.

- Beim **Ereignis-orientierten** Entwurf bilden die Ereignisse, auf die kurzfristig reagiert werden muss, den Ausgangspunkt der Überlegungen. Daraus werden dann die erforderlichen parallelen Abläufe im Rechnersystem und damit die Rechenprozesse oder Tasks festgelegt, die weiter in Aktionen verfeinert werden. Die dabei auszutauschenden Informationen (Daten) ergeben sich anschließend.
- Beim **Aufgaben-orientierten** Entwurf (auch als „Modul-orientierter Entwurf“ bezeichnet) werden die zu lösenden Aufgabenstellungen nach bestimmten Gesichtspunkten (z.B. nach dem dafür vorgesehenen Bearbeiter oder nach logischer Zusammengehörigkeit) zu Aufgabenbereichen zusammengefasst. Hierbei wird im allg. eine weitgehende Abgeschlossenheit der Aufgabenbereiche, eine eindeutige Definition der Schnittstellen und die Einhaltung des Prinzips des „Information hiding“ angestrebt. Anschließend werden dann die zwischen den Aufgabenbereichen auszutauschenden Informationen (Daten) und Funktionen (Prozeduren) definiert.
- Beim **Anlagen/Geräte-orientierten** Entwurf dominieren die zu verwendenden Anlagenteile und Gerätekomponten. Die Funktionen und Abläufe werden diesen Anlagen- und Geräteteilen zugeordnet. Daraus ergeben sich dann die über die Verbindungen der Geräte auszutauschenden Informationen (Daten).

Welche dieser methodischen Vorgehensweisen ist nun die beste? Die Ergebnisse von Untersuchungen zur Klärung dieser Frage zeigen, dass es keine „bes-

Leistungsumfang von EPOS

te" Vorgehensweise gibt, da jede Methode spezifische Vor- und Nachteile aufweist, die zudem von der jeweils vorliegenden Aufgabenstellung abhängen.

Daher ist EPOS nicht auf eine bestimmte Entwurfsmethode ausgerichtet. Es bleibt der Entscheidung des Anwenders (oder der Projektleitung) überlassen, wie er – in Abhängigkeit von der vorliegenden Aufgabenstellung – vorgehen will. EPOS stellt ihm dabei für die oben aufgeführten Entwurfsmethoden unterstützende Sprachmittel und Softwarewerkzeuge zur Verfügung.

3.3 Die Vorgehensweise bei der Anwendung von EPOS

Die Vorgehensweise bei der rechnerunterstützten Durchführung eines Projekts mit EPOS zeigt Abbildung 3. Die an einem Projekt Beteiligten formulieren die erarbeiteten Informationen (z.B. fachliche und technische Anforderungen, Projektstruktur, Arbeitspakete, Systemarchitektur, Softwareentwurf mit Daten- und Kontrollfluss) mit Hilfe der Spezifikationssprachen EPOS-R, EPOS-S und EPOS-P. Bei der Eingabe in den Rechner verwenden die Benutzer den in das EPOS-System integrierten Editor des jeweiligen Rechners.

Leistungsumfang von EPOS

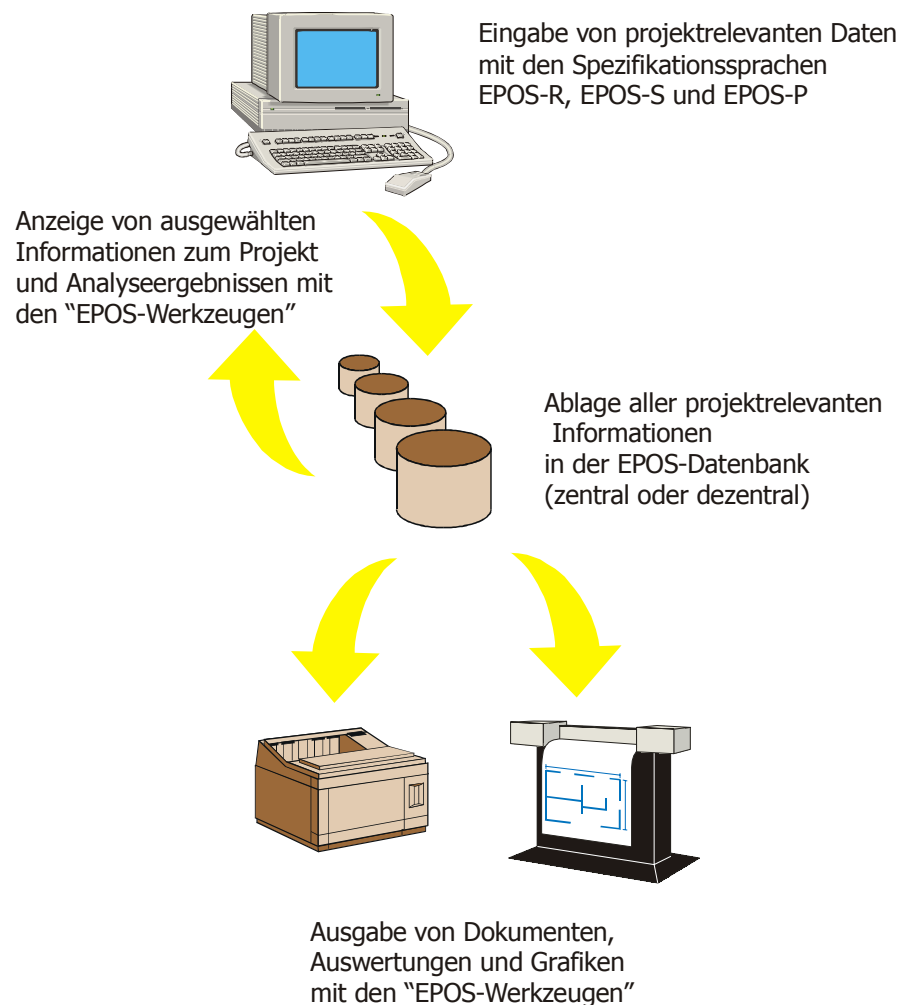


Abbildung 3 - Arbeiten mit EPOS

Im Rechner wird aus diesen Informationen eine Datenbank („EPOS-Datenbank“) aufgebaut. Diese Datenbank kann dann von den Dokumentations- und Auswertungsfunktionen von EPOS („EPOS-Werkzeuge“) verwendet werden. Die Ergebnisse, die diese Funktionen liefern, können am Arbeitsplatz, auf Druckern, oder, bei großformatigen Grafiken, auch auf Plottern ausgegeben werden.

Beim Einsatz von EPOS auf Mehrplatzsystemen (z.B. auf Anlagen DEC Alpha AXP) bietet EPOS die Möglichkeit dezentraler Datenbanken und erlaubt so die Aufteilung des Datenbestandes unter die Mitarbeiter eines Projektes, wobei Dokumentation und Auswertungsfunktionen stets auf der Grundlage des gesamten Datenbestandes durchgeführt werden können.

Leistungsumfang von EPOS

3.4 Komponenten von EPOS

Aus der Sicht des Anwenders besteht EPOS aus den folgenden Hauptkomponenten:

- **EPOS-R** umfasst die Beschreibungsmittel, die Dokumentations- und Auswertefunktionen für die Bearbeitung von fachlichen und technischen Anforderungen,
- **EPOS-S** umfasst die Beschreibungsmittel, die Dokumentations- und Auswertefunktionen für die Bearbeitung von Entwurfsinformationen in den unterschiedlichen Projektphasen sowie eine Reihe von Codegeneratoren für unterschiedliche Zielsprachen
- **EPOS-P** umfasst die Beschreibungsmittel, die Dokumentations- und Auswertefunktionen für die Bearbeitung von Projektmanagementinformationen sowie weitergehende Auswertefunktionen, in denen technische Informationen aus Sicht des Projektmanagements analysiert werden (z.B. Projektfortschrittskontrolle)
- Mit dem **Dokumentengenerator** lassen sich in Dokumenten mit einer nutzerdefinierten Struktur Informationen aus allen Bereichen von EPOS zusammenführen.
- Für den Anwender weitgehend verborgen, stellt die **Datenbank** als das zentrale Ablagesystem aller projektrelevanten Informationen eine weitere Hauptkomponente von EPOS dar.

Die Namensgeber der Hauptkomponenten sind dabei die Spezifikationssprachen EPOS-R, EPOS-S und EPOS-P, die jeweils spezielle Darstellungsmittel für die unterschiedlichen Aufgaben zur Verfügung stellen: Betrachtet man also etwa „EPOS-S“ so versteht man hierunter sowohl die Spezifikationssprache zur Beschreibung von Entwurfsinformationen als auch den gesamten Leistungsumfang von EPOS zur Unterstützung von Entwurfsaufgaben (Systemarchitektur entwerfen, Softwaregrob- und Feinentwurf) bis hin zur Implementierung (Codegenerierung).

4 Die EPOS-Spezifikations Sprachen

4.1 Vorbemerkungen

Die EPOS-Spezifikations Sprachen dienen zur Formulierung von Informationen, die in der EPOS-Datenbank abgelegt werden sollen. Bei der Eingabe erfolgt eine Überprüfung, ob die Syntax-Regeln der Spezifikations Sprachen eingehalten sind.

4.2 EPOS-R für die Anforderungsanalyse

4.2.1 Übersicht

Anforderungsanalysetätigkeiten sind zur Festlegung von fachlichen und technischen Anforderungen erforderlich. Die Ergebnisse werden in Dokumenten niedergelegt (nach dem Vorgehensmodell zum Beispiel: fachliche Anforderungen in den „Anwenderforderungen“, technische in den „Technischen Anforderungen“).

Anforderungsdokumente beschreiben die geforderten Sachverhalte mit Hilfe von freien Texten, Tabellen und Grafiken.

Innerhalb der Anforderungsdokumente sollen Einzelanforderungen als solche erkennbar und ansprechbar (zum Zwecke der Referenzierung) sein.

Wesentlich ist dabei auch die Möglichkeit, Begriffe mit einer festgelegten Bedeutung als solche kenntlich zu machen und, wo erforderlich, stets eine einheitliche Bedeutung abzudrucken.

EPOS unterstützt diesen Ansatz in seinen beiden EPOS-R-Bereichen „Aufgabenstellung“ (zur Aufnahme der fachlichen Anforderungen) sowie „Konzeption“ (für die technischen Anforderungen) sowie einem beiden Bereichen gemeinsamen Begriffslexikon.

Der Aufbau der beiden Dokumente, die mit Hilfe von EPOS-R erstellt werden können, richtet sich nach dem jeweils verwendeten Standard: So ist zum Beispiel eine Festlegung des Dokumentenaufbaus nach dem Vorgehensmodell oder nach anderen Prozessmodellen möglich.

Die EPOS Spezifikationssprachen

4.2.2 Dokumentenaufbau

Das Grundgerüst jeder EPOS-R Beschreibung bildet das dekadische Gliederungsschema. Es kann für beide Bereiche von EPOS-R jeweils ein Schema vom Anwender definiert werden. Dabei können Gliederungsschemata übernommen werden, wie sie von Normungsgremien vorgeschlagen werden, oder solche Schemata, die nach firmeninternen Standards bzw. projektabhängig definiert werden.

Die Gliederungsschemata bestehen formal aus einzelnen Kapiteln, die in Sektionen untergliedert sind.

4.2.3 Anforderungen und Randbedingungen

In einer umfassenden Anforderungsdefinition, sei es nun in den fachlichen oder den technischen Anforderungen wird grundsätzlich zu unterscheiden sein zwischen

- Verbindlichen Anforderungen, die im Rahmen der Entwicklung von der Entwicklungsorganisation aktiv zu realisieren sind und im Zuge der Verifikation und Validierung (z.B. bei Tests) nachzuweisen sind und
- Zusätzlichen Erläuterungen, Hinweisen oder Beispielen, die dem Verständnis des Gesamtdokuments dienen

Um diese Unterscheidung in den Anforderungsdokumenten klar zu machen, werden in EPOS-R Anforderungen („Requirements“) und Randbedingungen („Constraints“) als formale Einschübe gekennzeichnet. So gekennzeichnete Anforderungen und Randbedingungen bilden die Grundlage für die Anforderungsverfolgung („Requirements Tracing“).

Anforderungen lassen sich als hierarchische Strukturen festlegen, d.h. beginnend auf einem hohem Beschreibungsniveau werden Anforderungen schrittweise konkretisiert („verfeinert“). Eine sehr allgemeine Anforderung wird dann durch ihre jeweiligen Konkretisierungen ersetzt („substituiert“). Allgemeine Anforderungen gelten dann im weiteren Entwicklungsgang als erfüllt, wenn alle ihre Konkretisierungen erfüllt sind. Abbildung 4 zeigt einen Auszug aus einem Anforderungsdokument, in dem drei Anforderung identifiziert sind (66(1), 66(2) und 66(3)). Dabei wird die allgemeine Anforderung 66(1) durch ihre Konkreti-

Die EPOS Spezifikationsprachen

sierungen (66(2) und 66(3) ersetzt. Das Beispiel zeigt weiterhin die Möglichkeit, Anforderungen besondere Merkmale („Attribute“) zuzuordnen, die als Suchkriterium verwendet werden können.

6.6. Anforderungen an die Eingabefunktionen

Für den Dialogbetrieb werden die folgenden Anforderungen gestellt:

REQUIREMENT 66(1):

< GUI, Fehlertoleranz >

Alle Eingaben sind auf Konsistenz zu prüfen.

REQUIREMENT 66(2):

Datumseingaben sind auf Plausibilität zu prüfen.

REQUIREMENT 66(3):

Bei der Eingabe der Stückzahl dürfen nur positive Werte angenommen werden.

S U B S T I T U T E : REQUIREMENT 66(1) BY REQUIREMENT 66(2),
REQUIREMENT 66(3).

Abbildung 4 - Auszug aus einem Anforderungsdokument

Die EPOS Spezifikationssprachen

4.2.4 Das Begriffslexikon

Zur Unterstützung der Kommunikation zwischen den an einem Projekt beteiligten Personen, sowie zur Verbesserung der Lesbarkeit der Dokumentation dient ein Lexikon, das in der EPOS-Datenbank geführt wird. Es ermöglicht eine einheitliche Definition und Verwendung der Begriffe und trägt so wesentlich zur Vermeidung von Interpretationsfehlern bei.

Der Anwender kann Begriffe

- definieren, d.h. den Definitionstext ins Lexikon eintragen lassen
- markieren, d.h. das Vorkommen von Begriffen in den EPOS-R Dokumenten suchen und markieren und einen dazugehörigen Index erstellen lassen
- referenzieren, d.h. den Definitionstext im Rahmen eines EPOS-R-Dokuments ausgeben lassen
- realisieren, d.h. ein EPOS-S Entwurfsobjekt mit gleichem Namen definieren und damit einen Bezug von EPOS-R zum Systementwurf und dem daraus abgeleiteten Programm herstellen
- vervollständigen, d.h. den Definitionstext ändern bzw. erweitern

Verwandten Begriffen können Begriffskategorien zugeordnet werden.

Abbildung 5 zeigt die Definition des Begriffs „Brennerstörungskontakt“ mit den Begriffskategorien "Sensor" und "Sicherheit".

```
BEGRIFF Brennerstoerungskontakt  
  
< Sensor , Sicherheit >  
  
„Potentialfreier Kontakt (Schliesser) fuer  
Spannungen bis 60 V Gleichstrom oder 220 V  
Wechselstrom zur Meldung von Brennerstoerungen.“
```

Abbildung 5 - Definition eines Begriffs mit EPOS-R

Die EPOS Spezifikationsprachen

4.3 EPOS-S für den Entwurf

4.3.1 Übersicht

Zur formalen Beschreibung des Systementwurfs (d.h. von der Systemarchitektur bis hin zum Feinentwurf) werden in EPOS-S sogenannte Entwurfsobjekte verwendet. Sie werden durch Schlüsselwörter (wahlweise deutsch oder englisch) oder – bei der grafischen Darstellung – durch Symbole gekennzeichnet.

EPOS verwendet die folgenden Entwurfsobjekttypen:

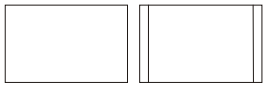

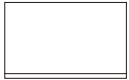




Entwurfsobjekttyp	Englisches Schlüsselwort	Deutsches Schlüsselwort	Grafische Darstellung (Symbol)
Aktion	ACTION	AKTION	
Modul	ACTION MODULE	AKTION MODUL	
Date (Informationseinheit)	DATA	DATE	
Schnittstelle	INTERFACE	SCHNITTSTELLE	
Ereignis	EVENT	EREIGNIS	
Bedingung	CONDITION	BEDINGUNG	
Ausführungseinheit	DEVICE	AUSFUEHRUNGSEINHEIT	

Tabelle 4 - EPOS-Entwurfsobjekttypen

Entwurfsobjekte stehen durch eine Reihe von Verknüpfungen untereinander in Verbindung, zum Beispiel durch einen Datenfluss (eine Informationseinheit wird von einer Aktion zu einer anderen übertragen), einen Kontrollfluss (zwei Aktionen stehen in einer logischen Abfolge) oder über ein Ereignis (ein Ereignis löst eine Aktion aus).

Abbildung 6 zeigt die möglichen Verknüpfungen von Entwurfsobjekten in der Übersicht: Ein Ereignis (E1) führt zur Auslösung einer Aktion (A1.1). Diese Akti-

Die EPOS Spezifikationsprachen

on bezieht eine Information (Date D1) von einer Schnittstelle (I1) und stellt eine andere Information (Date D2) an einer weiteren Schnittstelle (I2) bereit. Die Ausführungseinheiten DV1 und DV2 zeigen an, wie logische Funktionalitäten (Aktion, Schnittstelle) gerätetechnisch realisiert werden.

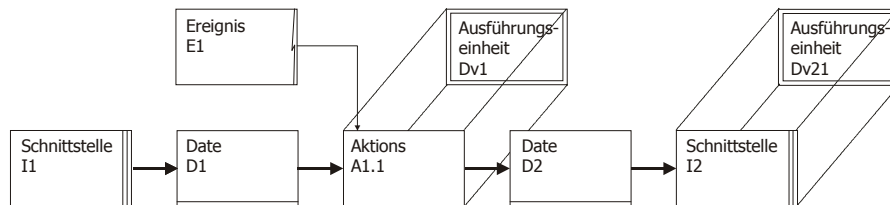


Abbildung 6 - Verknüpfung von Entwurfsobjekten

4.3.2 Grundsätzlicher Aufbau eines Entwurfsobjekts

Entwurfsobjekte sind die Darstellungseinheiten in der EPOS-Datenbank. Dies bedeutet, dass in der EPOS-Datenbank keine Grafiken oder Dokumente abgelegt werden, sondern die Entwurfsobjekte mit ihren Verknüpfungen. Aus diesen Informationen werden dann später Dokumente erzeugt bzw. es werden Auswertungen unter Verwendung der zu den Entwurfsobjekten abgelegten Informationen durchgeführt.

Alle Entwurfsobjekte besitzen denselben Grundaufbau: Ein Entwurfsobjekt wird durch sein jeweiliges Schlüsselwort identifiziert (z.B. ACTION, DATA, EVENT) und besitzt einen eindeutigen Namen, der vom Anwender frei wählbar ist.

Zu einem Entwurfsobjekt können Beschreibungen und Informationen zu den Verknüpfungen mit anderen Entwurfsobjekten hinterlegt werden.

Verknüpfungen zu anderen Entwurfsobjekten sind dabei zum einen die Beziehungen, die für das Entwurfsobjekt innerhalb der Beschreibungsebene festgelegt werden (z.B. kann eine AKTION eine DATE verwenden), zum anderen werden die Beziehungen zu anderen Beschreibungsebenen angegeben: Eine AKTION (mit der eine Funktionalität auf hohem Niveau beschrieben wurde) kann aus weiteren AKTIONen bestehen, die dann, zusammengenommen, dieselbe Funktionalität auf einem Niveau höherer Detaillierung beschreibt.

Die EPOS Spezifikationsprachen

Beispiel:

Das System „Flugsicherung“ besteht zunächst aus einer einzigen AKTION. Im Verlauf der Systemstrukturierung wird festgelegt, dass die AKTION Flugsicherung aus einer Reihe von (Teil-) AKTIONen besteht (z.B. „Anflugkontrolle“, „Landekontrolle“ usw.). Alle Teilaktionen zusammengenommen bilden die (Gesamt-) Funktionalität „Flugsicherung“.

Durch diese Vorgehensweise entsteht ein Ebenenmodell des zu entwickelnden Systems, das zugrundeliegende Prinzip nennt man „schrittweise Verfeinerung“. Die höheren (früh erstellten) Ebenen geben einen Systemüberblick mit knappen Informationen, die tieferen (später erstellten) fügen Details hinzu.

Das Schlüsselwort des Entwurfsobjekts mit einem angehängten „-END“ (z.B. ACTIONEND, DATAEND, EVENTEND) schließt die Informationen zu einem Entwurfsobjekt ab.

4.3.3 Die Entwurfsobjekte

Das Entwurfsobjekt vom Typ **ACTION** dient zur Beschreibung von Vorgängen ganz allgemein. Vorgänge verknüpfen in einem bestimmten zeitlichen Ablauf Daten, formen sie um oder verarbeiten sie. Aktionen werden durch Ereignisse ausgelöst und können ihrerseits andere Ereignisse auslösen. Aktionen, die Vorgänge sehr allgemein beschreiben, können in (Teil-) Aktionen gegliedert werden. Alle Teilaktionen zusammengenommen führen denselben Vorgang durch.

Abbildung 7 zeigt das Beispiel eines Entwurfsobjekts mit einem Beschreibungsteil, einem Verfeinerungsteil und mit Verknüpfungen zu anderen Entwurfsobjekten derselben Beschreibungsebene.

Die EPOS Spezifikationsprachen

ACTION Heizungsregelung .

DESCRIPTION :

PURPOSE : "

Regelung und Überwachung einer Heizungsanlage unter Verwendung eines Mikrorechners. Zu den Automatisierungsaufgaben gehören die Regelung der Temperatur in zwei unabhängigen Raumzonen durch Verstellen der Vorlauftemperatur sowie die Kesseltemperatur-Regelung und die Brennerüberwachung.

Nach dem Einschalten (externes Ereignis 'SYSTEMSTART') sollen zunächst die Sollwerte eingegeben werden (INITIALISIERUNG). Dann erst sollen die drei Regelkreise freigegeben werden, wobei drei verschiedene Abtastzyklen (ZYKLUS-1, ZYKLUS-2 und ZYKLUS-KESSEL) verwendet werden." .

NOTE : "

In einer späteren Ausbaustufe sollen die Sollwerte tageszeitabhängig vorgegeben und gespeichert werden." .

TEST : "

Testparameter siehe Unterlage 25/100" .

DATE : 25.07.02 .

FULFILS : REQUIREMENT 4 (0) .

DESCRIPTIONEND

DECOMPOSITION :

Initialisierung ;
SET (Zyklus-1 , Zyklus-2 , Zyklus-Kessel , Brennerstoerung ,
Anforderung)
(/ Regelung-1 , Regelung-2 , Regelung-Kessel ,
Sollwerte-Eingabe , Brenner-Ueberwachung /) .

TRIGGERED : SYSTEMSTART .

INPUT : Regelgroessen ,
Sollwerte FROM Eingabegeraet .

OUTPUT : Stellgroessen TO Ventile .

REALIZATION : SOFTWARE IN 'C' .

PROCESSED : Mikrorechner .

ACTIONEND

Abbildung 7 - Entwurfsobjekt AKTION (Beispiel)

Das Entwurfsobjekt vom Typ Modul (**ACTION MODULE**) dient zur Beschreibung von Teilsystemen, bei denen der wirkungsmäßige Zusammenhang und die

Die EPOS Spezifikationsprachen

Einbindung in einen Kontrollfluss (wie bei den Aktionen) zunächst nicht interessiert oder zum gegenwärtigen Zeitpunkt noch nicht festgelegt werden kann, sondern wo statt dessen eine logische Abgrenzung erfolgen soll.

Ein Modul kann als Sammelbegriff für - nach einem bestimmten logischen Gesichtspunkt zusammengefasste - Entwurfsobjekte aufgefasst werden. Dies bedeutet, dass die in einem bestimmten Modul enthaltenen Entwurfsobjekte wirkungsmäßig zu einem oder mehreren anderen Modulen gehören können. Die wirkungsmäßige Zuordnung wird dadurch beschrieben, dass Aktionen und Daten eines Moduls einerseits "exportiert", d.h. anderen Modulen zur wirkungsmäßigen Einbindung zur Verfügung gestellt werden können, und andererseits "importiert", d.h. in einem Modul verwendet werden können. Importierte Aktionen und Daten müssen notwendigerweise in einem anderen Modul, in dem sie zusammengefasst sind, exportiert, d.h. zur Verwendung freigegeben sein.

Das Entwurfsobjekt vom Typ **DATA** dient zur Beschreibung aller während des Entwurfsvorgangs auftretenden Informationen wie z. B. Stellgrößen, Messwerte, Uhrzeiten, usw. Neben den in EPOS-S bereits vordefinierten Datentypen hat der Anwender die Möglichkeit, eigene Datentypen festzulegen.

Die EPOS Spezifikationsprachen

Datentyp	Erläuterung
FIXED	Ganze Zahl mit oder ohne Angabe einer Einheit
FLOAT	Gleitkommazahl mit oder ohne Angabe einer Einheit
DUR	Zeitdauer
CLOCK	Uhrzeit
CHAR	Zeichenkette, Text
BIT	Bitkette
BOOLEAN	Wahrheitswert
POINTER	Verweis auf eine andere Informations-einheit

Tabelle 5 - Vordefinierte Datentypen in EPOS

Den Daten können Wertebereiche, Anfangswerte und konstante Werte zugewiesen werden; als Datenstrukturen sind z.B. Felder, Warteschlangen, Kellerspeicher und logische Datensätze zulässig.

Das Entwurfsobjekt vom Typ **INTERFACE** dient zur Beschreibung des Datenaustausches zwischen dem zu entwerfenden technischen System und seinem "übergeordneten System" (z. B. dem technischen Prozess, dem Bedienpersonal usw.), oder zur Beschreibung interner Schnittstellen.

Die EPOS Spezifikationsprachen

Zur Beschreibung von Ereignissen, die im System gemeldet werden und die den Ablauf von Aktionen beeinflussen, dient in EPOS-S das Entwurfsobjekt vom Typ **EVENT**. Dabei lassen sich verschiedene Kategorien unterscheiden:

- Interrupts,
- zyklische Ereignisse,
- Ereignisse zu bestimmten, absoluten Zeitpunkten.
- Ereignisse, die zur Synchronisierung eines Rendezvous verwendet werden.

Das Entwurfsobjekt vom Typ **CONDITION** dient zur Beschreibung datenabhängiger Bedingungen, die den Kontrollfluss von Aktionen steuern. Bedingungen können durch die Angabe eines logischen Ausdrucks verfeinert werden, der Entscheidungsrelationen (größer, größer gleich, etc.) und/oder Unterbedingungen enthalten kann.

Entwurfsobjekte vom Typ **DEVICE** dienen, wie der Name ausdrückt, zur Beschreibung von Einheiten, die Aktionen oder Schnittstellen ausführen, also z.B.

- Mensch-Maschine-Systeme oder organisatorische Einheiten,
- Geräte,
- Rechner-Hardwareeinheiten.

Abbildung 8 zeigt ein Beispiel für eine Gerätebeschreibung.

Die EPOS Spezifikationsprachen

```
DEVICE Mikrorechner-Steuergeraet .

DESCRIPTION :

PURPOSE : "
    Der Mikrorechner übernimmt die Überwachung der Prozesszustände
    und die Steuerung der Stellmotoren." .

DESCRIPTIONEND

COMPONENTS : CPU TYPE MC6809 ,
              PIA TYPE MC6821 ,
              ACIA ,
              Speicher ,
              Pegelumsetzer ,
              Verstaerker .

INOUT : _Motoranschluss ,
        _Sensoranschluss ,
        _V24-Anschluss ,
        _Spannungsversorgung .

CONNECTION :

CHANNEL SYSTEMBUS : BUS <==> CPU <==> Speicher <==> ACIA <==> PIA
                  .
                  PIA _PORTA <==> Verstaerker _EA ( 0 : 7 ) .
                  PIA _PORTB <==> Verstaerker _EA ( 8 : 15 ) .

SIGNAL INTERRUPT-1 : PIA _IRQB --> CPU _FIRQ .

SIGNAL INTERRUPT-2 : ACIA _IRQ --> CPU _IRQ .
                  ACIA _EA <==> Pegelumsetzer _V24TTL .
                  Verstaerker _AUS ( 0 : 7 )
                  ==> _Motoranschluss .
                  Verstaerker _EIN ( 8 : 15 )
                  <== _Sensoranschluss .
                  Pegelumsetzer _V24 <==> _V24-Anschluss .

FEATURES : "
    Die Steuerung ist so ausgelegt, dass die dynamischen Kenndaten
    der Stellmotoren berücksichtigt sind:..." .

DEVICEEND
```

Abbildung 8 - Entwurfsobjekt DEVICE (Beispiel)

Die EPOS Spezifikationsprachen

4.4 EPOS-P für das Projektmanagement

4.4.1 Übersicht

Die Spezifikationsprache EPOS-P dient zur Beschreibung der Sachverhalte und Informationsflüsse, die für das Projektmanagement, die Produktverwaltung und die Qualitätssicherung erforderlich sind. Um die Abbildung auf die EPOS-Datenbank zu ermöglichen und damit die Voraussetzungen für die Rechnerunterstützung durch die Toolsysteme zu schaffen, müssen diese Sachverhalte und Informationsflüsse mit Hilfe formaler Sprachmittel beschrieben werden. Dabei werden in EPOS-P Bezüge zu den fachlichen und technischen Beschreibungen (EPOS-R) und Entwürfen (EPOS-S) hergestellt, wodurch innerhalb von EPOS eine direkte Integration von Management und technischer Entwicklung erreicht wird.

Als Sprachmittel der Spezifikationsprache EPOS-P werden - in Entsprechung zu den für die Spezifikationsprache EPOS-S erläuterten Entwurfsobjekten - sogenannte Managementobjekte in EPOS verwendet.

Management-Objekttyp		Sachverhalte und Informationen, die beschrieben werden
Englisches Schlüsselwort	Deutsches Schlüsselwort	
ACTIVITY	AKTIVITAET	Element der Projektstruktur (Ablaufplanung, Netzplanung)
TEAMMEMBER	PROJEKTBETEILIGTER	Projektorganisation. Beschreibung von Funktionen und Kompetenzen
RESOURCE-UNIT	EINSATZMITTELART	Einsatzmittel und Kosten
PRODUCT	PRODUKT	Liefergegenstand
PROGRESS-REPORT	FORTSCHRITTS-BERICHT	Berichte über Fortschritte, Schwierigkeiten, Statusänderungen
CHANGE-PROPOSAL	AENDERUNGSANTRAG	Änderungsanträge und ihre Bewertung
ERROR-REPORT	FEHLERBERICHT	Meldungen über aufgetretene Fehler

Tabelle 6 - Managementobjekttypen in EPOS

Die EPOS Spezifikationsprachen

4.4.2 Beschreibungsmittel für Projektplanung und -steuerung

Zur Beschreibung von Projektmanagement-bezogenen Sachverhalten und Informationsflüssen dienen in EPOS-P die Managementobjekte der Typen Aktivität (ACTIVITY), Projektbeteiligter (TEAMMEMBER), Produkt (PRODUCT), Einsatzmittelart (RESOURCE-UNIT) und Fortschrittsbericht (PROGRESS-REPORT).

Das Managementobjekt vom Typ **ACTIVITY** dient zur Beschreibung von Teilprojekten, Arbeitspaketen, und Netzplankomponenten.

Das Managementobjekt vom Typ **TEAMMEMBER** dient zur Beschreibung der Projektorganisation, sowie der Funktionen, Zuständigkeiten und Kompetenzen der in einem Projekt beteiligten Personen, Abteilungen oder Firmen.

Mit dem Managementobjekttyp **PRODUCT** können die Ergebnisse der Entwicklung (Leistungsumfang, „Deliverables“) beschrieben werden und in Verbindung mit diesen festgelegt werden, zu welchen Zeitpunkten bestimmte Produkte (Entwicklungsergebnisse) vorliegen müssen

Mit dem Managementobjekt vom Typ **RESOURCE-UNIT** wird die Kosten- und Einsatzmittelplanung beschrieben. Das Ergebnis dieser Planung sind Datenbankeinträge mit der Zuordnung einer Aktivität zu allen projektrelevanten Strukturen:

- Anforderungsstruktur
- Projektstruktur
- Ablaufstruktur (Netz /Meilensteinplan)
- Kostenträgerstruktur
- Projektorganisation
- Qualifikationsstruktur
- Kostenstellenstruktur

Das Managementobjekt vom Typ **PROGRESS-REPORT** ermöglicht die Erfassung von "dynamischen" Daten während des Projektablaufs, wie z.B. angefallene Kosten, Fertigstellungszustände von Aktivitäten, soweit sie nicht automatisch erfasst werden können.

4.4.3 Beschreibungsmittel für das Änderungswesen

Zwei wesentliche Bereiche der Rechnerunterstützung für die Produktverwaltung und für die Qualitätssicherung sind

- Das Änderungswesen
- Das Störmeldeverfahren

Als Sprachelemente hierfür sind in EPOS-P zwei Management-Objekte enthalten: Das Management-Objekt vom Typ **CHANGE-PROPOSAL** erlaubt die Formulierung einer gewünschten Änderung, sowie deren anschließende Bewertung.

Mit dem Management-Objekt vom Typ **ERROR-REPORT** werden alle Meldungen über aufgetretene Fehler in der EPOS-Datenbank gesammelt.



Die EPOS Spezifikationsprachen

5 Die EPOS-Benutzeroberfläche

5.1 Übersicht

Bei der Arbeit am EPOS-Arbeitsplatz wird der Anwender durch das Bediensystem EPOS-C geführt.

Dieses Bediensystem ist vom Betriebssystem der jeweiligen Einsatzplattform unabhängig und bietet damit auf verschiedenen Zielrechnersystemen die gleiche Bedienoberfläche.

Das Bediensystem erfüllt sehr vielfältige Aufgaben:

- interaktive Dialogführung, um die verschiedenen EPOS-Funktionen richtig parametrisiert aufzurufen.
- Ansteuerung der Ein- und Ausgabemedien zur Erfassung der in EPOS-R, EPOS-S oder EPOS-P formulierten Beschreibungen von einer Datei oder interaktiv am Bildschirm über Text- oder Graphikeditoren, sowie Ausgabe der Dokumente, Prüfergebnisse usw. am Bildschirm, auf einem Drucker oder Plotter.
- Batchfähigkeit, um verschiedenste Arbeitsschritte auf Knopfdruck beliebig oft reproduzierbar auszuführen.
- dezentrale Arbeitsweise, die es erlaubt, die lokalen Entwicklungen einzelner Projektbeteiligter in einer EPOS-Sitzung zusammenzuschalten.

5.2 Das Dialogsystem

Mit Hilfe des Dialogsystems wählt der Benutzer die gewünschte EPOS-Funktion aus und bestimmt ihre Parameter. Er hat die Wahl zwischen zwei verschiedenen Dialogformen:

- "Kommando-Modus": In einem zeilenorientierten Dialog wählt der Benutzer Schritt für Schritt eine Funktion mit den gewünschten Parametern an. Um den Voraussetzungen und Wünschen verschiedener Benutzer gerecht zu werden, bietet der Kommando-Modus drei unterschiedlich detaillierte Bedienformen.
- "Masken-Modus": Wie Abbildung 9 zeigt, werden mehrere Abfragen zu einer Bildschirmmaske zusammengefasst. Zusätzliche Informationen über die (entsprechend der aktuellen Cursorposition) möglichen Eingaben

Die EPOS Benutzeroberfläche

werden im unteren Fenster angezeigt. Wird eine Maske aufgerufen, so werden die zuletzt in dieser Maske eingetragenen Werte automatisch wieder eingesetzt.

In beiden Dialogformen kann der Benutzer zu jeder Zeit über eine Hilfe-Funktion zusätzliche Informationen über das Dialogsystem und die möglichen Eingaben abrufen.

Die Kommunikation kann durch benutzerdefinierte Makros, die mehrere Anwendungen zusammenfassen, vereinfacht werden.



Abbildung 9 - Beispiel Maskendialog von EPOS

5.3 Eingabe von EPOS-Spezifikationen

Sämtliche Informationen können in Textform, entsprechend der Syntax und Semantik der Spezifikationssprachen EPOS-R, -S und -P eingegeben werden. Das EPOS-System bietet hierzu die Möglichkeit, jeden auf dem Zielrechner verfügbaren Texteditor einzubinden.

Eine weitere Möglichkeit der Dateninteraktion steht mit dem „Datenbankbrowser“ zur Verfügung. Damit ist es möglich, in graphischen Darstellungen am Bildschirm einzelne Objekte auszuwählen und weitere Informationen über diese Objekte abzurufen, z.B. weitere graphische Darstellungen, oder das Objekt mit Hilfe des Editors zu ändern. Nach der Änderung kann die graphische Darstellung entsprechend den Änderungen aktualisiert werden, ohne den Bediendialog dafür zu wiederholen.

6 Die EPOS-Werkzeuge

6.1 Unterstützung von Entwurfsmethoden

Entsprechend seiner Zielsetzung ist EPOS nicht auf eine Entwurfsmethode beschränkt. Die hier vorgestellten Werkzeuge bieten Rechnerunterstützung für verschiedene, vom Anwender einstellbare Entwurfsmethoden.

Die für ein Entwicklungsprojekt jeweils günstigste Entwurfsmethode kann abhängig sein:

- von der Art der Aufgabenstellung
- von der Projektphase
- von der Art der organisatorischen Abwicklung eines Projekts

Die Vorgehensweise beim Entwurf mit EPOS-S richtet sich dann nach der gewählten Entwurfsmethode.

- Beim **Funktions-orientierten Entwurf** wird auf der Ebene 1 das zu entwerfende System als Entwurfsobjekt vom Typ AKTION spezifiziert. Diese Aktion wird dann auf den folgenden Entwurfsebenen in Aktionen verfeinert, wobei gleichzeitig der Ablauf (d.h. der Kontrollfluss) festgelegt wird. Die in den Aktionsbeschreibungen referenzierten Entwurfsobjekte, wie z.B. Daten, Schnittstellen usw., werden anschließend spezifiziert.
- Wird EPOS im Rahmen der Anforderungsanalyse verwendet, so kann hier der **Datenfluss-orientierte Ansatz** eingestellt werden. Bei der „Strukturierten Analyse“ (nach Yourdon/DeMarco) werden Entwurfsaufgaben mit Hilfe von Datenflussgraphen in Teilaufgaben (sog. "Prozesse") zerlegt, die dann ihrerseits weiter verfeinert werden.
- Beim **Datenstruktur-orientierten Entwurf** nach Jackson (sog. JSP-Methode als Abkürzung für Jackson strukturierte Programmierung) wird in folgenden Schritten vorgegangen: 1. Entwurf aller Datenstrukturen; 2. Untersuchung und Beseitigung von Strukturkonflikten ; 3. Abbildung aller Datenstrukturen auf eine gemeinsame Programmstruktur; 4. Überführung in einen Pseudocode.
- Beim **Objekt-orientierten Entwurf** (OOD) wird das zu entwerfende Softwaresystem aus Objekten aufgebaut, die sich aus Informationen (Da-

Die EPOS-Werkzeuge

ten) und Operationen zusammensetzen. EPOS unterstützt den objektorientierten Entwurf in der speziellen Ausprägung „HOOD“ (Hierarchical Object Oriented Design). Im Gegensatz zu einigen OOD-Verfahren, bei denen alle betrachteten Objekte auf derselben Ebene (mit gleichen Rechten) stehen, wird bei HOOD das Prinzip der Hierarchisierung eingeführt. Das bedeutet, dass Objekte zerlegt werden können in weitere Objekte. EPOS bietet die erforderlichen Darstellungsmittel zur Verwendung von HOOD.

- Beim **Ereignis-orientierten** Entwurf werden zunächst die azyklisch oder zyklisch auftretenden Ereignisse und die zugehörigen Prozesse („Tasks“) als Aktionen beschrieben. Schon auf den oberen Entwurfsebenen werden hierbei die erforderlichen Synchronisierungsbedingungen spezifiziert. Das weitere Vorgehen ist dann gleich wie beim funktionsorientierten Entwurf, d.h. die in den Aktionen referenzierten Entwurfsobjekte werden anschließend spezifiziert.
- Beim **Aufgabenbereichs-orientierten Entwurf** (auch als "Modulorientierter Entwurf" bezeichnet) werden die zu lösenden Aufgabenstellungen nach bestimmten Gesichtspunkten (z.B. nach dem dafür vorgesehenen Bearbeiter oder nach logischer Zusammengehörigkeit) zu Aufgabenbereichen zusammengefasst. Hierbei wird im allg. eine weitgehende Abgeschlossenheit der Aufgabenbereiche, eine eindeutige Definition der Schnittstellen und die Einhaltung des Prinzips des "Information Hiding" angestrebt. Anschließend werden dann die zwischen den Aufgabenbereichen auszutauschenden Informationen (Daten) und Funktionen (Procedures) definiert.
- Beim **Anlagen-/Geräteorientierten Entwurf** wird zunächst die Gerätestruktur mit Hilfe der Entwurfsobjekte vom Typ Ausführungseinheit spezifiziert. Die weitere Verfeinerung erfolgt dann für die Rechnerhardware und für die auf diesen Rechnern ablaufende Software getrennt: Die Hardware wird in Entwurfsobjekte vom Typ Ausführungseinheit verfeinert, während die Software nach einer der oben erwähnten Entwurfsmethoden (z.B. funktions-orientiert) entworfen wird.

Zur Unterstützung einer Vorgehensweise nach den oben erläuterten Entwurfsmethoden kann über das Bediensystem die im betreffenden Fall gewählte Entwurfsmethode eingestellt werden. Mit Hilfe entsprechender Programme wird

dann der Entwickler geführt und unterstützt. Gleichzeitig können mit Hilfe von EPOS-A Entwurfsmethoden-abhängige Konsistenzprüfungen durchgeführt werden.

Für den Datenfluss-orientierten Ansatz stehen dem Anwender noch weitere Hilfsmittel zur Verfügung

- Schnittstellen zu Werkzeugen, die auf der Datenfluss-orientierten Methode aufgebaut sind (CORE)
- Umwandlung der Darstellung bei Verwendung des Datenfluss-orientierten Ansatzes in andere Notationen derselben Methode (Notation nach Yourdon/DeMarco oder MASCOT).

6.2 Fehleranalysen

6.2.1 Fehlervermeidung und frühzeitige Fehlerentdeckung

Untersuchungen in DV-Projekten haben ergeben, dass zwei Drittel aller Fehler, die im Laufe der Systemlebensdauer festgestellt werden, Fehler aus den frühen Entwicklungsphasen (konzeptionelle Fehler und Entwurfsfehler) sind, deren Behebung erfahrungsgemäß mit sehr hohem Aufwand verbunden ist. Die Anwendung von EPOS bringt folgende Vorteile:

- Weitgehende Vermeidung von Fehlern während der Konzeptions- und Entwurfsphasen durch
 - rechnergestützte Verwendung bewährter Entwurfsmethoden
 - projektbegleitende, rechnergestützte Produktion von Entwicklungsdokumenten zum Ausschluss von Fehlern durch unzureichende Kommunikation
- frühzeitige Aufdeckung von Konzeptions- und Entwurfsfehlern durch automatische Prüfprogramme in allen Phasen des Projekts
- Unterstützung der Qualitätssicherung bei der Durchsetzung vorgegebener Entwicklungsrichtlinien durch Festschreibung der Phasenergebnisse und rechnergestützte Produktion der Phasendokumente
- Vermeidung von Fehlern in der Phase Codierung durch rechnergestützte Umsetzung des Entwurfs in die Zielsprache

Die EPOS-Werkzeuge

- Vermeidung von Entwurfsfehlern durch zu große Interpretationsspielräume
- Vermeidung von Systemunzulänglichkeiten durch mangelhafte Umsetzung der Anforderungen

6.2.2 Analyse von Anforderungen

Die in der Spezifikationsprache EPOS-R formulierten Eingaben werden mit dem Fehler-Analyse-Toolsystem folgenden Prüfungen unterzogen:

- Prüfung auf syntaktische Richtigkeit,
- Prüfung auf Verträglichkeit mit den Einträgen in der EPOS-Datenbank,
- Prüfung von Entscheidungsprozessen auf Redundanz, Eindeutigkeit und Vollständigkeit
- Prüfung von Zustandsprozessen auf Redundanz, Eindeutigkeit und Vollständigkeit
- Prüfung der Beschreibung von Dialogsystemen (Dialogmasken) auf Redundanz, Vollständigkeit und Eindeutigkeit

Darüber hinaus prüft das Fehler-Analyse-Toolsystem, ob und in welchem Umfang identifizierbare Aufgabenkomponenten ersetzt oder quittiert oder sogar redundant quittiert werden, ebenso, bei welchen Aufgabenkomponenten die Spezifikation fehlt.

Dabei wird auch die Konsistenz zwischen fachlichen Anforderungen, technischen Anforderungen und den Entwürfen (System- und SW-Architektur, Softwareentwurf) überprüft: Nach Abschluss der jeweiligen Aktivitäten wird ermittelt, ob die in den fachlichen bzw. technischen Anforderungsdokumenten angegebenen Anforderungen und Voraussetzungen (d.h. die identifizierbaren Aufgabenkomponenten) in den Entwürfen (EPOS-S) berücksichtigt (quittiert) werden.

6.2.3 Feststellung von Entwurfsfehlern

Die Fehlerprüfungen umfassen folgende Prüfungen von EPOS-S Beschreibungen:

- Überprüfung der Grundeigenschaften : Vollständigkeit, konsistente Namensverwendung (Namenskonflikte) , nicht referenzierte (isolierte) Entwurfsobjekte
- Überprüfung entwurfsmethodenspezifischer Eigenschaften wie: Fehlerhafte Prozeduraufrufe, Konsistenz im Kontrollfluss, vollständige Freigabe von Ereignissen, Einhaltung des Gültigkeitsbereichs von Daten, undefinierte Angaben, usw.,
- Überprüfung hierarchischer Eigenschaften: Typ-Konsistenz, hierarchische Vollständigkeit, hierarchische Konsistenz des Gültigkeitsbereichs bei Daten,
- Überprüfung auf Vorhandensein optionaler Beschreibungsteile,
- Überprüfung der Synchronisierungen
- Überprüfung der Umsetzbarkeit in die Zielsprache (z.B. in Ada)

6.3 Metriken

Nicht allein die Tatsache, dass eine bestimmte Funktionalität eines Anwendungssystems korrekt realisiert wurde, ist heute wesentlich, sondern es zählt auch die „Güte“ der Implementierung. Mit Hilfe von Metriken lassen sich nicht-funktionale Leistungsmerkmale in einem Anwendungssystem beurteilen. Metriken setzen in aller Regel bereits in den Entwurfsphasen ein, um möglichst frühzeitig im Zuge der Entwicklung Schwachstellen aufzudecken.

EPOS bietet hierzu eine Reihe von Metrikanalysen an:

- Die Hierarchieanalyse untersucht Aufrufbeziehungen
- Mit der Modulanalyse kann eine Bewertung des modularen Aufbaus eines Entwurfs beurteilt werden
- Die Schlüsselwort- und Umfanganalyse erlaubt Rückschlüsse auf die Komplexität einzelner Entwurfsobjekte
- Die Kopplungsanalyse untersucht die Verknüpfungen von Entwurfsobjekten

Die EPOS-Werkzeuge

- Die Namensanalyse ermöglicht die Beurteilung hinsichtlich der Verwendung aussagekräftiger („sprechender“) Bezeichner
- Eine Übersichtsanalyse stellt unterschiedliche Metriken im Zusammenhang dar:
 - Gesamtvolumen
 - Systemtiefe
 - Modulgröße
 - Funktionalitätskopplung
 - Vernetzung
 - Globaler Datenfluss
 - Externe Schnittstellen
 - Operandenzahl
 - Zyklomatische Komplexität
 - Parallelitätsgrad
 - Synchronisationsgrad
 - Anweisungsichte
 - Namenslänge

Die Ergebnisse der Metriken können wahlweise in Tabellenform oder als Grafik (Abbildung 10) dargestellt werden.

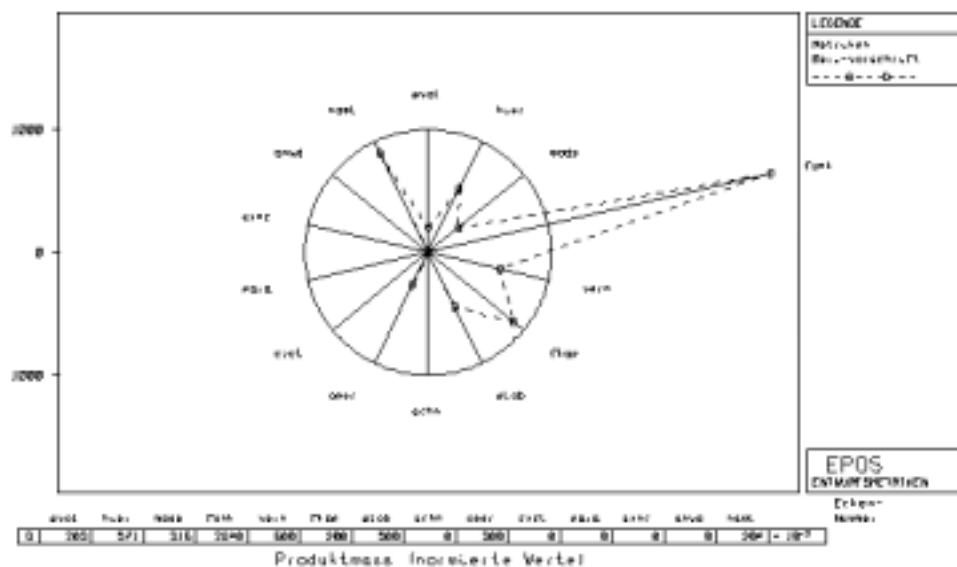


Abbildung 10 - Metriken in EPOS: Beispiel Übersicht

6.4 Dokumentation

Die Dokumentationsfunktionen bieten zwei Dokumentations-Varianten:

- die sog. "Standard-Dokumentation" zur automatischen Erzeugung von Einzeldokumenten aus den in der Datenbank abgespeicherten Informationen
- die Verwendung des Dokumenten-Generators zur automatischen Erzeugung von projektübergreifend definierten Dokumenten, die mit Hilfe einer speziellen Dokumentationsprache bezüglich Struktur, Layout und Inhalt spezifiziert werden. Dabei können z.B. textuelle und grafische Dokumentationsteile einer Dokumentenseite kombiniert werden. Solche Dokumente können zum Beispiel die Phasendokumente sein, die jeweils am Ende eines Entwicklungsschritts vorzulegen sind.

Für beide Varianten der Dokumentationserzeugung gilt: Die mit EPOS produzierte Dokumentation gibt stets den aktuellen Stand der Entwicklung wieder. Das zeitaufwendige "Nachziehen" der Dokumentation entfällt dadurch bzw.

Die EPOS-Werkzeuge

wird vom Rechner automatisch auf der Basis der aktuellen Entwurfsinformationen durchgeführt.

6.4.1 Variante 1 – Standarddokumentation (Anforderungen)

Die Beschreibung der fachlichen bzw. technischen Anforderungen wird in formatfreier Form erstellt und - nach einer Syntax- und Verträglichkeitsprüfung - in der EPOS-Datenbank abgelegt. Mit Hilfe der Dokumentationsfunktionen kann der Rechner daraus folgende Dokumente auf entsprechenden Ausgabegeräten (Drucker, Bildschirm) erzeugen:

- Gesamt-Dokumentation mit Inhaltsverzeichnis ,
- Ausschnitts-Dokumentation,
- Liste der identifizierbaren Aufgabenkomponenten (gekennzeichnete Anforderungen und Randbedingungen), geordnet nach verschiedenen Kriterien,
- Begriffslexikon, alphabetisch geordnet,
- Auszug aus dem Lexikon oder aus der Liste der identifizierbaren Aufgabenkomponenten, nach einer bestimmten Kategorie ausgewählt,
- Stichwortverzeichnis,
- Begriffsliste,
- Change-Management-Informationen (Revisionsstand, Änderungsvorgänge).

6.4.2 Variante 2 - Standarddokumentation (Entwurfsinformationen)

Als Darstellungsmittel zur Dokumentation des Systementwurfs werden verwendet:

- graphische Darstellungsmittel,
- Darstellungsmittel zur textuellen Dokumentation.

Zur textuellen Darstellung des Entwurfs bieten die EPOS-Dokumentationsfunktionen mehrere unterschiedliche Aufbereitungsformen:

- Projektdokumentation
- Entwicklungsdokumentation.

Eine Besonderheit ist in diesem Zusammenhang die sogenannte „natürlichsprachliche Dokumentation“, bei der die formalen Teile eines Entwurfsobjekts automatisch in einen verständlichen Text umgesetzt werden.

Der Mensch hat die Fähigkeit, Zusammenhänge, die sich verbal oft nur umständlich ausdrücken lassen, in einer bildlichen Darstellung mit einem Blick zu erkennen und zu verstehen. EPOS bietet aus diesem Grunde zahlreiche graphische Darstellungsmöglichkeiten, durch deren Anwendung die Lesbarkeit und Verständlichkeit der Dokumentation (und damit indirekt die Zuverlässigkeit und Wartbarkeit des operationellen Systems) verbessert werden kann.

Die folgenden Diagramm-Arten können mit den Dokumentationsfunktionen aus den in der EPOS-Datenbank abgelegten EPOS-S-Spezifikationen automatisch erzeugt und auf einem geeigneten Ausgabegerät (Plotter, Bildschirm, Drucker) ausgegeben werden:

- **Hierarchie-Diagramme** in drei verschiedenen Ausführungsarten: Standardform (vernetzt), Baumform (unvernetzte Form, ein Beispiel zeigt Abbildung 11).
- **Ablauf-Hierarchie-Diagramme**, die eine Kombination von Hierarchie-Diagrammen und Flussdiagrammen darstellen; neben der Verfeinerung von Entwurfsobjekten wird noch der Kontrollfluss dargestellt.

Die EPOS-Werkzeuge

- **Datenstruktur-Diagramme** als Kombination von Hierarchie-Diagramm für Daten und Diagramme der Datenbeziehungen. Die verwendete Symbolik für die Sequenz, die Iteration und die Selektion sind aus der Jackson-Methode übernommen.
- **Blockdiagramme**, um das Zusammenwirken verschiedener Typen von Entwurfsobjekten übersichtlicher darzustellen.
- **Flussdiagramme**, auch als Ablaufdiagramme oder Programmablaufpläne bezeichnet.
- **Datenflussdiagramme** zur graphischen Veranschaulichung des Datenflusses (Abbildung 12 zeigt ein Beispiel eines solchen Diagramms (Bildschirmausgabe)).
- **Modulverknüpfungspläne** zur Darstellung der spezifizierten Import/Export-Beziehungen zwischen Modulen.
- **Datenrelationsdiagramm** und Datenrelationsmatrix zur graphischen Darstellung der spezifizierten Relationen zwischen Daten.
- **Mascot-Diagramme** zur graphischen Darstellung des Entwurfs gemäß der Mascot Vorgehensweise.
- **CORE-Diagramme** als graphische Veranschaulichung der Spezifikation in der nach der CORE Methode vorgesehenen Form.
- **SA-Diagramme** zur Veranschaulichung des Datenflusses eines nach der SA-Methode orientierten Analyseergebnisses in der Notation nach Yourdon.
- **Nassi-Shneiderman-Diagramme** (Struktogramme) (Abbildung 13).
- **Petri-Netze** als Graphen zur Darstellung zeitlich parallel ablaufender Vorgänge und ihres Zusammenwirkens (Abbildung 14).
- **Hardware-Blockschaltbilder** zur grafischen Darstellung von Hardware-Einheiten (Abbildung 15).
- **SA-Diagramme**, wie oben, jedoch in der Notation nach Gane/Sarson (Abbildung 16).
- **Task-Rendezvous-Diagramme** zur Darstellung von Synchronisationsmechanismen nach dem Ada-Rendezvouskonzept.

- **HOOD-Diagramme** zur Darstellung der Zusammenhänge zwischen Objekten, die auf der Grundlage eines objekt-orientierten Entwurfs definiert wurden (Abbildung 17).
- **Aufruf-Hierarchiediagramme** („Structure Charts“ nach Gane/Sarson) zur Darstellung von hierarchischen Aufrufbeziehungen in einem nach „Structured Design“ erstellten Entwurf.

Die EPOS-Werkzeuge



Abbildung 11 - Beispiel eines Hierarchiediagramms in EPOS

Die EPOS Werkzeuge

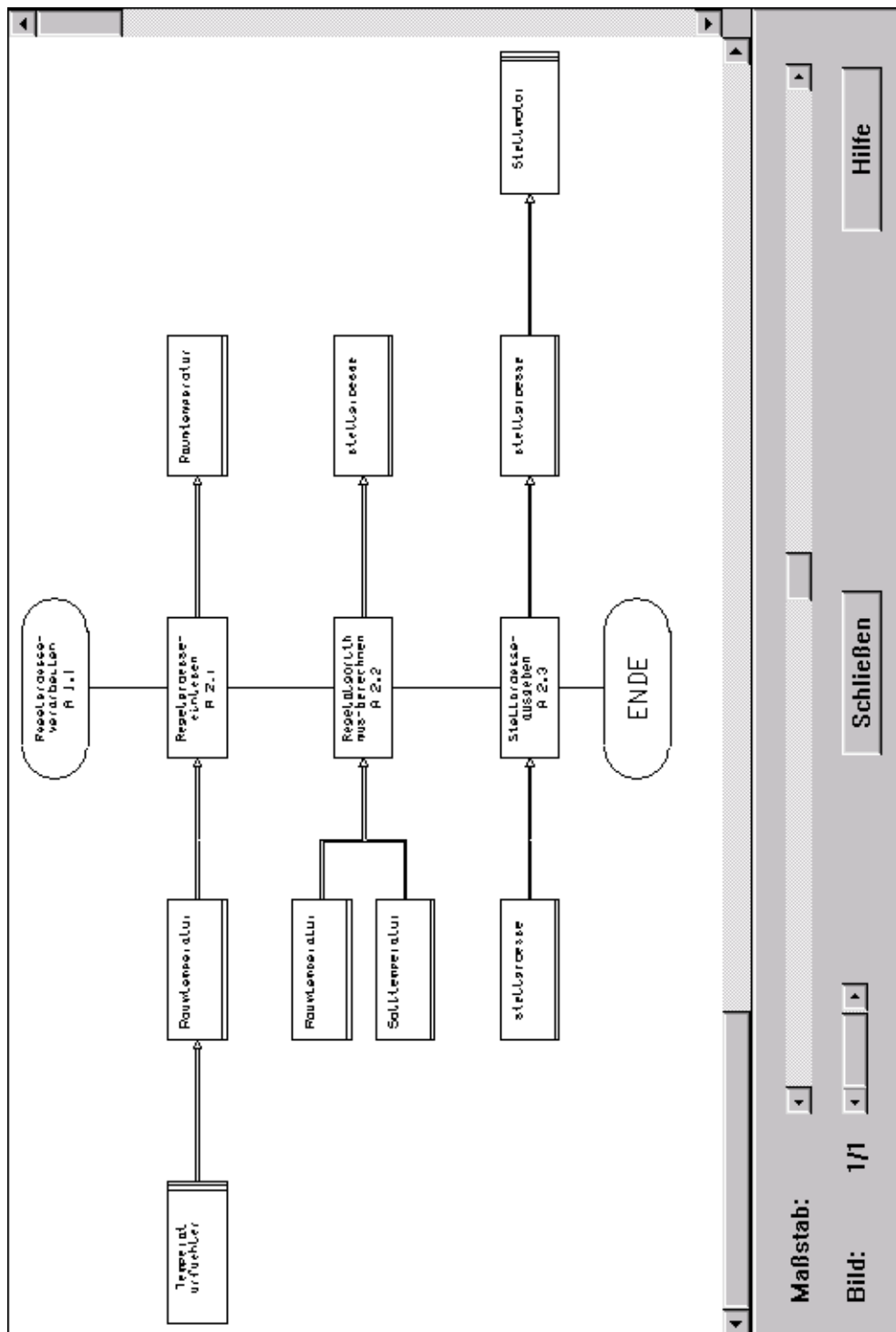


Abbildung 12 – Beispiel Datenflussdiagramm (aktionsorientiert)

Die EPOS-Werkzeuge

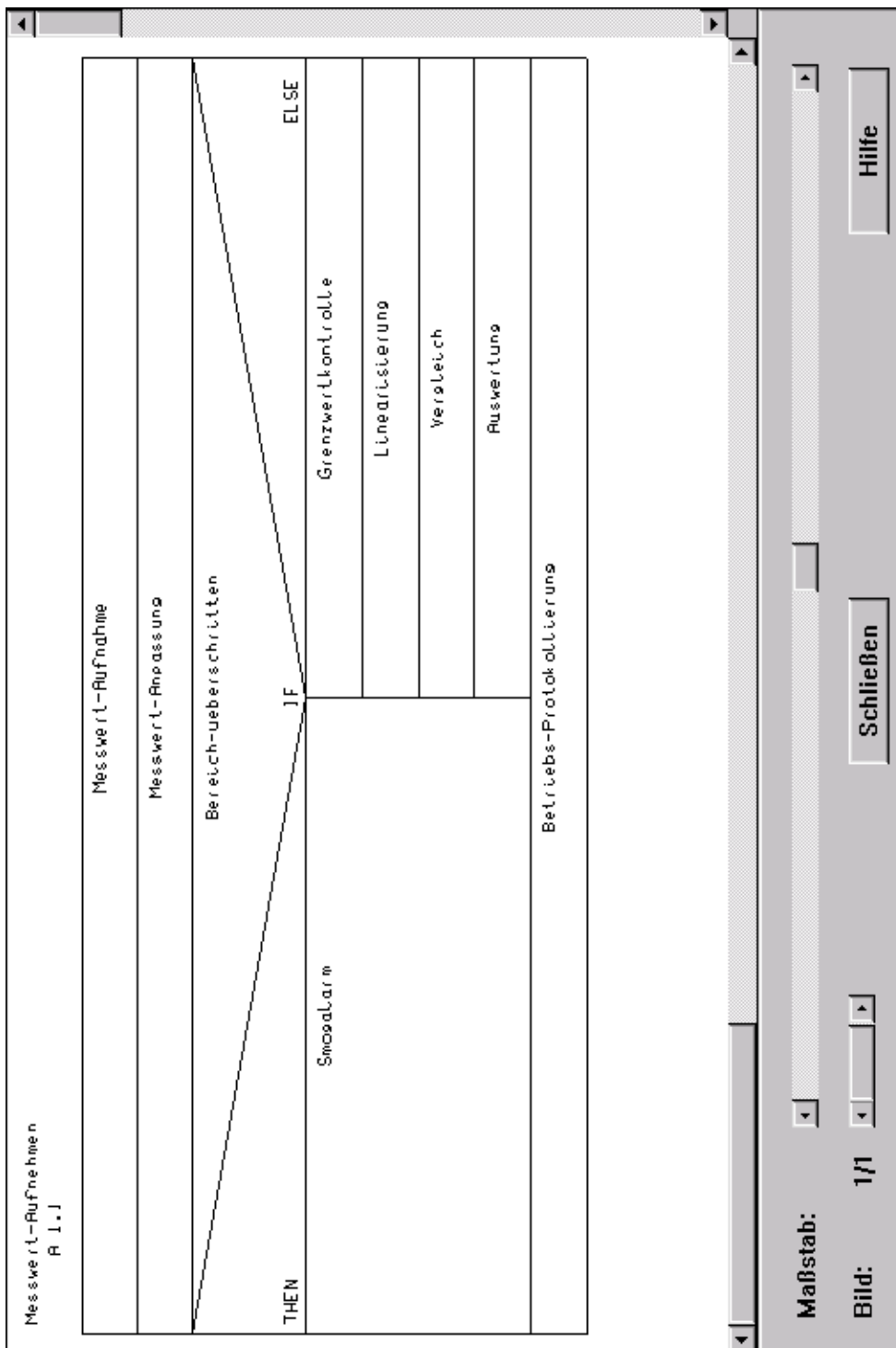


Abbildung 13 - Nassi-Shneiderman-Diagramm

Die EPOS Werkzeuge

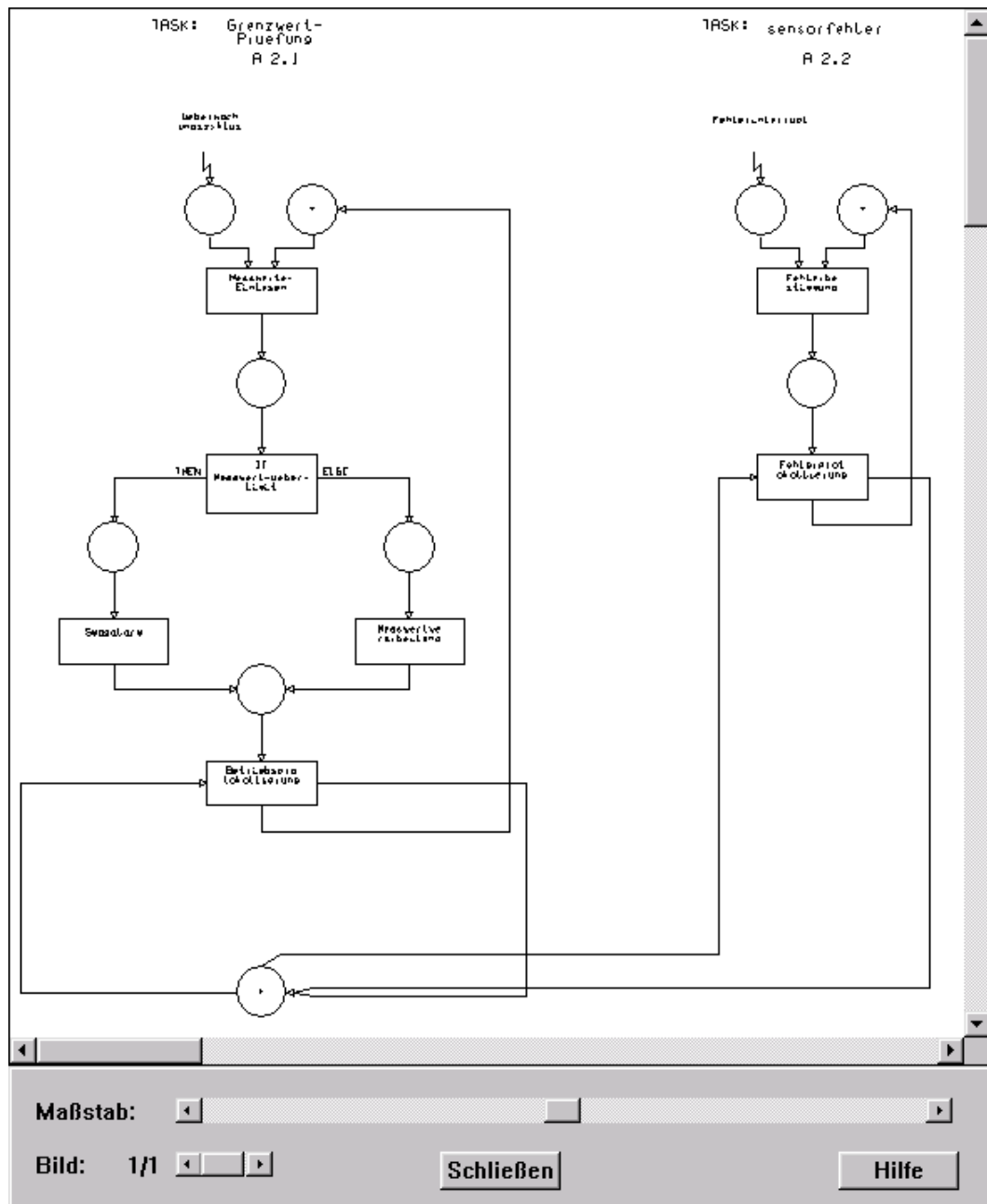


Abbildung 14 - Synchronisationsbeziehungen als Petrinetz

Die EPOS-Werkzeuge

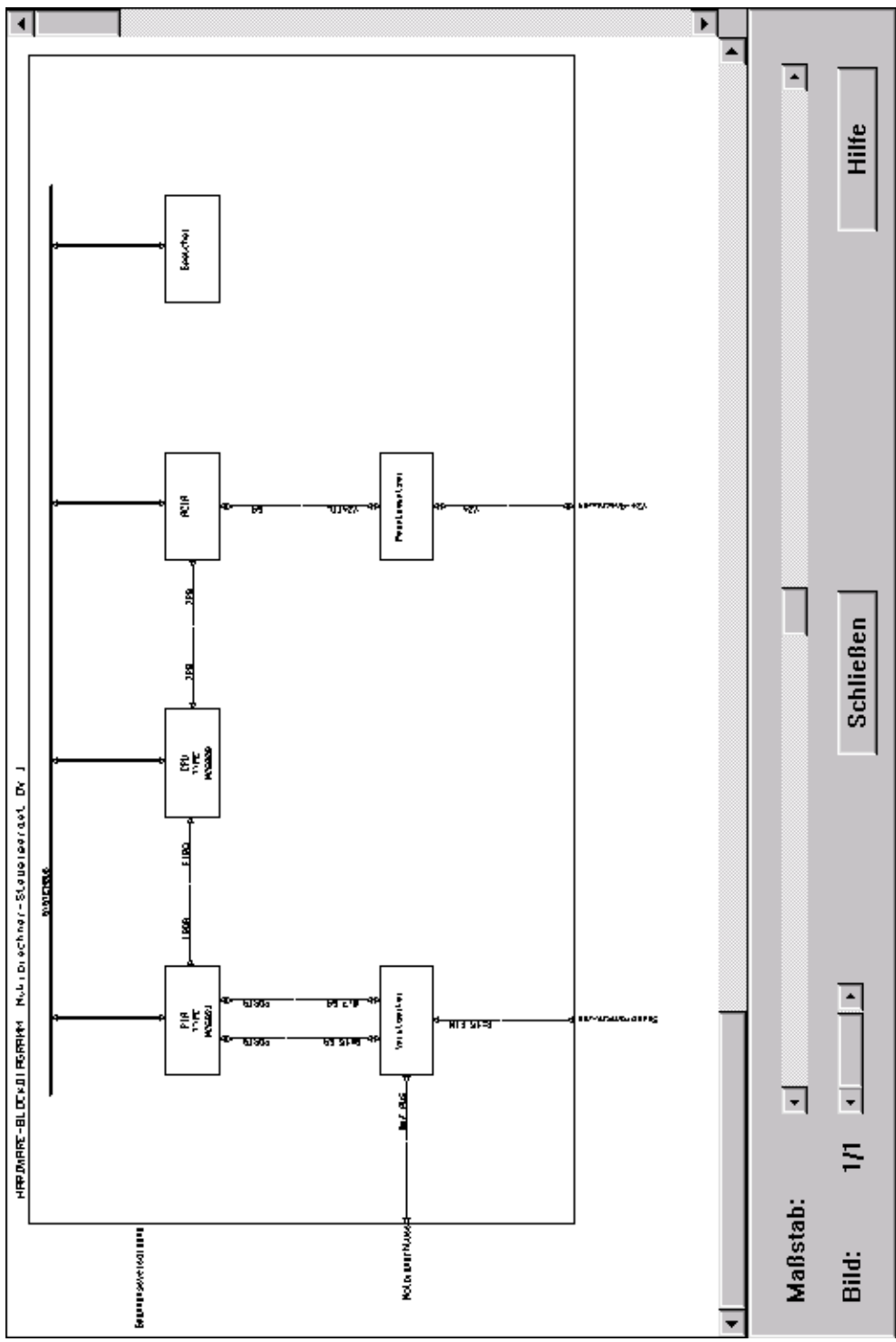


Abbildung 15 - Hardware-Blockschaltbild

Die EPOS Werkzeuge

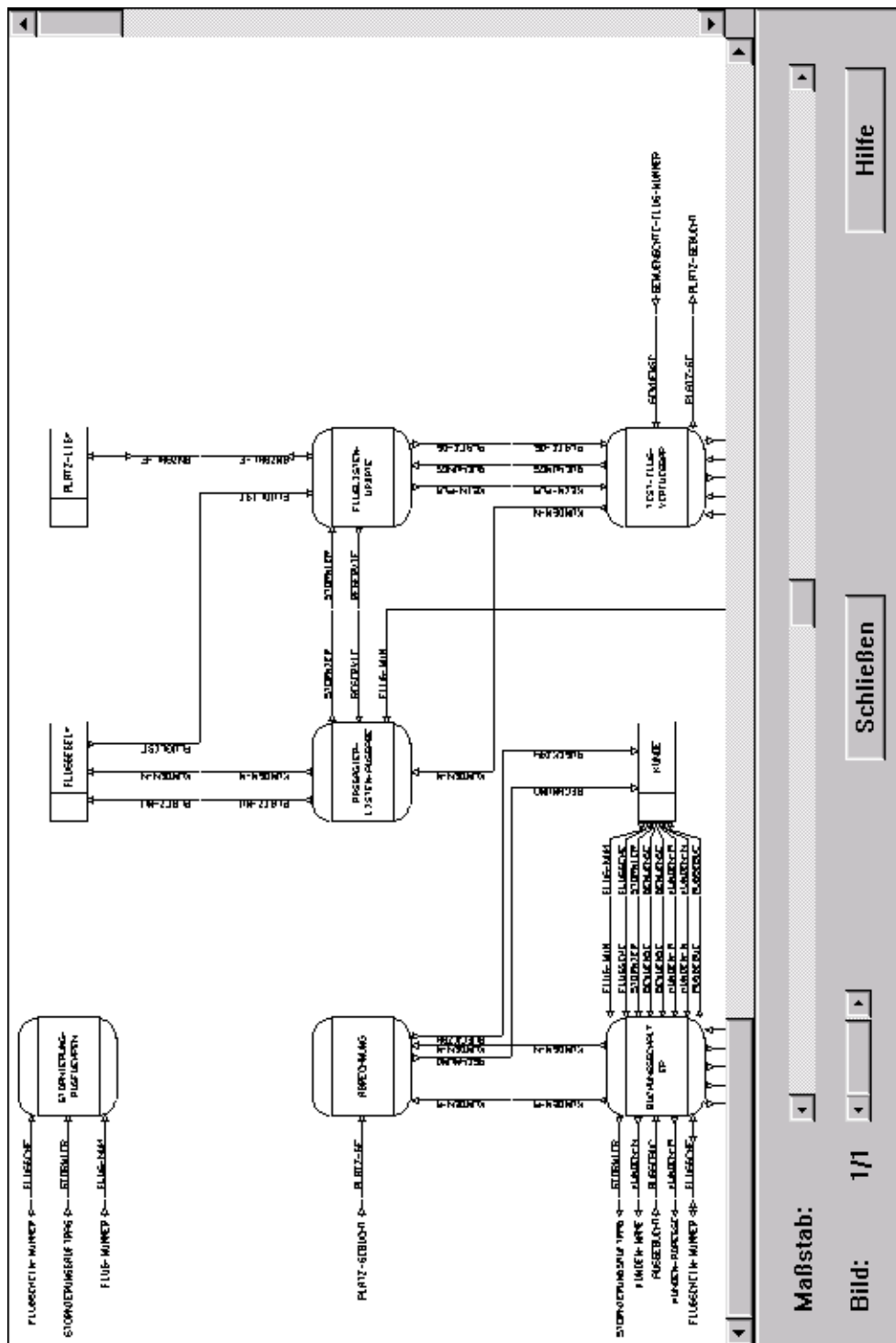


Abbildung 16 - Datenflussdiagramm (Notation Gane/Sarson)

6.4.3 Der Dokumentengenerator

Der EPOS-Dokumentengenerator bietet dem Anwender die Möglichkeit, spezielle Dokumente, in denen die Sachverhalte aus der EPOS-Datenbank dargestellt werden sollen, projektübergreifend festzulegen und diese dann automatisch zu generieren. Dabei werden Struktur, Layout und Inhalt allgemein in einem Dokumentenschema beschrieben. Aus diesem (vom Inhalt der EPOS-Datenbank völlig unabhängigen) Dokumentenschema erzeugt der Dokumentengenerator über eine vom Anwender noch beeinflussbare Zwischenstufe die festgelegten Dokumente, in denen dann vom Anwender ausgewählte textuelle und graphische Darstellungen kombiniert werden können und Informationen aus den unterschiedlichen EPOS-Bereichen (EPOS-S, EPOS-R und EPOS-P) auch gemischt werden können. Neben den in den Standarddokumenten enthaltenen Informationen können auch Auswertungen aus der EPOS-Datenbank miteinbezogen werden, so dass Dokumente entstehen, die alle Aspekte eines Projekts betreffen. Solche Dokumente sind zum Beispiel die im Vorgehensmodell beschriebenen Produkte.

Die vom Dokumentengenerator verwendete Layoutinformation kann vom Anwender frei definiert werden und auch bei der Generierung der Standarddokumente verwendet werden. Dabei ist es möglich, für jedes (Standard-) Dokument ein eigenes Layout vorzugeben.

Um die Struktur, das Layout und den Inhalt festlegen zu können, stehen entsprechende Beschreibungsmittel zur Verfügung.

6.5 Codegenerierung

6.5.1 Möglichkeiten der Codegenerierung

Die Generierung von Programmcode aus einer EPOS-S-Spezifikation weist gegenüber einem Vorgehen "von Hand" entscheidende Vorteile auf: Es geht nicht nur um den Aufwand beim Umsetzen der EPOS-S-Entwurfsobjekte bzw. der mit EPOS-D erzeugten Flussdiagramme oder Nassi-Shneiderman-Diagramme in den Code einer Programmiersprache. Viel wichtiger ist die Vermeidung der Fehler, die dabei begangen werden können. Dieser Gesichtspunkt ist besonders bei Änderungen im Entwurf von Bedeutung, wobei der Programmcode automatisch nachgeführt wird.

Die EPOS-Werkzeuge

Aus der mit EPOS-S formulierten Entwurfsspezifikation kann für beliebige Programmiersprachen ein Programmquellfile generiert werden. Dabei hängt es vom Abstraktionsgrad der jeweiligen Programmiersprache ab, in wieweit der Programmcode automatisch erzeugt wird Oder in wieweit ein interaktives Vorgehen erforderlich ist:

- für **anwendungsbezogene Programmiersprachen** (sog. "very high level languages") ist eine vollautomatische Codegenerierung möglich. Ein entsprechender Generator für C/ATLAS ist verfügbar.
- für die **höheren Programmiersprachen** Ada, PEARL, PASCAL, C und Fortran ist eine weitgehend automatische Codeerzeugung möglich. Lediglich für die mit EPOS nicht weiter verfeinerten Objekte ("terminale Objekte") muss direkt 'Zielsprachen Code' innerhalb dieser terminalen Objekte eingefügt werden. So kann für diese Programmiersprachen die gesamte Kontrollflussstruktur einschließlich aller erforderlichen Datendeklarationen automatisch erzeugt werden. Für Programmiersprachen mit entsprechenden Möglichkeiten (z.B. Ada) werden außerdem die Synchronisations-Anweisungen von EPOS in entsprechende Konstrukte der Zielsprache umgesetzt.
- für **beliebige Programmiersprachen** (d.h. für höhere Sprachen wie Modula, für niedere Programmiersprachen wie PLM, sowie insbesondere auch für Assemblersprachen) ist ein Verfahren zur interaktiven Codegenerierung verfügbar. Dabei müssen im Rahmen des EPOS-S-Entwurfs bei den Entwurfsobjekten der Typen Aktion, Bedingung und Date Programmsequenzen der betreffenden Programmiersprache im Code-Part manuell durch den Anwender eingegeben werden. Daraus erzeugt dann ein Generator ein ablauffähiges Programm.

6.5.2 Coderückführung

Beim Austesten von Programmen muss damit gerechnet werden, dass bei der Beseitigung von Fehlern, die aus der EPOS-S-Spezifikation generierten Quellfiles geändert werden. Solche Änderungen führen dazu, dass die Entwurfsbeschreibung in EPOS-S nicht mehr mit den tatsächlichen Quellprogrammen übereinstimmt.

Um diese Übereinstimmung wieder herzustellen, wird eine Coderückführung und eine automatische Aktualisierung der EPOS-Datenbank durchgeführt. Die EPOS-S-Beschreibung wird damit zur änderungsstabilen "Uniquelle".

6.5.3 Einsatz von EPOS in Programmierumgebungen

Das Endprodukt der Implementierungsaktivitäten bei der Softwareentwicklung ist ein Quellfile in einer bestimmten Programmiersprache. Im Rahmen der Software-Implementierung wird dieses Quellfile mit Hilfe eines Compilers bzw. Assemblers in Maschinencode übersetzt, mit anderen Programmteilen integriert und auf der Zielmaschine ausgetestet.

Dabei ist der Entwicklungsrechner nicht notwendigerweise mit dem Zielrechner identisch, d.h. Software-Entwurf und -Implementierung werden unter Umständen auf unterschiedlichen Rechnern durchgeführt.

Zur Bearbeitung von Programmen im Rahmen der Implementierung werden Software-Werkzeuge eingesetzt (wie z.B. Editoren, Compiler, Binder, Lader, Debugsysteme usw.), die auf die jeweilige Programmiersprache ausgerichtet sind. Die Gesamtheit dieser Werkzeuge wird als Programmierumgebung bezeichnet.

Das EPOS-System (als programmiersprachen-unabhängige Software-Entwicklungsumgebung) ergibt zusammen mit einer Programmierumgebung eine für die betreffende Programmiersprache geeignete Software-Entwicklungs- und Implementierungsumgebung. So wird man z.B. zur Entwicklung von PEARL-Programmen das EPOS-System in Verbindung mit einer PEARL-Programmierungsumgebung, zur Entwicklung von Programmen in Pascal in Verbindung mit einer Pascal-Programmierungsumgebung einsetzen.

Die EPOS-Werkzeuge

Für die Programmiersprache Ada wurden verschiedene Arten von Programmierumgebungen definiert: Die Kernumgebung KAPSE (Kernel Ada Programming Support Environment), die Minimalumgebung MAPSE (Minimal Ada Programming Support Environment) und die umfassende Ada-Umgebung APSE. EPOS eignet sich hervorragend zur Entwicklung von Ada-Programmen. Die Verbindung zwischen dem EPOS-System und einer "MAPSE" stellt damit eine leistungsfähige "APSE" dar.

6.6 Projektplanung

Die Planung eines Projekts beinhaltet

- die Analyse der technischen Anforderungen und die Aufstellung eines Lasten- bzw. Pflichtenhefts,
- die Definition von Arbeitspaketen mit Kosten- und Terminvorgaben als Grundlage für einen Projektstrukturplan, sowie für Termin- und Kostenpläne,
- die Festlegung der Mitarbeiter, Abteilungen oder Firmen, die an dem Projekt beteiligt sind (d.h. der "Projektbeteiligten"), und die Aufstellung eines Organisationsplans,
- die Zuordnung von Bearbeitern (Projektbeteiligten) zu Arbeitspaketen, sowie die Festlegung von Zuständigkeiten, Kompetenzen usw.,
- die Planung des zeitlichen Projektablaufs mit Hilfe eines Netzplans,
- die Zuordnung von (kontierten) Einsatzmitteln mit frei definierbarer Verrechnungseinheit zu Arbeitspaketen.

Die Elemente dieser Planungen (Aufgabenstellungen und Anforderungen im Lasten- bzw. Pflichtenheft, Arbeitspakete im Projektstrukturplan, Projektbeteiligte im Organisationsplan) werden mit Hilfe der Spezifikations Sprachen EPOS-R sowie EPOS-P beschrieben und in die EPOS-Datenbank eingegeben. Die Projektmanagementkomponente von EPOS wertet diese Daten aus und erzeugt

- Listen und Tabellen nach Sortierungs- und Auswahlkriterien,
- Matrizen zur Darstellung von Zuständigkeiten, Kompetenzen,
- Projektstrukturpläne,

- Produktflussdiagramme zur Darstellung der Verwendung von (Zwischen-) Ergebnissen innerhalb des Projekts,
- Balkenplane (Abbildung 19),
- Netzpläne (Abbildung 18),
- Tabellen zur Übersicht über die geplante Verteilung von Einsatzmitteln,
- Histogramme über die geplante Verwendung von Einsatzmitteln (Abbildung 20),
- Summationsdiagramme über den geplanten Einsatzmittelbedarf,
- Meilensteindiagramme usw.

6.7 Projektkontrolle

Die Projektsteuerung beinhaltet das Ermitteln von Abweichungen (Soll/Ist-Vergleich) und das Ergreifen von Maßnahmen. Grundlage ist hierbei die Kontrolle des Projektfortschritts und des laufenden Aufwands. Die Projektmanagementfunktionen von EPOS können zwei Arten von Darstellungen erstellen:

- textuelle und graphische Darstellungen über die Erreichung genau definierter Kontrollpunkte (wie z.B. Meilensteine, Abschluss von Netzplanaktivitäten), sowie über die bisherigen Aufwendungen. Dazu gehören Balkenplane und Listen/Tabellen, sowie Histogramme, die die aktuellen und geplanten Termine, abgeschlossene Aktivitäten, sowie angefallene Kosten verdeutlichen.
- Tabellen und Diagramme als Hilfen zur objektiven Beurteilung des Projektfortschritts zwischen Kontrollpunkten.

In "Fortschrittsdiagrammen" wird angezeigt, an welchen Teilen eines Entwurfsausschnitts die Arbeit beendet ist, wo weitere Entwicklungsarbeiten notwendig sind, und welche Teile des Entwurfsausschnitts innerhalb der letzten Zeit (seit einem Bezugsdatum) entstanden sind.

Diese und ähnliche Darstellungen zeigen die speziellen Möglichkeiten der integrierten Rechnerunterstützung von Entwicklung und Projektmanagement: Entwicklungsdaten können - in aufbereiteter Form - direkt dem Projektmanagement verfügbar gemacht werden.

Die EPOS-Werkzeuge

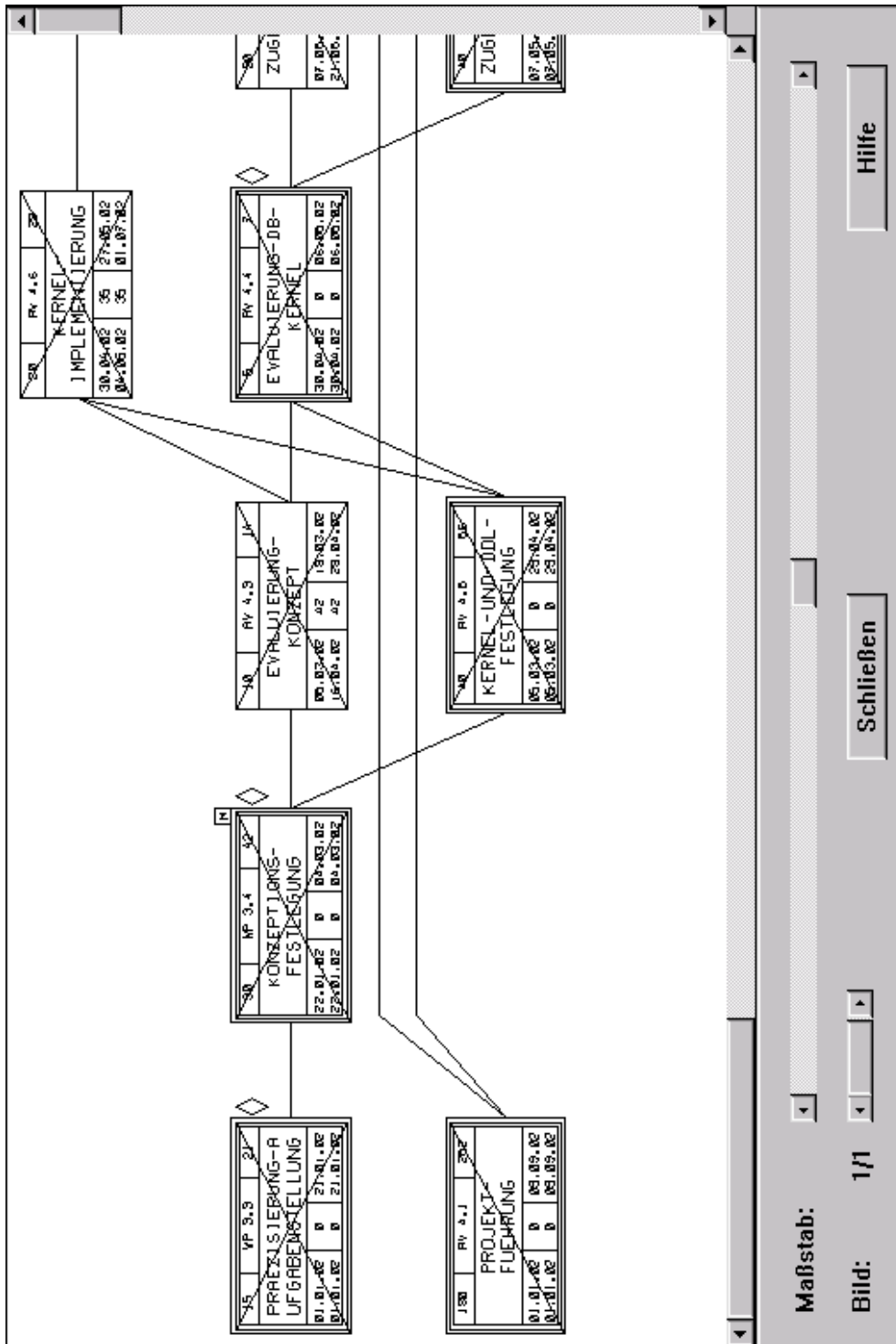


Abbildung 18 - Ausschnitt aus einem Netzplan

Die EPOS-Werkzeuge

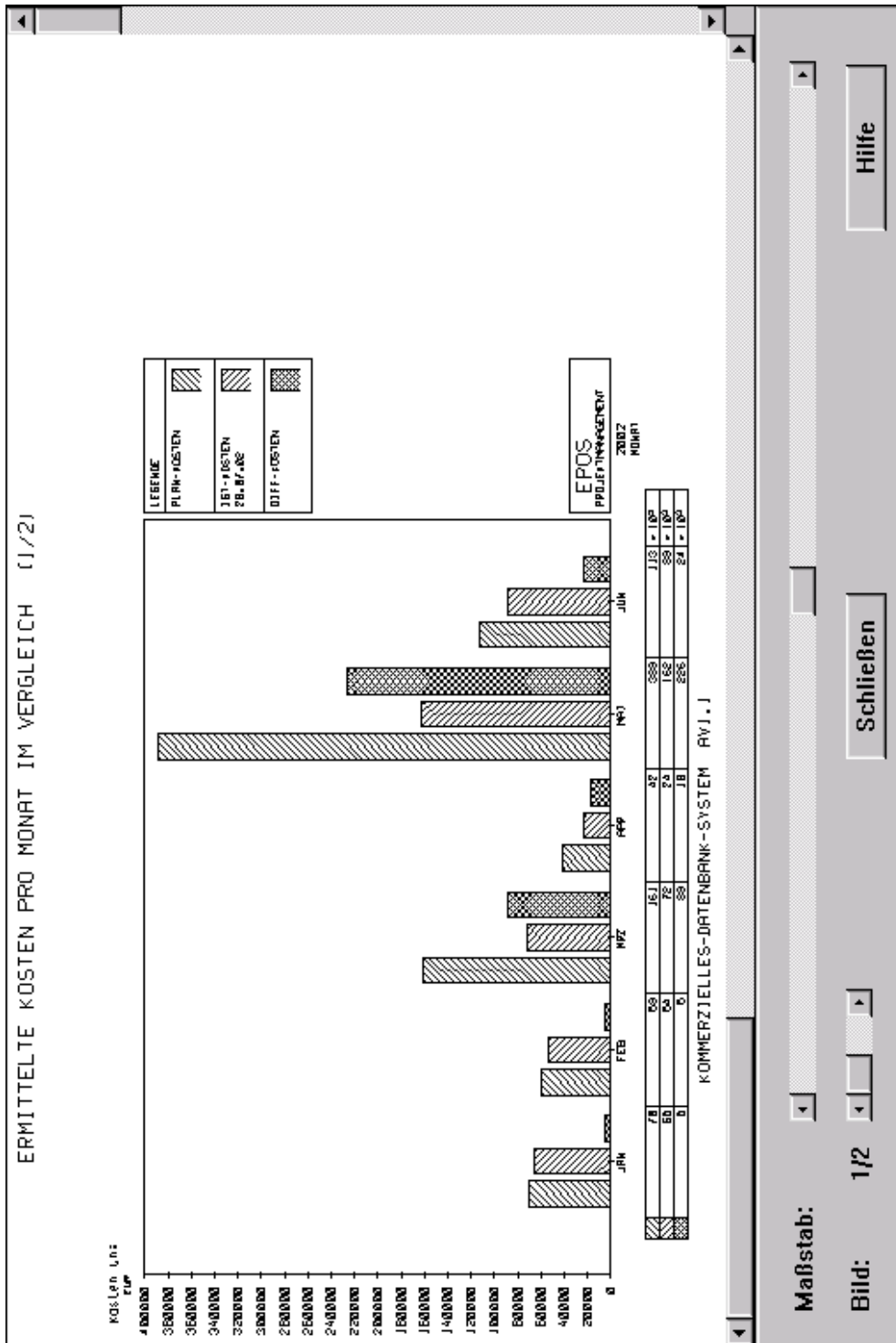


Abbildung 20 - Vergleich von Soll- und Istkosten



Die EPOS Werkzeuge

7 Verfügbarkeit von EPOS

EPOS ist auf den folgenden Einsatzplattformen verfügbar:

- Open VMS for VAX auf VAX-Anlagen (ehem. Digital Equipment Corp., jetzt HP)
- Open VMS for AXP auf Alpha AXP-Anlagen (ehem. Digital Equipment Corp., jetzt HP)
- Intel Pentium II, III und IV-Anlagen (oder kompatibel) unter Microsoft Windows NT 4.0 (Service Pack 6) bzw. Windows 2000

Weitere Einsatzplattformen auf Anfrage.



Verfügbarkeit von EPOS