EDV-unterstützte Durchführung von Geschäftsprozessen

DIPLOMARBEIT

zum Erwerb des Grades eines Diplom-Informatikers

Ralf Thormann
Schmiedingstrasse 10
44137 Dortmund

UNIVERSITÄT DORTMUND

Universität Dortmund, Fachbereich Informatik
Lehrstuhl Softwaretechnologie

Gutachter:

Erstgutachter : Prof Dr. E.-E. Doberkat

Zweitgutachter: Prof Dr. W. Schäfer

Dortmund, im Dezember 1994

An dieser Stelle danke ich allen, die mich bei der Erstellung meiner Diplomarbeit unterstützt haben. Für die Betreuung danke ich Prof. Dr. Ernst-Erich Doberkat. Mein Dank für wertvolle und konstruktive Kritik gebührt Dr. Volker Gruhn und Hans-Gerald Sobottka.

Dortmund, im Dezember 1994 Ralf Thormann

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einleitung

1.1 Motivation

Die Größe vieler Unternehmen macht heute eine systematische und effiziente Abwicklung von Geschäftsprozessen ohne EDV-Unterstützung unmöglich. Unter einem Geschäftsprozeß verstehen wir hierbei eine Menge von Aktivitäten, die miteinander in logischen und zeitlichen Abhängigkeiten stehen und alle zum Erreichen eines bestimmten Zustandes beitragen, wobei zumindest teilweise die Mitwirkung von Personen notwendig wird. Die Aktivitäten, die der Mitwirkung von Personen bedürfen, entsprechen dabei den Aufgaben der verschiedenen Sachbearbeiter einer Unternehmung.
Die Vielfalt der Geschäftsprozesse, wie Rechnungen stellen, Briefe schreiben, Adreßkarteien pflegen, Terminkalender führen, usw., hat in vielen Büros derart überhand genommen, daß eine Überschaubarkeit dieser Prozesse nicht mehr gegeben ist. Benötigt werden Geschäftsprozeß–Management–Systeme, die die Sachbearbeiter eines Unternehmens bei ihren Tätigkeiten innerhalb strukturierter Arbeitsabläufe über mehrere Arbeitsplätze hinweg unterstützen. Solche Systeme finden vor allem in letzter Zeit zunehmend Beachtung. Sie repräsentieren schon heute den Lebensnerv vieler Unternehmen und sind aus ihnen nicht mehr wegzudenken.

Neue Bestrebungen gehen dahin, Geschäftsprozeß–Management–Systeme zu entwickeln, die neben der Ausführung von Geschäftsprozessen auch deren Modellierung und Analyse innerhalb eines Gesamtsystems ermöglichen.

Dadurch erhöhen sich die Einsatzmöglichkeiten dieser Gechäftsprozeß–Management–Systeme, da jedes Unternehmen seine Geschäftprozesse selber modellieren und an die unternehmensspezifischen Gegebenheiten anpassen kann. Auf eventuelle Änderungen innerhalb der Geschäftsprozesse kann somit schnell reagiert werden, da die Prozesse von den Mitarbeitern des Unternehmens selbst gewartet werden können.

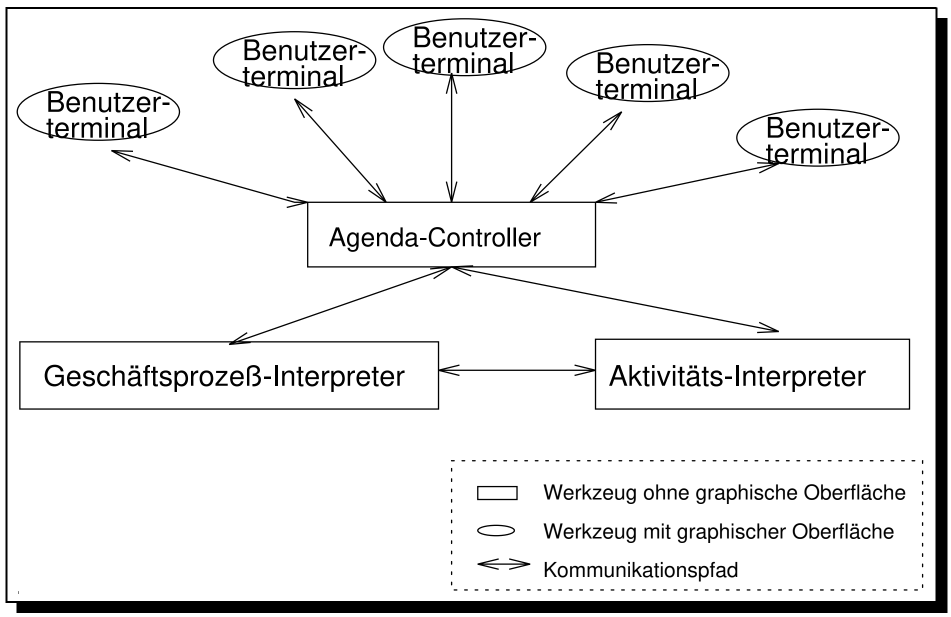


Abbildung 1.1: Allgemeiner Aufbau von Workflow–Management–Systemen

Viele Geschäftsprozeß–Management–Systeme sind als verteilte Systeme realisiert und in ihrem Aufbau ähnlich (siehe Abb. 1.1).

Sie bestehen aus:

- einem oder mehreren *Geschäftsprozeß-Interpretern*
  Die *Geschäftsprozeß-Interpreter* ermitteln die innerhalb der Geschäftsprozesse auszuführenden Aktivitäten.

- den *Aktivitäts-Interpretern*
  Die *Aktivitäts-Interpreter* übernehmen bzw. unterstützen die Ausführung der Aktivitäten, die innerhalb der Geschäftsprozesse zur Bearbeitung anstehen.

- den oder dem *Agenden-Controller*
  Die *Agenden-Controller* schreiben die innerhalb der Geschäftsprozesse zu bearbeitenden Aktivitäten, deren Bearbeitung personelle Interaktion erfordert, unter Berücksichtigung der Zuständigkeiten in die Agenden der Sachbearbeiter.

- den *Benutzerterminals*
  Die *Benutzerterminals* zeigen jedem Sachbearbeiter die von ihm zu erledigenden Aufgaben (Aktivitäten) an.

1.2 Begriffsdefinitionen

In diesem Abschnitt werden die für diese Arbeit grundlegenden allgemeinen Begriffe definiert. Weitere Begriffe, die für das Verständniss des in Kapitel 2 beschriebenen Geschäftsprozeß-Management-System *LEU* wichtig sind, werden im Abschnitt 2.2 erläutert. Auf sie wird im weiteren Text mit $\xrightarrow{2.2}$*Begriff* verwiesen. Auf die in diesem Abschnitt definierten Begriffe wird im weiteren Verlauf mit $\xrightarrow{1.2}$*Begriff* referenziert.

- **Agenden / ToDo-Listen**
  Die Agenden sind die "Arbeitseingangskörbe". Jeder Benutzer hat in seiner Agenda alle anstehenden Aufgaben eingetragen, die derzeit in den laufenden Geschäftsprozessen ausgeführt werden müssen und in seinen Zuständigkeitsbereich fallen. Innerhalb dieser Diplomarbeit, wird die Oberfläche eines Geschäftsprozeß-Management-Systems entwickelt, die eine solche Agenda darstellt.

- **Agenden-Controller**
  Der Agenden-Controller eines Geschäftsprozeß-Management-Systems schreibt die vom Geschäftsprozeß-Interpreter übertragenden manuellen $\xrightarrow{1.2}$Aktivitäten unter Berücksichtigung der individuellen Benut-

zerberechtigungen und Zuständigkeiten in die Agenden der Sachbearbeiter.

- **Agendeneintrag**
  Die in den Geschäftsprozessen zur Bearbeitung anstehenden manuellen Aktivitäten bilden die Agendeneinträge. Sie müssen von den Sachbearbeitern, die an den Geschäftsprozessen teilnehmen, bearbeitet werden.

- **Aktivitäten**
  Die Geschäftsprozesse bestehen aus manuellen und automatischen Aktivitäten, die miteinander in logischen und zeitlichen Abhängigkeiten stehen und in ihrer Gesamtheit den Geschäftsprozeß ausmachen. Für die Bearbeitung der manuellen Aktivitäten ist die Interaktion von Benutzern notwendig. Die manuellen Aktivitäten bilden die Agendeneinträge.

- **Berechtigungen**
  Die Teilnehmer von Geschäftsprozeß-Management-Systemen benötigen für die in ihre Zuständigkeit fallenden Tätigkeiten Berechtigungen. Sie werden in Geschäftsprozeßmanagement-System $\xrightarrow{1.2}$LEU mit Hilfe von $\xrightarrow{1.2}$Rollen definiert (siehe Abschnitt 2.3.4).

- **Geschäftsprozeß-Interpreter**
  Der Geschäftsprozeß-Interpreter eines Geschäftsprozeß-Management-Systems interpretiert zur Ausführungszeit die Geschäftsprozeßmodelle. Das bedeutet, daß in jedem Prozeßzustand die ausführbaren Aktivitäten von ihm ermittelt und zur Bearbeitung weitergeleitet werden.

- **Geschäftsprozeßmodelle**
  Geschäftsprozeßmodelle bilden die Grundlage für die auszuführenden Geschäftsprozesse. Zu einem Geschäftsprozeßmodell gehören alle Informationen, die benötigt werden, um einen Geschäftsprozeß auszuführen. Die Geschäftsprozeßmodelle bestehen somit aus den $\xrightarrow{2.2}$Ablaufmodellen, den Datenmodellen und dem $\xrightarrow{1.2}$Stellenmodell.

- **Geschäftsprozesse**
  Geschäftsprozesse bilden die Arbeitsabläufe in den Unternehmen. Sie bestehen aus manuellen und automatischen Aktivitäten, die miteinander in logischen und zeitlichen Abhängigkeiten stehen und in ihrer Gesamtheit den Geschäftsprozeß ausmachen. Geschäftsprozesse können im *SINGLE_USER_MODE* und im *MULTI_USER_MODE* ausgeführt werden. Während die im *SINGLE_USER_MODE* ausgeführten Geschäftsprozesse von nur einem Benutzer bearbeitet werden können, dürfen an der Ausführung der im *MULTI_USER_MODE* laufenden Geschäftsprozesse mehrere Personen beteiligt sein.

- **LEU**
  Geschäftsprozeßmanagement-System, das die Modellierung, Simulation und Ausführung von Geschäftsprozessen unterstützt. Diese Diplomarbeit bildet einen Teil der Ausführungskomponente dieses Systems. Das LEU-System wird in Kapitel 2 weiter beschrieben.

- **Organigramm / Stellenmodell**
  Das Organigramm bildet die Aufbaustruktur eines Unternehmens. Die Aufbaustruktur repräsentiert die Organisationsstrukturen, in die die Geschäftsprozesse eingebettet sind.

- **Rollen**
  Rollen dienen der Definition von Benutzerberechtigungen [War89]. Im Rahmen dieser Rollen können individuelle Berechtigungen definiert werden. So kann zum Beispiel mit der Rolle "Sekretärin" das Berechtigungsprofil für eine Sekretärin, mit der Rolle "Abteilungsleiter" das eines Abteilungsleiter usw. definiert werden.
  Um bei der Ausführung der Geschäftsprozesse zu gewährleisten, daß jeder Mitarbeiter genau die in seinen Zuständigkeitsbereich fallenden Aktivitäten in seine Agenda geschrieben bekommt, werden den verschiedenen Stellen innerhalb des Unternehmens Rollen zugewiesen. Die Mitarbeiter, die dieser Stelle zugehören, besitzen dann die Berechtigung, die sie durch die Stellen, die sie besetzen, bekommen haben.

- **Stellen**
  Jedes Unternehmen gliedert sich in eine Stellenhierarchie. Sie bildet

das $\xrightarrow{1.2}$Organigramm des Unternehmens. Jeder Mitarbeiter eines Unternehmens ist einer oder mehreren Stellen zugeordnet.

1.3 Ziel und Aufbau dieser Arbeit

Innerhalb des Implementierungsteils dieser Arbeit wurde die Benutzeroberfläche für die Ausführungskomponente eines Geschäftsprozeß-Management-System entwickelt und in das $\xrightarrow{1.2}$LEU-Sytem integriert. Hierbei wurde besonders auf eine ergonomische Benutzerführung sowie eine einfache und intuitive Bedienung des Werkzeugs geachtet. Des weiteren wurde innerhalb des Implementierungsteils ein Agenden-Controller entwickelt, der die innerhalb der Geschäftsprozesse anstehenden Aufgaben, unter Berücksichtigung der individuellen Benutzerberechtigungen, in die Agenden der einzelnen Sachbearbeiter schreibt.

Kapitel 1 beschreibt die Motivation für die Arbeit und gibt eine Einordnung in die bestehende Literatur. In Kapitel 2 wird das Geschäftsprozeß-Management-System $\xrightarrow{1.2}$LEU kurz beschrieben. Um einen Überblick über das System zu geben, sind alle Komponenten und Werkzeuge des Systems zumindest kurz erwähnt und die zum weiteren Verständnis der Arbeit notwendigen Abschnitte werden ausführlicher behandelt. In Kapitel 3 und 4 werden dann die beiden innerhalb dieser Arbeit entwickelten Werkzeuge (der Agenden-Controller und das Agenda-Terminal) näher erläutert. In den beiden gleich strukturierten Kapiteln werden zunächst die Anforderungen, die die Werkzeuge zu erfüllen haben, dargelegt und danach auf ihre Realisierung eingegangen. Die Entwicklung der Benutzeroberfläche wurde aus Kapitel 4 ausgelagert, da ihr Entwicklungsprozeß unter besonderen Bedingungen nach einem Prototyping-Verfahren verlief und eine gesonderte Betrachtung verdient. Mit ihr beschäftigt sich Kapitel 5. Kapitel 6 beschreibt die Testmethoden, die verwendet wurden, um die korrekte Funktionalität der Werkzeuge sicherzustellen.
Im abschließenden Kapitel 7 werden die wesentlichen Punkte der Arbeit kurz zusammengefaßt und ein Ausblick auf mögliche Erweiterungen gegeben.

Der Anhang unterteilt sich in zwei Teile.
Anhang A ist Bestandteil dieses Dokuments. Hier wird der Ablauf einer Geschäftsprozeß-Ausführung exemplarisch vorgestellt. Dieses Beispiel soll das Zusammenspiel der einzelnen Werkzeuge der LEU-Ausführungskomponente verdeutlichen und die innerhalb dieser Arbeit entwickelten Werkzeuge im praktischen Einsatz zeigen.
Anhang B ist ein eigenständiges Dokument.
Er enthält die Feinentwürfe des Agenda-Terminals und des Agenden-Controllers und die Protokolle der innerhalb der Interprozeß-Kommunikation (IPC) zwischen den Werkzeugen auszutauschenden Nachrichten.

1.4 Einordnung in die Literatur

Software-Prozeß-Management und Geschäftsprozeß-Management haben sich trotz großer inhaltlicher Nähe und beinahe identischer Ziele unabhängig voneinander entwickelt. Insbesondere bei den Ansätzen zur Entwicklung von Sprachen und Werkzeugen zum Software- und Geschäftsprozeß-Management ist es nur in sehr geringem Maße zu einem Austausch zwischen den beiden Bereichen gekommen. Dies ist umso erstaunlicher, da Software-Prozesse nur eine bestimmte Art von Geschäftsprozessen darstellen. Im folgenden werden Ursprünge und Ziele des Software-Prozeß-Managements und des Geschäftsprozeß-Managements diskutiert.

1.4.1 Software-Prozeß-Management

Software-Prozeß-Management ist ein Teilgebiet des Software-Engineerings, dem seit ca. 10 Jahren zunehmend Beachtung geschenkt wird. Ein Ziel des Software-Prozeß-Managements ist es, das Vorgehen in der Software-Entwicklung mit formalen Mitteln zu beschreiben. Das Ergebnis einer solchen Beschreibung wird als Software-Prozeßmodell bezeichnet. Im Rahmen des Software-Prozeß-Managements werden Software-Prozeßmodelle analysiert (beispielsweise durch Simulation von Prozessen) und letztendlich als Grundlage realer Software-Prozesse benutzt. Da ein Software-Prozeß nicht vollständig durch das ihm zugrunde liegende Modell bestimmt ist, wird der Begriff der Ausführung von Software-Prozeßmodellen in der Regel vermieden und statt dessen mit dem kaum übersetzbaren Enaction (In-Gang-Setzen) bezeichnet.

Viele Projekte in Europa (ALF [Zuc91], [GJLO89], SPADE [BFGG92], MERLIN [EJP+91], [PS92], MELMAC [Gru91a], IPSE2.5 [War89]) haben zu Prozeßmodellierungssprachen und Werkzeugen zur Unterstützung des Software-Prozeß-Managements geführt. Mittlerweile hat sich eine gemeinsame Terminologie bezüglich der wichtigsten Begriffe des Software-Prozeß-Managements etabliert [Lon93]. Eine nennenswerte industrielle Verbreitung von Software-Prozeß-Management-Ansätzen hat bisher jedoch nicht stattgefunden. Akademisch geprägte Projekte und Experimente haben die Nützlichkeit von Software-Prozeßmodellierung und Software-Prozeßanalyse belegt, lassen jedoch eine gewisse Skepsis gegenüber der Software-Prozeß-Enaction als begründet erscheinen. Die bisher gesammelten Erfahrungen deuten darauf hin, daß Software-Prozesse nach wie vor nicht im Detail verstanden sind und stark von der Kreativität und Flexibilität der an der Entwicklung beteiligten Personen geprägt sind. Ansätze zur rechnergestützten Durchführung von Software-Prozessen werden oft als Einschränkung der Kreativität verstanden und stoßen somit auf Akzeptanzprobleme. [Leh87], [Leh89]

Chroust unterscheidet verschiedene Arten von Prozessen gemäß ihrem Anteil an Routinearbeit, gemäß ihrer Strukturierung und gemäß ihrer Automatisierbarkeit. Er kommt zu dem Ergebnis, daß Software-Prozesse zu den weniger verstandenen und strukturierten Prozessen gehören und somit einem systematischen Management im obigen Sinne nur insofern zugänglich sind, als daß sie modelliert und analysiert werden können [Chr92]. Durchführungsunterstützung scheint demgegenüber eher für gut verstandene, stark strukturierte und von Routinearbeit geprägte Prozesse anwendbar zu sein.

1.4.2 Geschäftsprozeß-Management

Fast unabhängig vom Software-Prozeß-Management hat sich in der Wirtschaftsinformatik das Management von Geschäftsprozessen (auch als

Workflow-Management bezeichnet) als ein Gebiet mit wachsender Bedeutung herausgebildet. Im Geschäftsprozeß-Management geht es um die Modellierung, Analyse und Durchführung (im Sinne von Enaction) von beliebigen Geschäftsprozessen. Ein Geschäftsprozeß ist dabei eine logisch zusammengehörige Menge von Aktivitäten, die einem bestimmten Geschäftszweck dienen. Unter diese allgemeine Beschreibung fallen natürlich auch die zu Beginn diskutierten Software-Prozesse. Allerdings beschränken sich die Anwendungen des Geschäftsprozeß-Managements in den allermeisten Fällen auf kaufmännische und administrative Bereiche.

Das Management gerade dieser Geschäftsprozesse gewinnt zunehmend an Bedeutung [Dav93], [HC93]. Die Gründe dafür sind vielfältig. Zum einen sind Geschäftsprozesse oft nicht transparent und verständlich und werden mehr oder weniger zufällig in einer bestimmten Art und Weise durchgeführt, so daß eine systematische Betrachtung dringend erforderlich scheint. Zum anderen gibt es eine Reihe von, die Organisation von Unternehmen betreffende Tendenzen, die eine Orientierung an Geschäftsprozessen unabdingbar werden lassen. Beide Arten von Gründen werden im folgenden diskutiert.

In den westlichen Industriegesellschaften hat sich in zahlreichen Bereichen eine hochgradig bürokratische Arbeitsweise etabliert. Organisatorische Einheiten arbeiten einen Teil eines gesamten Prozesses ab und übergeben das Ergebnis an die nächste Einheit, was ein fehlendes Gesamtverständnis für den durchzuführenden Prozeß zur Folge haben kann. Flexible Reaktion wird erschwert, weil es zwischen den beteiligten Einheiten keine einheitliche Sicht auf den Prozeß gibt [HC93]. Solche Prozesse splitten sich oftmals in stark reglementierte Teile und solche, die nicht näher spezifiziert und somit in keinster Weise reglementiert sind, auf. Gerade für Prozesse, die von ihrer Art her wenig reglementiert sind, gilt, daß ohne Prozeßorientierung viele typische Prozesse (beispielsweise Auftragsabwicklung) derart durchgeführt werden, daß die *Fäden* erst bei der Geschäftsführung zusammenlaufen. Jede beteiligte Stelle kümmert sich um ihre Funktion, aber der übergreifende Geschäftsprozeß gerät aus dem Blickfeld.

Qualität von Geschäftsprozessen bedeutet auch, daß die richtigen Informationen zum richtigen Zeitpunkt vorhanden sind. Die Beschwerde eines Kunden kann einfacher bearbeitet werden, wenn der Sachbearbeiter unmittelbar Zugriff auf die bisherigen Kundendaten hat. Die Bestellung eines Produkts kann besser behandelt werden, wenn frühere Sonderwünsche des Bestellers von vornherein mit berücksichtigt werden können. Solche Standardprozesse sind qualitativ zu verbessern, indem den Prozeßbeteiligten die entsprechenden Informationen zur Verfügung gestellt werden. Da die qualitative Verbesserung von Geschäftsprozessen in einem wettbewerbs-geprägten Umfeld als ein entscheidendes Kriterium angesehen wird, gewinnt das Management von Geschäftsprozessen allein aufgrund seines Qualitätssteigerungspotentials an Bedeutung.

Geschäftsprozeß-Modellierungssprachen und Werkzeuge zur Unterstützung des Geschäftsprozeß-Managements sind in zunehmendem Maße verfügbar. Im Gegensatz zu Softwareprozeß-Modellierungssprachen sind entsprechende Geschäftsprozeß-Modellierungssprachen im allgemeinen weniger komfortabel und weniger problemangepaßt. Graphische Modellierung ist die Ausnahme und die Verwendung formaler Spezifikationssprachen ganz selten zu finden.

Aus den genannten Gründen steht bei den meisten Werkzeugen zur Unterstützung des Geschäftsprozeß-Managements auch nicht die komfortable Modellierung, sondern die Durchführung realer Geschäftsprozesse im Vordergrund. Ein gegenseitiger Austausch zwischen Software-Prozeß-Management und Geschäftsprozeß-Management erscheint daher, insbesondere aufgrund der unterschiedlichen Schwerpunkte beider Gebiete, ebenso vielversprechend wie notwendig.

Kapitel 2

Die Lion-Entwicklungsumgebung

LEU steht für **L**ion–**E**ntwicklungs–**U**mgebung, die zur Zeit bei der **LION GmbH** in Bochum entwickelt wird. Ziel ist es, LEU ab 1995 unter anderem bei verschiedenen Unternehmen der Bau- und Wohnungswirtschaft einzusetzen, um bau- und wohnungswirtschaftliche Geschäftsprozesse in der Praxis zu unterstützen. Die erste LEU-Projektphase ist beendet und die Modellierungskomponenten werden bereits in der Praxis eingesetzt, um Geschäftsprozesse aus dem Bereich der Bau- und Wohnungswirtschaft zu modellieren. Die Analyse- und Ausführungskomponenten werden innerhalb der zweiten LEU-Projektphase (bis Ende 1994) entwickelt und mit den Modellierungswerkzeugen zum LEU-System integriert. Es ist zu erwähnen, daß der Anwendungsbereich der bau- und wohnungswirtschaftlichen Geschäftsprozesse exemplarisch herausgegriffen wurde, da zur Zeit Arbeiten von Anwendungsentwicklern in diesem Bereich stattfinden. Letztendlich sind die in LEU zur Verfügung stehenden Sprachmittel und Komponenten jedoch ebenso für das Management anderer Geschäftsprozesse geeignet, die sich durch die Teilnahme von Personen einerseits und durch einen engen Bezug zu den zugrundeliegenden Informationssystemen andererseits auszeichnen.

Im folgenden Abschnitt 2.1 werden die Architektur und die Werkzeuge von LEU vorgestellt. Abschnitt 2.2 enthält die Definition wichtiger LEU-Begriffe. In 2.3 wird näher auf die LEU-Modellierungswerkzeuge, in 2.4 auf die Simulation und Analyse von LEU und in 2.5 auf die Ausführungswerkzeuge der **LION-Entwicklungsumgebung** eingegangen.

Weitere Informationen zu LEU können [Gru93a] entnommen werden.

2.1 Die LEU-Architektur

LEU ist eine Umgebung, die sowohl datenintegriert (durch ein durchgängig verwendetes Data Dictionary) als auch oberflächenintegriert (durch ein durchgängig verwendetes Fenstersystem) sowie kontrollintegriert (durch eine gemeinsame Ablaufsteuerung) ist [Som92], [KPS93].

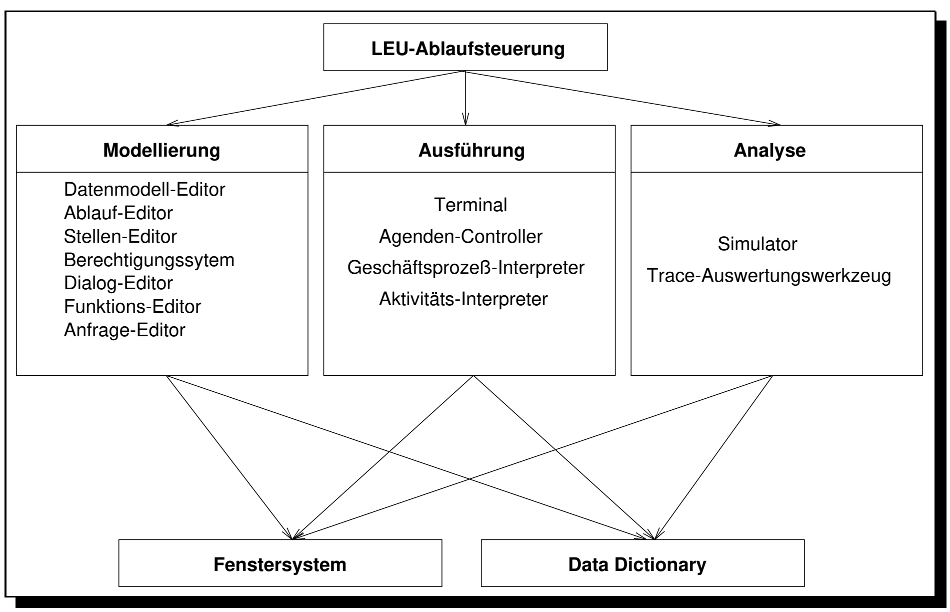


Abbildung 2.1: Architektur der LION-Entwicklungsumgebung

Die Abbildung 2.1 zeigt die Architektur von LEU. Aus der Ablaufsteuerung können die LEU-Werkzeuge für die Modellierung, Ausführung und Analyse gestartet werden. Auf der untersten Ebene der Hierarchie steht das Fenstersystem, das mit Hilfe des *ISA-Dialogmanagers* [ISA92] entwickelt wurde, und das Data-Dictionary, das in der relationalen Datenbank *ORACLE* abgelegt wird. Mit dem *ISA-Dialogmanager* wurde eine graphische Benutzeroberfläche mit möglichst einheitlichem *Look and Feel* für alle LEU-Werkzeuge realisiert. Das Data-Dictionary stellt ein Modell des Informationssystems dar, in dem alle Komponenten ihre Daten ablegen.

LEU hat zwei Gruppen von Benutzern, die *Anwendungsentwickler* und die *Anwendungsbenutzer*. Jede Gruppe arbeitet mit einem Teil der LEU–Werkzeuge. Die Anwendungsentwickler realisieren mit LEU die Applikationen für die Anwendungsbenutzer. Sie benutzen dazu die Werkzeuge für die *Modellierung* und *Analyse*. Die zweite Gruppe der LEU-Benutzer bilden die Anwendungsbenutzer. Die Anwendungsbenutzer sind die *Sachbearbeiter* in den Unternehmen, die mit den erstellten $\xrightarrow{2.2}$LEU-Anwendungen arbeiten. Die Begriffe *Anwendungsbenutzer* und *Sachbearbeiter* werden im folgenden Text synonym benutzt. Die LEU-Ausführungswerkzeuge steuern die Ausführung der für sie erstellten Applikationen.

2.2 Wichtige LEU-Begriffe

In diesem Abschnitt werden die für das Verständnis des LEU-Systems wichtigen Begriffe definiert. Auf diese hier erläuterten Begriffe wird im weiteren Text mit $\xrightarrow{2.2}$*Begriff* verwiesen.

- **Ablaufmodelle**
  Die Ablaufmodelle bilden die Grundlage für die ausgeführten Geschäftsprozesse. Sie werden in LEU als $\xrightarrow{2.2}$FUNSOFT-Netze modelliert (siehe Abschnit 2.3.1). In einem Ablaufmodell werden die logischen und zeitlichen Abhängigkeiten, die zwischen den Aktivitäten eines Geschäftsprozesses bestehen, festgelegt. Eine Eigenschaft der Ablaufmodelle ist die *Inkarnationsfähigkeit*. Unter einem inkarnationsfähigen Ablaufmodell versteht man ein Ablaufmodell, zu dem

zur Ausführungszeit mehrere parallel laufende $\xrightarrow{1.2}$Geschäftsprozesse ausgeführt werden können.

- **Agendensprünge**
  Da in vielen Geschäftsprozessen, an deren Ausführung mehrere Sachbearbeiter beteiligt sind, Abschnitte existieren, die logisch so zusammenhängend sind, daß sie von nur einem Anwendungsbenutzer bearbeitet werden können, bietet LEU die Möglichkeit, Agendensprünge für diese Abschnitte zu definieren. Ist für einen Abschnitt ein Agendensprung definiert, so werden alle Aktivitäten des Abschnitts von genau einem Benutzer durchgeführt.

- **FUNSOFT-Netze**
  FUNSOFT-Netze sind abstrakte Petri-Netze [HRV84]. Einige der in FUNSOFT-Netzen realisierten Konzepte haben ihren Ursprung in Funktionsnetzen [God83]. FUNSOFT-Netze haben eine eindeutige, auf Standard-Petri-Netzen basierende Semantik, die mit Hilfe von Prädikat/Transitions-Netzen [Gen87] formal definiert ist [Gru91b]. FUNSOFT-Netze erlauben eine kompaktere Darstellung von Ablaufmodellen als beispielsweise Prädikat/Transitions-Netze oder Coloured Petri Nets [Jen87], da sie unter besonderer Berücksichtigung der Anforderungen an Prozeßmodellierungssprachen entworfen wurden [EG91].

- **Geschäftsprozesse**
  Unter dem schon in Abschnitt 1.2 definierten Begriff des *Geschäftsprozesses* verstehen wir in LEU die Ausführung einer Ausprägung (im Sinne einer Datenbank-Kopie) des zugehörigen $\xrightarrow{1.2}$Geschäftsprozeßmodells.

- **Instanzen**
  In $\xrightarrow{2.2}$FUNSOFT-Netzen werden die $\xrightarrow{1.2}$Aktivitäten von Geschäftsprozeßmodellen als Instanzen dargestellt. Die manuellen Instanzen erfordern bei ihrer Ausführung die Interaktion von Benutzern, wohingegen die automatischen Instanzen ohne Benutzerinteraktion durch das System ausgeführt werden können.

Des weiteren unterscheidet man zwischen *prädikatbehafteten* und *nicht prädikatbehafteten Instanzen.* Zur Ausführung (zum Schalten, zum Feuern) einer nicht prädikatbehafteten Instanz genügt es, daß alle Vorbereichskanäle ($\xrightarrow{2.2}$Kanäle) gefüllt sind. An prädikatbehaftete Instanzen ist eine Prädikatsfunktion angebunden. Sie macht die Ausführung der Instanz von objektspezifischen Bedingungen abhängig. Zum Schalten einer prädikatbehafteten Instanz reicht es nicht aus, daß alle Vorbereichskanäle gefüllt sind, sondern die Objekte in den Vorbereichskanälen müssen zusätzlich den Bedingungen der Prädikatfunktion genügen.

- **Kanäle**
  Kanäle stellen die Objektspeicher dar. Die Kanäle bilden die Vor- und Nachbereiche der Instanzen eines FUNSOFT-Netzes. (Der Vorbereich einer Instanz besteht aus den Kanälen, auf die die Instanz lesend bzw. kopierend zugreift. Der Nachbereich einer Instanz besteht aus den Kanälen, die nach dem Schalten der Instanz gefüllt werden.) Demgemäß spricht man von den Vor- und Nachbereichskanälen einer Instanz. Die Kanäle in FUNSOFT-Netzen werden mit den Objekttypen des zum Geschäftsprozeß gehörenden Datenmodells typisiert. Ein Kanal kann nur Objekte des Objekttyps aufnehmen, für den er typisiert wurde. In FUNSOFT-Netzen bilden die Kanäle also die Objektspeicher für die in den Geschäftsprozessen bearbeiteten Daten. Der Vorbereich, respektive die Vorbereichskanäle einer Instanz, sind entscheidend für ihre Aktivierbarkeit. Eine Instanz kann schalten (kann aktiviert werden), sobald ihr Vorbereich gefüllt, daß heißt alle Vorbereichskanäle mit mindestens einem Objekt besetzt sind und gegebenenfalls die Objekte der Prädikatsfunktion genügen. Nach der Beendigung der Instanz werden Objekte in ihre Nachbereichskanäle geschrieben.

- **Konflikte in FUNSOFT-Netzen**
  Konflikte in FUNSOFT-Netzen entstehen dadurch, daß ein $\xrightarrow{2.2}$Kanal im Vorbereich mehrerer Instanzen liegt. Abbildung 2.2 zeigt einen sol-

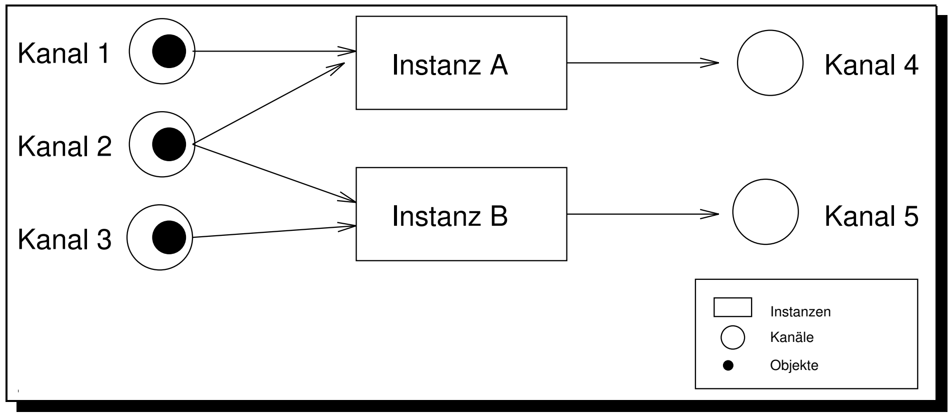


Abbildung 2.2: Konfliktbeispiel in einem FUNSOFT-Netz

chen Konflikt. *„Kanal 2"* liegt sowohl im Vorbereich von *„Instanz A"* als auch im Vorbereich von *„Instanz B"*. Die Vorbereiche beider Instanzen sind komplett gefüllt und so können beide Instanzen schalten. Entsteht dieser Konflikt zwischen manuellen Instanzen, so wird die Instanz ausgeführt, die von den Benutzern, die diese Aktivität in ihren Agenden stehen haben, zuerst zur Bearbeitung ausgewählt wird. Entsteht der Konflikt zwischen zwei automatischen Instanzen, so wird er vom $\xrightarrow{1.2}$Geschäftsprozeß-Interpreter selbständig gelöst. Bei der Modellierung der FUNSOFT-NETZE kann für diesen Fall eine Häufigkeitsverteilung angegeben werden, in wievielen Fällen zugunsten der einen oder zugunsten der anderen Instanzen entschieden werden soll. Im letztmöglichen Fall, daß der Konflikt zwischen manuellen und automatischen Instanzen besteht, können die manuellen Instanzen eine festgesetzte Zeitspanne in den Agenden von den Benutzern zur Bearbeitung ausgewählt werden, bevor der Konflikt zugunsten der automatischen Instanzen aufgelöst wird. Wird in unserem Beispiel der Konflikt zugunsten der *„Instanz A"* aufgelöst, so werden die Objekte aus *„Kanal 1"* und *„Kanal 2"* abgezogen und die *„Instanz A"* aktiviert. Nach der Konfliktlösung kann dann *„Instanz B"* nicht mehr schalten.

- **LEU-Anwendung**
  Eine LEU-Anwendung besteht aus allen zu einem Anwendungsbereich mit LEU modellierten Geschäftsprozeßmodellen.

- **Systemkanäle**
  Eine besondere Art von $\xrightarrow{2.2}$Kanälen bilden die Systemkanäle. Wie oben ausgeführt, werden die Kanäle mit den Objekttypen des zu dem Geschäftsprozeß gehörenden Datenmodells typisiert. In LEU existieren neben den *lokalen Objekttypen* eines Datenmodells auch *globale Objekttypen*. Die *globalen Objekttypen* können in allen zu einer $\xrightarrow{2.2}$LEU-Anwendung gehörenden Datenmodellen verwendet werden. Diese Objekttypen bilden die Schnittstellen zwischen den Geschäftsprozessen unterschiedlicher Datenmodelle. Die Systemkanäle werden mit diesen *globalen Objekttypen* typisiert. Mit ihrer Hilfe kann zwischen Geschäftsprozessen, die zu verschiedenen Datenmodellen gehören, kommuniziert werden.

2.3 Die LEU-Modellierung

In den folgenden Abschnitten werden die zur LEU-Modellierung gehörenden Werkzeuge näher beschrieben. Für das weitere Verständnis der Arbeit sind hierbei besonders die Ablauf- und Berechtigungsmodellierung relevant und werden deshalb ausführlicher erläutert.

2.3.1 Ablaufmodellierung

Die Ablaufmodellierung bildet den wesentlichen Bestandteil der $\xrightarrow{2.2}$LEU-Anwendungen. Mit der graphischen Benutzeroberfläche des Ablauf-Editors werden die Abläufe aller zu einer LEU-Anwendung gehörenden $\xrightarrow{1.2}$Geschäftsprozesse modelliert. Als zugrundeliegendes Beschreibungsmittel für die Geschäftsprozesse kommen, wie bereits ausgeführt, in LEU $\xrightarrow{2.2}$FUNSOFT-Netze zum Einsatz.
Abbildung 2.3 zeigt das Ablaufmodell des Geschäftsprozesses *Auftragserteilung*. Das Beispiel enthält die manuellen Instanzen *Auftragserfassung*, *Auftragskorrektur*, *Auftragsprüfung*, *Auftragssonderprüfung*, *Genehmigen nach*

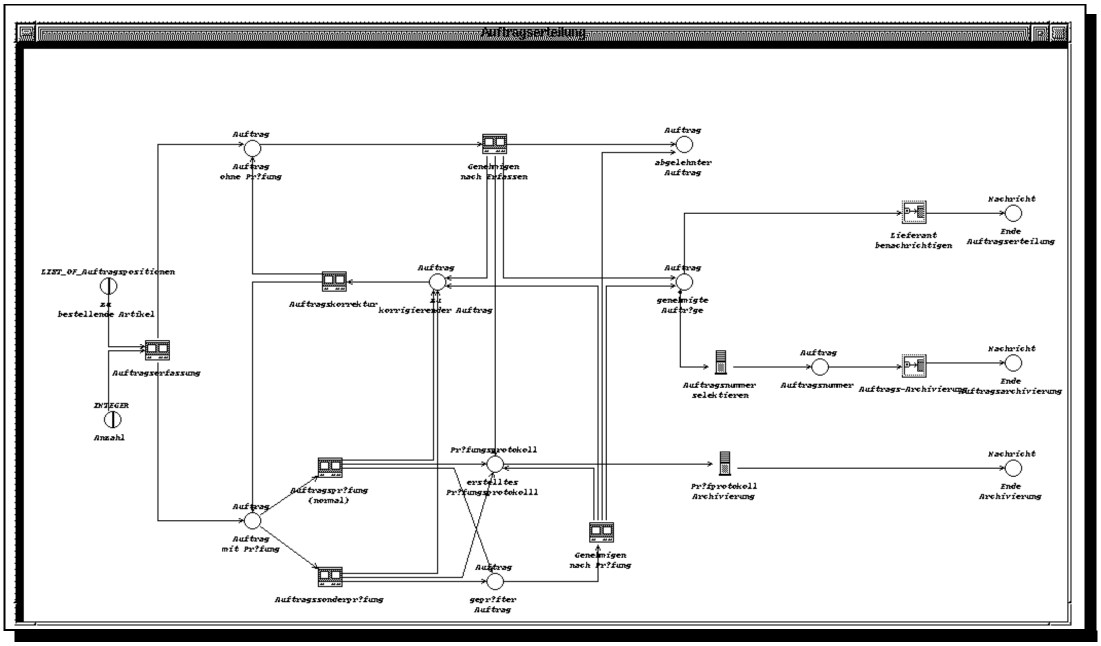


Abbildung 2.3: Beispiel eines Ablaufmodells

Erfassen und *Genehmigen nach Prüfung* (in der Abbildung durch das Symbol einer Dialogmaske dargestellt). Über manuelle Instanzen können beispielsweise Fremdwerkzeuge, wie eine Textverarbeitung, eine Tabellenkalkulation, etc. oder die mit dem LEU-Dialogeditor (s. Kap. 2.3.3) erstellten Dialoge, in die Geschäftsprozesse eingebunden werden.
Des weiteren enthält das Beispiel die durch das Rechnersymbol dargestellten automatischen Instanzen *Auftragsnummer selektieren* und *Prüfungsprotokoll Archivierung*. Die automatischen Instanzen der Geschäftsprozesse bestehen aus Funktionen der Programmiersprache C, die mit dem LEU-Funktionseditor (s. Kap. 2.3.5) implementiert bzw. aus Datenbankanfragen, die mit dem LEU-Anfrageeditor (s. Kap. 2.3.6) generiert wurden.
Die Instanzen *Lieferant benachrichtigen* und *Auftragsarchivierung* sind verfeinerte Instanzen. An die verfeinerten Instanzen werden Subnetze des Geschäftsprozeßmodells gehängt. Durch die Strukturierungsmöglichkeiten, die die verfeinerten Instanzen bieten, lassen sich auch große Geschäftsprozeßmodelle übersichtlich modellieren und darstellen.

Die in der Abbildung dargestellten Kreise sind die $\xrightarrow{2.2}$Kanäle (Objektspeicher) des Netzes. Die Pfeile, die die Kanäle und Instanzen miteinander verbinden, symbolisieren den Arbeitsablauf innerhalb des Geschäftsprozesses. Sie beschreiben die logischen und zeitlichen Abhängigkeiten der Aktivitäten im jeweiligen Geschäftsprozeß.

2.3.2 Die Datenmodellierung

Mit dem Datenmodell-Editor werden die Datenmodelle der $\xrightarrow{2.2}$LEU-Anwendungen erfaßt. In einem Datenmodell werden die einem Geschäftsprozeß zugrundeliegenden Objekttypen und deren Beziehungen definiert. Die Datenmodellierung arbeitet auf der Basis von Entity-Relationship-Modellen [Bar90]. In LEU wird für jedes Projekt (siehe Projektverwaltung Abschnitt 2.3.4.4 ein eigenes Datenmodell modelliert. Die zu diesem Projekt gehörenden Geschäftsprozesse arbeiten auf den Objekttypen dieses Datenmodells. Um auch einen projektübergreifenden Datenaustausch zwischen Geschäftsprozessen verschiedener Projekte zu ermöglichen, können neben den lokalen Objekttypen auch globale Objekttypen modelliert werden. Sie können aus anderen Datenmodellen importiert werden und bilden die Schnittstellen zum projektübergreifenden Datenaustausch zwischen Geschäftsprozessen.

2.3.3 Die Dialogmodellierung

Neben den bei der Datenmodellierung automatisch generierten *Standard-Dialogfenstern*, die die Datenerfassung zu genau einem Objekttyp ermöglichen, können mit dem Dialog-Editor *Individual-Dialoge* erstellt werden, mit denen dann beliebige Teile verschiedener Objekttypen und ihre Verbindungen innerhalb eines Dialogfensters bearbeitet werden können. Die Dialogfenster dienen der Datenerfassung und -verarbeitung.

2.3.4 Die Berechtigungsmodellierung

Jeder Geschäftsprozeß ist in eine organisatorische Struktur eingebettet. Sie wird mit dem LEU-Berechtigungssystem modelliert. Das LEU-

Berechtigungssystem umfaßt den Stellen-Editor, die Projektverwaltung, die Benutzerverwaltung und den Rollen-Editor. Mit dem Berechtigungssystem können sowohl die Modellierungsberechtigungen für die Anwendungsentwickler als auch die Ausführungsberechtigungen für die Anwendungsbenutzer vergeben und den in der Benutzerverwaltung erfaßten Personen zugewiesen werden. Im folgenden werden die Aufgaben der einzelnen Komponenten des LEU-Berechtigungssystems näher erläutert.

2.3.4.1 Die Benutzerverwaltung

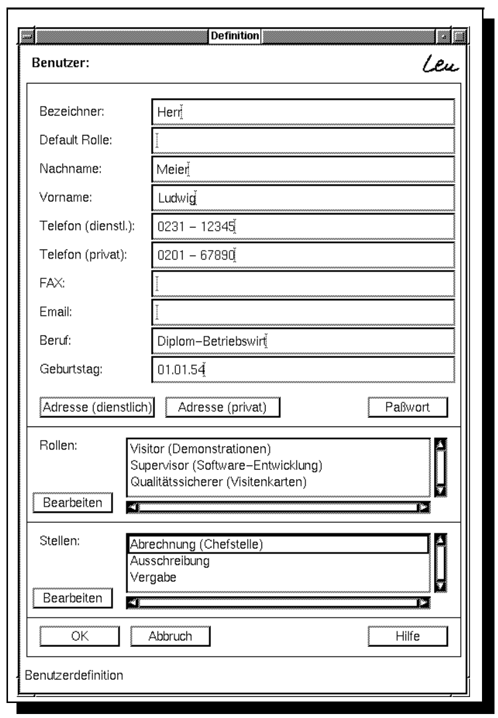


Abbildung 2.4: Dialogmaske zum Erfassen von Mitarbeitern

In der Benutzerverwaltung werden die Daten aller Mitarbeiter eines Unternehmens erfaßt. Abbildung 2.4 zeigt hierbei die Bildschirmmaske zum Erfassen eines neuen Mitarbeiters. In der Benutzerverwaltung können jedem Mitarbeiter, unabhängig von seinen Stellenzugehörigkeiten, Rollen zugewiesen werden.

2.3.4.2 Der Stellen-Editor

Im Stellen-Editor wird das Organigramm einer Unternehmung modelliert. Das Stellenmodell repräsentiert die Organisationsstrukturen, in die die Geschäftsprozesse eingebettet sind. Im Stellen-Editor können alle in die Ausführung der Geschäftsprozesse involvierten Stellen und ihre Beziehungen zueinander definiert werden. Das Stellenmodell verdeutlicht die organisatorischen Zusammenhänge, die den Geschäftsprozessen zugrunde liegen. Es repräsentiert Weisungsbefugnisse und trägt zum Verständnis von Kommunikationsstrukturen bei [SR92].

Die mit der Benutzerverwaltung erfaßten Personen können hier den Stellen zugeordnet werden. Zusätzlich bietet der Stellen-Editor durch die Zuordnung von Rollen zu Stellen (siehe Kap. 2.3.4.3) die Möglichkeit, den Mitarbeitern einer Stelle anhand ihres Aufgabenbereiches Berechtigungen zuzuweisen.

Abbildung 2.5 zeigt ein Beispiel eines Stellenmodells. In diesem Beispiel ist Herrn Meier die Leitung der gesamten Abteilung AVA (Ausschreibung-Vergabe-Abrechnung) übertragen. Die Stelle hat drei untergeordnete Stellen (Ausschreibung, Vergabe, Abrechnung). Die Stelle Abrechnung unterteilt sich wiederum in die Stellen Vorprüfung, Prüfung und Genehmigung. Die Leitung der mit N.N gekennzeichneten Stellen ist nicht besetzt.

Das Beispiel verdeutlicht die graphische Modellierung des Organigramms mit dem Stellen-Editor.

2.3.4.3 Der Rollen-Editor

Der Rollen-Editor dient der Definition von Benutzerberechtigungen. Es können werkzeugbasierte Berechtigungen, objektbasierte Berechtigungen und aktivitätsbasierte Berechtigungen vergeben werden. Innerhalb der werkzeugbasierten Berechtigungen werden die LEU-Werkzeuge festgelegt, die von den Benutzern innerhalb einer Rolle benutzt werden dürfen. Mit den objektbasierten Berechtigungen kann festgelegt werden, welche Objekte

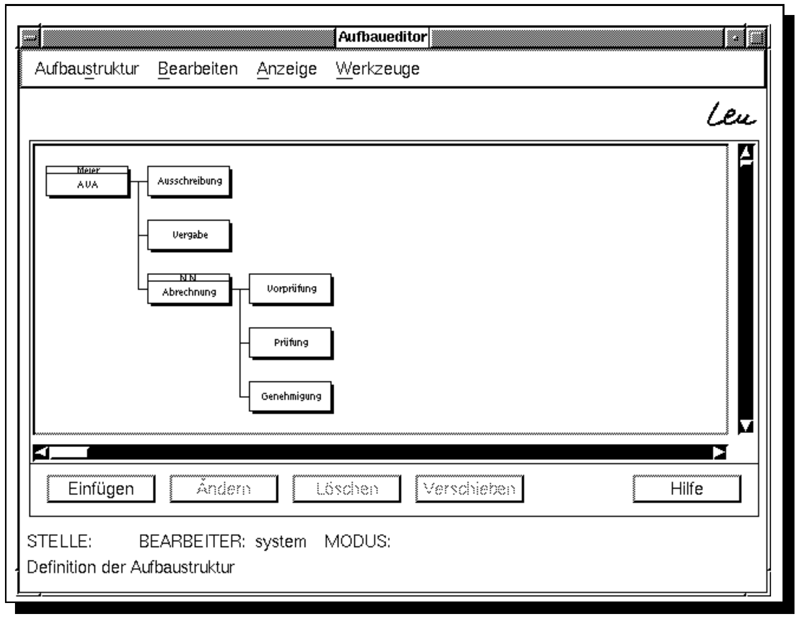


Abbildung 2.5: Beispiel für eine Aufbaustruktur eines Unternehmens

und Objekttypen die Benutzer innerhalb einer Rolle bearbeiten dürfen. Über die aktivitätsbasierten Berechtigungen werden die Aktivitäten innerhalb der Geschäftsprozesse mit einer Rolle verbunden. Der diese Rolle besitzende Benutzer darf dann alle Aktivitäten ausführen, für die er durch diese Rolle Berechtigungen besitzt. Rollen werden innerhalb eines Projektes definiert. Einem Benutzer werden die Berechtigungen über die Stellen, denen er zugeordnet ist, oder auch direkt, das heißt stellenunabhängig, in der Benutzerverwaltung zugeordnet. Die Summe der Berechtigungen eines Benutzers wird als seine Rolle im System bezeichnet. Ebenfalls rollenbezogene Systeme sind die Software-Prozeß-Managementsysteme MELMAC [DG90], [Gru91a], MERLIN [EJP+91] und IPSE2.5 [War89].

2.3.4.4 Die Projektverwaltung

Projekte bieten in LEU die Möglichkeit die Geschäftsprozesse einer $\xrightarrow{2.2}$LEU-Anwendung zusammenzufassen, die inhaltlich direkt zusammengehören. Bei der Aufteilung einer LEU-Anwendung in Projekte ist auf wohldefinierte und möglichst schmale Schnittstellen zwischen den Projekten zu achten. Zu jedem Projekt gehört in LEU ein eigenes Datenmodell. In der Projektverwaltung können neue Projekte angelegt und bestehende geändert oder gelöscht werden. Jedem Projekt kann hier ein Projektleiter zugeordnet werden.

2.3.5 Die Funktionsmodellierung

Mit dem Funktions-Editor können C-Funktionen erstellt, übersetzt und getestet werden. Diese Funktionen werden in Dialogen aufgerufen oder an die automatischen Instanzen der Geschäftsprozesse angebunden.

2.3.6 Die Anfragemodellierung

Der Anfrage-Editor bietet die Möglichkeit, Datenbankanfragen komfortabel zu formulieren, wobei eine kontextabhängige Eingabe für eine korrekte Syntax sorgt.

2.4 Die LEU-Simulation und -Analyse

Um zu gewährleisten, daß nur syntaktisch korrekte Ablaufmodelle ausgeführt werden, kann im Ablaufeditor ein Syntaxcheck durchgeführt werden. Verläuft dieser erfolgreich, so wird für das getestete Ablaufmodell ein entsprechendes Flag in der Datenbank gesetzt, das bei der Ausführung Berücksichtigung findet. Wird ein korrektes Ablaufmodell verändert, so wird das Flag zurückgesetzt, und der Syntaxcheck ist erneut durchzuführen.

Des weiteren steht zur Validation von Geschäftsprozeßmodellen ein FUNSOFT-Netzsimulator zur Verfügung. Er interpretiert die FUNSOFT-Netze und führt dabei die simulationsspezifischen Anteile der Aktivitätsbeschreibungen aus. Die Simulation eines Geschäftsprozesses wird mittels einer Animationskomponente graphisch visualisiert. Der Geschäftsprozeßmodellierer kann durch geeignete Entscheidungen während der Simulation (Konfliktlösung, etc.) Einfluß auf den Simulationsverlauf nehmen. Alle Simulationsereignisse werden mitprotokolliert und in einem Protokoll gespeichert. Auf der Basis dieses Protokolls arbeitet eine im LEU-Ablaufeditor integrierte Komponente, die die Simulationen statistisch auswertet. Sie gibt Aufschluß über die durchschnittliche Wartezeit von Objekten bestimmter Typen auf Weiterverarbeitung, über den Parallelitätsgrad des Geschäftsprozeßmodells, das heißt, wieviele Aktivitäten innerhalb des Geschäftsprozeßmodells gleichzeitig ausführbar sind und berechnet den kritischen Pfad des Geschäftsprozeßmodells. Weitere Informationen zur Simulation von FUNSOFT-Netzen können [Gru93b] und [DG91] entnommen werden.

Analysekomponenten befinden sich auch in der LEU-Datenmodellierung. Sie bieten den Modellierern Hilfen bei der Erstellung ihrer Datenmodelle [Sch94].

2.5 Die LEU-Ausführung

Die LEU-Ausführungskomponente führt die mit den LEU-Modellierungswerkzeugen erstellten Anwendungen aus.
Sie wurde als verteiltes System auf der Basis einer Client-Server-Architektur realisiert. Abbildung 2.6 zeigt alle an der Ausführungskomponente beteiligten LEU-Werkzeuge.

Der Datenaustausch zwischen den als eigenständige UNIX-Prozesse realisierten Werkzeugen erfolgt über eine Interprozeß-Kommunikation (im folgenden auch kurz als IPC bezeichnet) auf der Basis von *UNIX-Sockets* [Ste92]. Jedes Werkzeuge (in Abb. 2.6 durch den oval umrundeten Werkzeugnamen dargestellt) bildet einen eigenständigen Prozeß. Die Linien verbinden dabei die Werkzeuge, die miteinander kommunizieren. Die Beschriftung der Kommunikationspfade mit $n : m$-Beziehungen sagt aus, in welcher Anzahl die verschiedenen Prozesse miteinander verbunden sind. Beispielsweise kommuniziert in der Abbildung *ein* LEU-Controller mit n verschiedenen Agenda-Terminals. Wie in Abbildung 2.6 zu sehen (Beschriftung der Kommunikationspfade mit *LS* und *AS*) besteht die LEU-Ausführungskomponente aus zwei Arten von Controllerstrukturen.

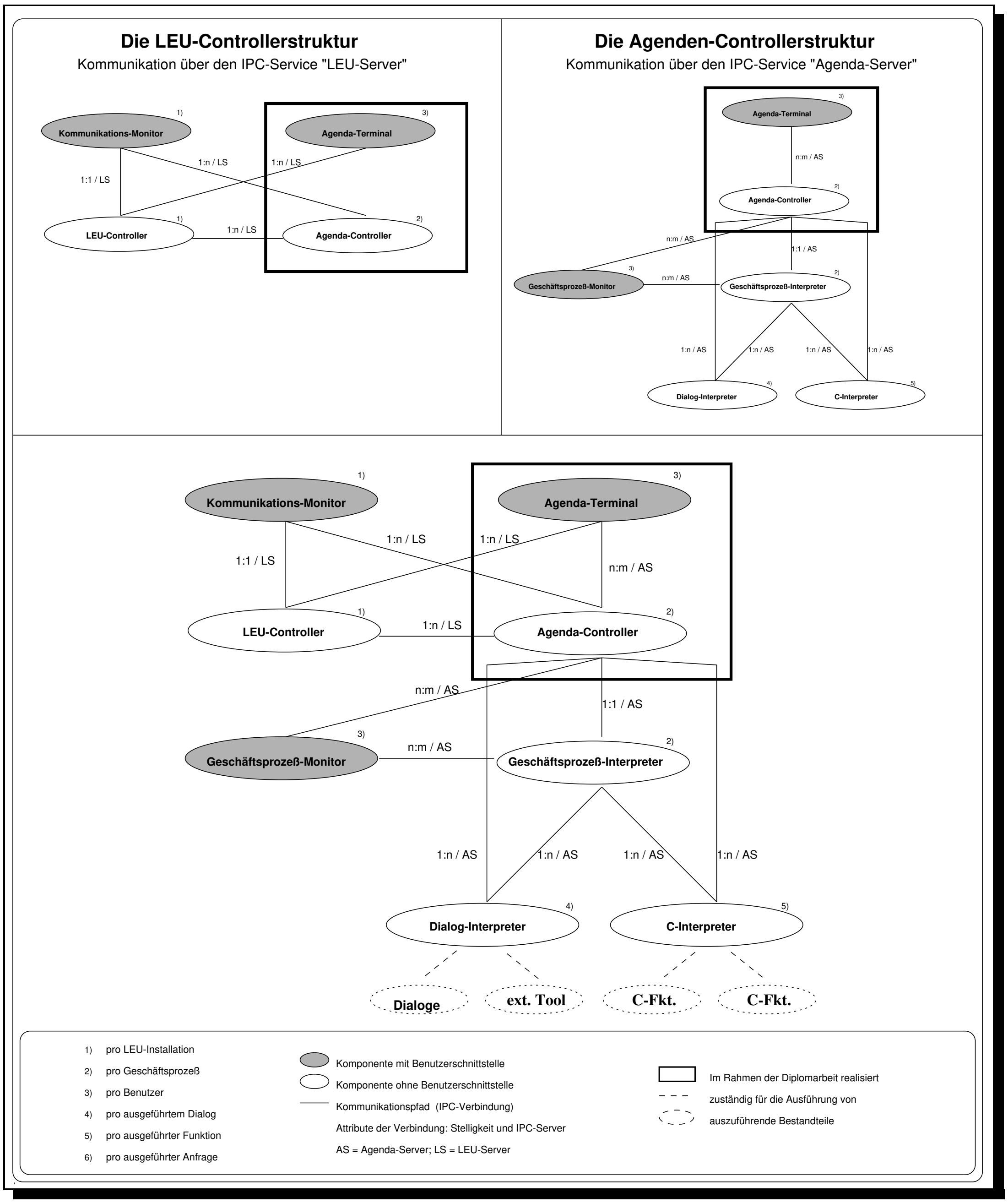


Abbildung 2.6: Aufbau der LEU-Ausführungswerkzeuge

Das heißt, innerhalb der LEU-Controllerstruktur kommuniziert der LEU-Controller mit dem Kommunikations-Monitor, den Agenden-Controllern und den Agenda-Terminals und der Kommunikations-Monitor mit den Agenda-Controllern.
Innerhalb der Agenda-Controllerstrukturen kommuniziert der Agenden-Controller mit den Agenda-Terminals, den Geschäftsprozeß-Monitoren, dem Geschäftsprozeß-Interpreter, den Dialog-Interpretern und den C-Interpretern. Ebenfalls innerhalb der Agenda-Controllerstrukturen kommuniziert der Geschäftsprozeß-Interpreter mit den Geschäftsprozeß-Monitoren, den Dialog-Interpretern und den C-Interpretern.
Innerhalb einer Controllerstruktur kommunizieren alle Clients über den gleichen IPC-Service. Die Struktur besteht aus allen Clients, die über den IPC-Service kommunizieren.

- **Die LEU-Controllerstruktur**
  Sie wird pro LEU-Installation nur einmal gestartet. Die IPC-Kommunikation innerhalb der LEU-Controllerstruktur läuft über den IPC-Service "LEU-Server". Der LEU-Controller (siehe Abschnitt 2.5.1) übernimmt die Controlleraufgaben dieser Struktur. Hierzu zählen der Auf- und Abbau der IPC-Kommunikation und die Verwaltung der bestehenden IPC-Strukturen. Der IPC-Service und der LEU-Controller werden schon bei der Installation von LEU, bzw. bei jedem Hochfahren der beteiligten Rechner gestartet. Innerhalb dieser Controllerstruktur kommunizieren der LEU-Controller, der Kommunikations-Monitor (siehe Abschnitt 2.5.7), die Agenden-Controller und die Agenda-Terminals miteinander.

- **Die Agenda-Controllerstrukturen**
  Pro Geschäftsprozeß wird in LEU eine neue Agenda-Controllerstruktur gestartet. Die IPC-Kommunikation innerhalb der Agenda-Controllerstrukturen läuft über den IPC-Service "Agenda-Server". Der Agenda-Server steuert die IPC-Kommunikation innerhalb dieser Struktur. Für jede neue Agenda-Controllerstruktur wird ein neuer Agenda-Server gestartet. Über diesen Service wird der gesamte bei der Ausführung des Geschäftsprozesses anfallende Datenaustausch zwi-

schen den einzelnen Clients der Struktur realisiert. Der Agenden-Controller übernimmt die Controlleraufgaben dieser Struktur. Innerhalb der Agenda-Controllerstruktur kommunizieren alle direkt an der Ausführung beteiligten Werkzeuge der LEU-Ausführungskomponente miteinander.

Bevor in Kapitel 3 und 4 die beiden innerhalb dieser Diplomarbeit realisierten LEU-Ausführungskomponenten beschrieben werden, wird in den folgenden Abschnitten kurz auf die Aufgaben und Funktionalitäten aller an der Ausführung beteiligten LEU-Komponenten eingegangen.

2.5.1 Der LEU-Controller

Der LEU-Controller kontrolliert und verwaltet die Interprozeßkommunikation der LEU-Ausführungswerkzeuge. Innerhalb dieser Aufgabe baut er beispielsweise die Verbindung zwischen den Terminals und den jeweiligen in Ausführung befindlichen Geschäftsprozessen auf und kontrolliert deren Start und Beendigung. Die Aufgaben des LEU-Controllers werden bei der Beschreibung des zwischen den einzelnen LEU-Ausführungsweerkzeugen bestehenden Nachrichtenflusses in Kapitel 3.2.2 deutlich.

2.5.2 Der Agenden-Controller

Der Agenden-Controller übermittelt die in den Geschäftsprozessen zur Bearbeitung anstehenden Aktivitäten unter Berücksichtigung der individuellen Benutzerberechtigungen an die Agenda-Terminals. Dieses Werkzeug wurde im Implementierungsteil dieser Arbeit entwickelt und in die Ausführungskomponente des Geschäftsprozeß-Management-Systems LEU integriert. Kapitel 3 beschreibt das Werkzeug und seine Entwicklung.

2.5.3 Das Agenda-Terminal

Das Agenda-Terminal bildete die Oberfläche des Geschäftsprozeß-Management-Systems LEU. Es wurde ebenfalls im Implementierungsteil dieser Arbeit realisiert und in die LEU-Ausführungskomponente integriert. Kapitel 4 und 5 beschreiben die Entwicklung dieses Werkzeugs.

2.5.4 Der Geschäftsprozeß-Interpreter

Der Geschäftsprozeß-Interpreter steuert die Ausführung eines Geschäftsprozesses. Pro Geschäftsprozeß wird hierzu ein solcher Interpreter gestartet.

Die manuellen Aktivitäten innerhalb der Geschäftsprozesse benötigen zu ihrer Abarbeitung die Interaktion von Benutzern. Diese Aktivitäten werden an den Agenden-Controller übermittelt und von ihm an die jeweiligen Terminals weitergeleitet. Des weiteren kommuniziert der Geschäftsprozeß-Interpreter mit den anderen, ihm untergeordneten Aktivitäts-Interpretern und weist ihnen die Aktivitäten zu, für deren Abarbeitung sie innerhalb des jeweiligen Geschäftsprozesses zuständig sind.

2.5.5 Der Dialog-Interpreter

Der Dialog-Interpreter verarbeitet die an die manuellen Instanzen angebundenen Dialoge. Die Dialoge bestimmen das Layout der Fenster, die im Rahmen der Ausführung von Aktivitäten auf den Bildschirmen der einzelnen Benutzer erscheinen und geben den Fenstern ihre Funktionalität. Über den Dialog-Interpreter kann auch beliebige Standard-Software in die Geschäftsprozesse eingebunden werden.

2.5.6 Der C-Interpreter

Der C-Interpreter übernimmt die Ausführung von C-Funktionen. C-Funktionen kommen im Rahmen von automatischen Aktivitäten zum Einsatz. Der C-Interpreter ruft hierbei die Funktionen auf und erhält Informationen über ihre Terminierung und die von ihnen bearbeiteten Objekte.

2.5.7 Der LEU-Kommunikationsmonitor

Der LEU-Kommunikationsmonitor ist ein administratives Werkzeug der LEU-Ausführungskomponente. Durch seine graphische Oberfläche visualisiert er die innerhalb der verschiedenen Controllerstrukturen bestehenden Kommunikationspfade zwischen allen innerhalb einer Controllerstruktur existierenden LEU-Ausführungswerkzeugen. Der Systemverwalter kann dieses Werkzeug beispielsweise dazu benutzen, Informationen über alle laufenden Geschäftsprozesse und nicht korrekt terminierten LEU-Werkzeuge zu erhalten. Abbildung 2.7 zeigt einen "Screendump" vom LEU-Kommu-

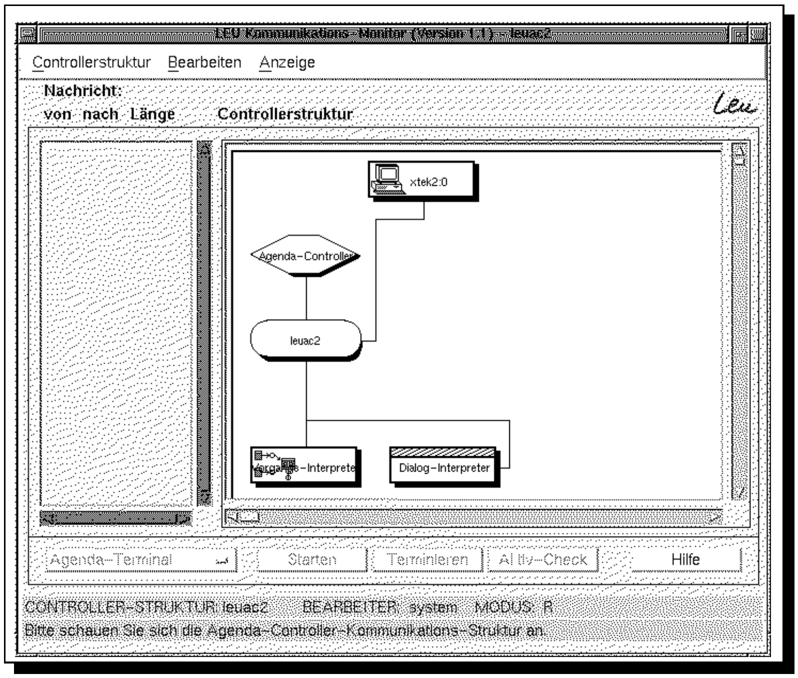


Abbildung 2.7: Screendump des LEU-Kommunikationsmonitors

nikationsmonitor. Seine Oberfläche visualisiert eine beispielhafte Agenda-Controllerstruktur. Das IPC-Service mit dem Namen *leuac2* ist oval umrundet. Mit diesem Service sind ein Agenda-Terminal (mit *xtek2* bezeichnet), der *Agenden-Controller*, der Geschäftsprozeß-Interpreter (*Vorgangs-Interpreter*) und ein *Dialog-Interpreter* verbunden. Durch das Anklicken der verschiedenen Symbole können zusätzliche Informationen zu den Clients erfragt werden.

Durch die Oberfläche dieses Werkzeugs können die innerhalb der verschiedenen Controllerstrukturen (*LEU-Controllerstruktur* und *Agenda-Controllerstrukturen*) bestehenden Kommunikationspfade angezeigt werden.

2.5.8 Der Geschäftsprozeß-Monitor

Durch seine graphische Oberfläche visualisiert der Geschäftsprozeß-Monitor den aktuellen Ausführungsstand der Geschäftsprozesse. Mit diesem Werkzeug kann sich jeder Mitarbeiter, der dazu berechtigt ist, über den Arbeitsfortschritt in den Geschäftsprozessen informieren.

Abbildung 2.8 zeigt einen "Screendump" vom Geschäftsprozeß-Monitor. In diesem Beispiel zeigt der Geschäftsprozeß-Monitor den Bearbeitungsstand vom Gechäftsprozeß *"Anzeigen Reparaturauftrag"* an. Mit diesem Prozeß können alle bearbeiteten und noch durchzuführenden Reparaturaufträge angezeigt werden. Die automatische Instanz *Start* liest alle Reparaturaufträge aus der Datenbank aus und schreibt sie in die Nachbereichskanäle. An der manuellen Instanz *Auswählen Auftrag* ist ein Dialog zum Auswählen eines Reparaturauftrags angebunden. Diese Instanz ist derzeit aktiviert (durch die schlechte Abbildungsqualität schwer zu erkennen). Nach dem Auswählen eines Reparaturauftrags wird dieser mit dem, an die Instanz *Anzeigen Auftrag* angebundenen Dialog dem Benutzer angezeigt.

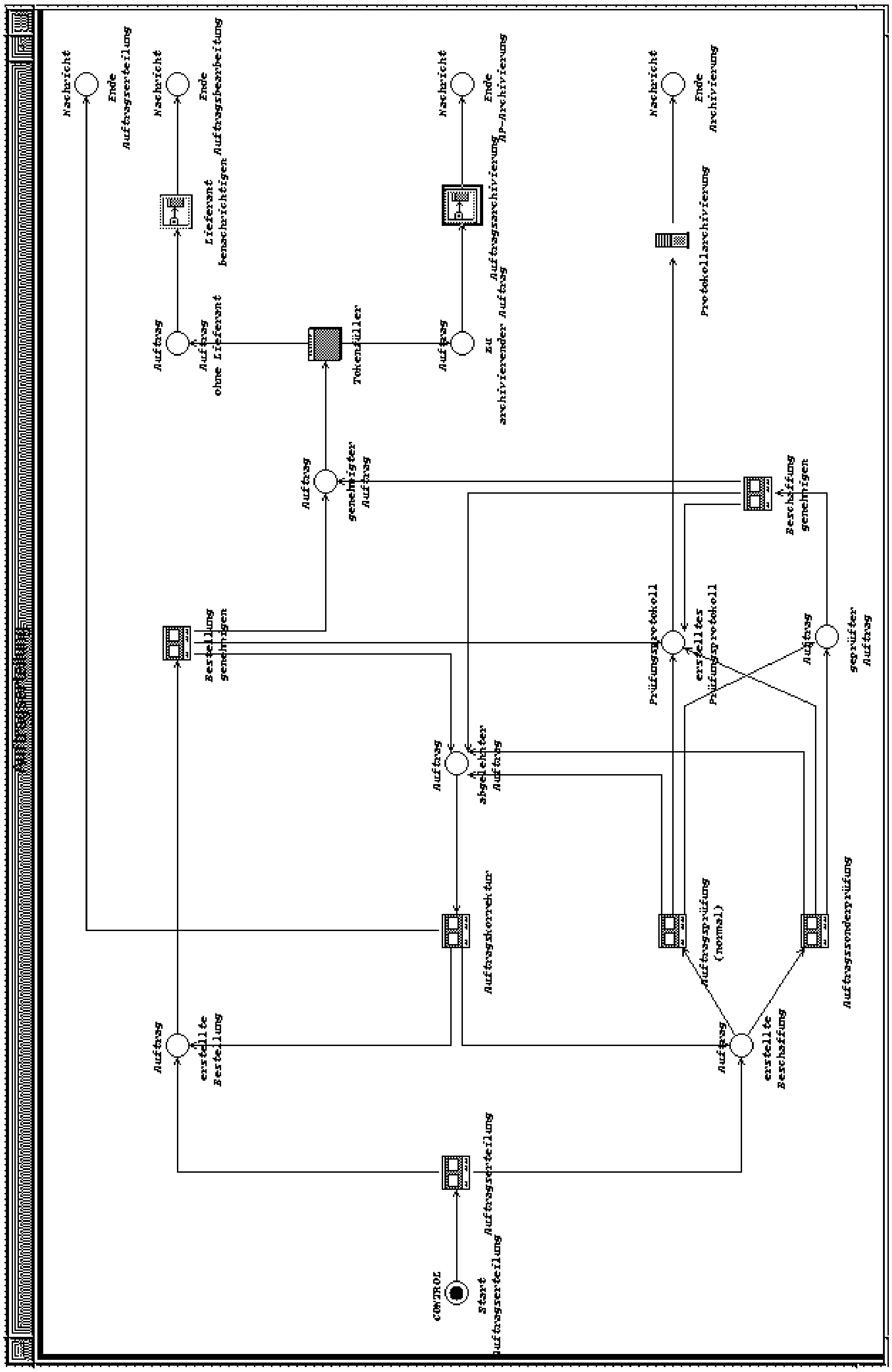


Abbildung 2.8: Screendump des LEU-Geschäftsprozeß-Monitors

Kapitel 3

Der Agenden-Controller

Der Agenden-Controller wurde während dieser Diplomarbeit entwickelt und in die Ausführungskomponente von LEU integriert. Der erstellte Programmcode wurde bei der Abgabe am Lehrstuhl Softwaretechnologie hinterlegt. In diesem Kapitel werden die Anforderungen an den Agenden-Controller und seine Realisierung beschrieben. Abschnitt 3.1 befaßt sich mit den Anforderungen an das Werkzeug. Hier werden zum einen seine Aufgaben bei der Berechtigungsüberprüfung und beim Füllen der Agenden und zum anderen seine Aufgaben als Controller einer Controllerstruktur erläutert. Des weiteren wird hier auf das Thema System-Sicherheit eingegangen. Unter System-Sicherheit wird hierbei das Abfangen unkontrollierter Terminierung von Clients innerhalb der LEU-Controllerstrukturen verstanden. Abschnitt 3.2 beschreibt die Realisierung des Agenden-Controllers innerhalb der LEU-Ausführungskomponente.

3.1 Anforderungen

3.1.1 Allgemeine Anforderungen

Alle Geschäftsprozesse bestehen aus einer Menge von manuellen und automatischen Aktivitäten, die miteinander in logischen und temporalen Abhängigkeiten stehen und in ihrer Gesamtheit die Prozesse ausmachen. Für die Abarbeitung der manuellen Aktivitäten ist die Interaktion von Personen notwendig. Diese Aktivitäten entsprechen den Aufgaben der Sachbearbeiter in den Unternehmen. Die manuellen Aktivitäten innerhalb der verschiedenen Geschäftsprozesse fallen in verschiedene Sach- und Aufgabengebiete und müssen von den Mitarbeitern der entsprechenden Abteilungen eines Unternehmens bearbeitet werden. Hierzu wurde noch während der Modellierungsphase das Organigramm des Unternehmens im System nachgebildet. Die zu den verschiedenen Stellen gehörenden Personen wurden im System erfaßt und den Abteilungen, denen sie angehören, zugewiesen. Des weiteren wurde das Berechtigungsprofil jeder Abteilung des Unternehmens mit Hilfe des Rollensystems (siehe Kapitel 2.3.4) modelliert. Zum Ende der Ablaufmodellierung der Geschäftsprozesse wurden alle manuellen Aktivitäten in Aufgabengebiete unterteilt und entsprechend den Abteilungen, von denen sie zu bearbeiten sind, mit Rollen versehen. Auf diese Weise lassen sich zur Ausführungszeit der Geschäftsprozesse, die in ihnen anstehenden Aufgaben den Abteilungen und Sachbearbeitern zuordnen, die für ihre Bearbeitung zuständig sind.

Die Hauptaufgabe des Agenden-Controllers besteht nun darin, jedem Sachbearbeiter alle in den Geschäftsprozessen anstehenden Aufgaben, die zu seinem Sachgebiet gehören und zu deren Bearbeitung er berechtigt ist, in seine Agenda zu schreiben. Da die Anwendungsbenutzer nicht immer mit ihrer maximalen Rolle, also mit ihrer maximalen Berechtigung arbeiten (die Anwendungsbenutzer können sich z.B. auch gezielt mit nur einigen Rollen, die sie besitzen in die Ausführung der Geschäftsprozesse einloggen), muß der Agenden-Controller jederzeit über das aktuelle Berechtigungsprofil jedes Benutzers informiert sein. Werden dem Agenden-Controller vom $\xrightarrow{1.2}$Geschäftsprozeß-Interpreter des Geschäftsprozesses neue Aktivitäten übermittelt, oder nimmt ein neuer Sachbearbeiter an der Bearbeitung des Geschäftsprozesses teil, so vergleicht der Agenden-Controller das Berechtigungsprofil der Aktivitäten mit dem aktuellen Berechtigungsprofil der Sachbearbeiter und übermittelt die Aktivitäten an die Agenden der berechtigten Anwendungsbenutzer.

Weiter Anforderungen an den Agenden-Controller bestehen darin, die von den Benutzern ausgewählten Aktivitäten dem Geschäftsprozeß-Interpreter des Geschäftsprozesses zu übermitteln und die Agenden aller am Geschäftsprozeß beteiligten Personen immer auf dem aktuellen Stand zu halten. Wird beispielsweise eine Aktivität durch einen Benutzer zur Bearbeitung ausgewählt, oder ist eine Aktivität durch eine Konfliktlösung (siehe $\xrightarrow{2.2}$Konflikte in FUNSOFT-Netzen) nicht mehr aktivierbar, so muß sie aus den Agenden aller Sachbearbeiter gelöscht werden, die sie vorher in ihren Agenden stehen hatten.

3.1.2 Controlleraufgaben einer Client-Server-Struktur

Für jeden Geschäftsprozeß wird eine neue Agenda-Controllerstruktur aufgebaut. Der Agenden-Controller übernimmt die Controlleraufgaben dieser Struktur. Zu den Controlleraufgaben zählen das Starten des IPC-Service, die Verwaltung der Kommunikationsstrukturen und der Auf- bzw. Abbau der Kommunikationspfade (siehe auch Abb. 2.6 auf Seite 25).

3.1.3 System-Sicherheit

Ein wichtiges Kriterium und Qualitätsmerkmal bei der Ausführung der Geschäftsprozesse bildet die System-Sicherheit. Stürzt während der Ausführung eines Geschäftsprozesses eine Komponente der Agenda-Controllerstruktur (siehe Abb. 2.6 auf Seite 25) ab, steht unter Umständen der gesamte Geschäftsprozeß. Um diese Gefahr zu beseitigen, muß in der LEU-Ausführungskomponente ein Konzept realisiert werden, bei dem alle Werkzeuge nach einer unkontrollierten Terminierung automatisch wieder hochgefahren werden.

3.2 Grobentwurf und Realisierung

3.2.1 Grundlegende Entwurfsentscheidungen

Dieser Abschnitt befaßt sich mit einigen grundlegenden in der Entwurfsphase des Agenden-Controllers getroffenen Entwurfsentscheidungen.

- **Ein Agenden-Controller pro auszuführendem Geschäftsprozeß**
  Da an der Ausführung eines Geschäftsprozesses unter Umständen eine Vielzahl von Sachbearbeitern teilnehmen, wird für jeden neuen Geschäftsprozeß ein weiterer Agenden-Controller gestartet. Auf diese Weise kann eine schnelle Verarbeitung der IPC-Messages garantiert und somit ein Engpaß bei der Geschäftsprozeß-Ausführung vermieden werden.

- **Agenden-Controller als Controller der Agenden-Controllerstrukturen**
  Um den Geschäftsprozeß-Interpreter nicht mit zusätzlicher Funktionalität zu belasten, werden die Controlleraufgaben innerhalb der Agenda-Controllerstruktur vom Agenden-Controller übernommen. Die Controlleraufgaben wurden schon im Abschnitt 3.1.2 beschrieben.

- **Ein Konzept für die "Systemsicherheit"**
  Wie schon in Abschnitt 3.1.3 erläutert, muß gewährleistet sein, daß die *unkontrollierte Terminierung* (Absturz) aller an der Ausführung eines Geschäftsprozesses beteiligten LEU-Werkzeuge aufgefangen wird, da ansonsten unter Umständen die Ausführung des gesamten Geschäftsprozesses steht. Bei der Erstellung und Realisierung des Konzepts für die "Systemsicherheit" werden die vom IPC-Service verschickten Controller-Nachrichten genutzt. Stürzt ein Werkzeuge innerhalb einer Controllerstruktur ab, so wird vom IPC-Service die Controller-Nachricht *CLIENT_CRASHED* an alle anderen Clients der Controllerstruktur verschickt. Aus dieser Nachricht läßt sich genau ermitteln, welches Werkzeug abgestürzt ist. Auf diese Weise erkennt der LEU-Controller die unkontrollierte Terminierung eines Agenden-Controllers und startet ihn erneut. Der Absturz eines Geschäftsprozeß-Interpreters wird vom Agenden-Controller aufgefangen. Unkontrolliert terminierte Dialog- bzw. C-Interpreter werden vom Geschäftsprozeß-Interpreter erneut gestartet. Das genaue Vorgehen wird am Ende des Abschnitts 3.2.2 beschrieben.

- **Initialer Aufbau der Agenda-Controllerstrukturen**
  Ist in einem Agenda-Terminal ein weiterer Geschäftsprozeß zum Starten ausgewählt worden, so wird für diesen Geschäftsprozeß eine weitere Agenda-Controllerstruktur gestartet. Im Konzept zum initialen Aufbau einer neuen Strukur wurde hierbei festgelegt, daß zunächst vom LEU-Controller ein neuer Agenden-Controller gestartet wird. Der Agenden-Controller startet daraufhin den Geschäftsprozeß-Interpreter und baut die IPC-Kommunikation zu allen Agenda-Terminals auf. Die Dialog- und C-Interpreter werden bei Bedarf (eine Aktivität muß ausgeführt werden) vom Geschäftsprozeß-Interpreter gestartet. Das Starten eines neuen Geschäftsprozesses und die hierfür zwischen den LEU-Werkzeugen ausgetauschten IPC-Nachrichten werden im Abschnitt 3.2.2 unter →*Starten eines neuen Geschäftsprozesses* ausführlich beschrieben.

3.2.2 Einbettung in die LEU-Ausführungskomponente

Der Agenden-Controller wurde im Rahmen dieser Diplomarbeit implementiert und in die Ausführungskomponente des Geschäftsprozeß-Managementsystems LEU integriert. Wie in Abbildung 2.6 auf Seite 25 zu sehen, ist der Agenden-Controller sowohl Client in der LEU-Controllerstruktur als auch in der Agenda-Controllerstruktur, deren Controlleraufgaben (siehe Abschnitt 3.1.2) er übernimmt. Der Datenaustausch zwischen den Clients erfolgt über eine Interprozeßkommunikation auf der Basis von *UNIX-Sockets*.

Innerhalb der LEU-Controllerstruktur kommuniziert der Agenden-Controller über den LEU-Server (in Abbildung 2.6 auf Seite 25 an der Beschriftung des Kommunikationspfades mit "LS" zu erkennen) mit dem LEU-Controller und dem LEU-Kommunikationsmonitor. Der Agenden-Controller wird vom LEU-Controller gestartet und teilt ihm den Start und die Beendigung des Geschäftsprozesses mit. Mit den anderen Clients innerhalb der LEU-Ausführungskomponente kommuniziert er über den Agenda-Server. Im folgenden werden die zum Datenaustausch zwischen dem Agenden-Controller und den anderen Clients der LEU-Ausführungswerkzeuge spezifizierten Nachrichten kurz beschrieben. Es werden hierbei nur die "NORMAL-MESSAGES" beschrieben, die zum Datenaustausch zwischen den Prozessen dienen. Auf die "CONTROLL-" und "SYSTEM-MESSAGES", die den Auf- und Abbau der Kommunikationsstrukturen innerhalb der Controllerstrukturen realisieren, soll hier nicht weiter eingegangen werden. Für den Aufbau der "NORMAL-MESSAGES" wurde mit Hilfe der erweiterten BNF (EBNF) ein Protokoll spezifiziert (siehe Anhang B).

Im nachfolgenden Abschnitt werden zunächst alle "NORMAL-MESSAGES" kurz beschrieben. Hierbei werden zuerst die Nachrichten, die zum Datenaustausch zwischen dem Agenden-Controller und dem LEU-Controller dienen, dann die Nachrichten, die zwischen dem Agenden-Controller und dem Geschäftsprozeß-Interpreter ausgetauscht werden müssen, und schließlich die Nachrichten, über die der Agenden-Controller mit den Agenda-Terminals und den anderen Clients der Agenda-Controllerstruktur (Kommunikationsmonitor, Dialog-Interpreter, C-Interpreter und Geschäftsprozeß-Monitor) kommuniziert, beschrieben. In diesem Zusammenhang wird erklärt, welchem Zweck die einzelnen Nachrichten dienen. Im Anschluß an diesen Teil wird der Nachrichtenfluß, den die verschiedenen Ereignisse bei der Ausführung der Geschäftsprozesse auslösen, beschrieben.

Nachrichtenaustausch mit dem LEU-Controller

- **KSB_LC_ACB_START_INC** (LEU-Contr. → Agenden-Contr.)
  Aus dieser Nachricht erfährt der Agenden-Controller, welcher Geschäftsprozeß gestartet werden soll.
- **KSB_LC_ACB_RESTART_INC** (LEU-Contr. → Agenden-Contr.)
  Diese Nachricht teilt dem Agenden-Controller mit, daß ein angehaltener Geschäftsprozeß nun weiter ausgeführt werden soll.
- **KSB_LC_ACB_RESTART_ACB** (LEU-Contr. → Agenden-Contr.)
  Diese Nachricht wird dem Agenden-Controller, nachdem er nach einem Absturz vom LEU-Controller erneut gestartet worden ist, übermittelt.
- **KSB_LC_ACB_STOP_INC** (LEU-Contr. → Agenden-Contr.)
  Diese Nachricht teilt dem Agenden-Controller mit, daß die Ausführung des

Geschäftsprozesses gestoppt und der Geschäftsprozeß aus der Datenbank gelöscht werden soll.

- **KSB_LC_ACB_SUSPEND_INC** (LEU-Contr. → Agenden-Contr.)
  Die Nachricht teilt dem Agenden-Controller mit, daß die Ausführung des Geschäftsprozesses angehalten werden soll.

- **KSB_LC_ACB_SYSTEM_CHANNEL** (LEU-Contr. → Agenden-Contr.)
  Von einem anderen Geschäftsprozeß wurde ein $\xrightarrow{2.2}$Systemkanal gefüllt. Die Nachricht ist an den Geschäftsprozeß-Interpreter weiterzuleiten, damit dieser, falls der Systemkanal auch im aktuellen Geschäftsprozeß benutzt wird, auf dieses Ereignis reagieren kann.

- **KSB_ACB_LC_LEUAC_HELLO** (Agenden-Contr. → LEU-Contr.)
  Mit dieser Nachricht meldet sich der gestartete Agenden-Controller beim LEU-Controller an.

- **KSB_ACB_LC_INC_STARTED** (Agenden-Contr. → LEU-Contr.)
  Die Nachricht teilt dem LEU-Controller den erfolgreichen Start des Geschäftsprozesses mit.

- **KSB_ACB_LC_END_OF_INC** (Agenden-Contr. → LEU-Contr.)
  Die Nachricht teilt dem LEU-Controller die Beendigung des Geschäftsprozesses mit.

- **KSB_ACB_LC_INC_SUSPENDED** (Agenden-Contr. → LEU-Contr.)
  Die Nachricht teilt dem LEU-Controller mit, daß der Geschäftsprozeß angehalten worden ist.

- **KSB_ACB_LC_INC_STOPPED** (Agenden-Contr. → LEU-Contr.)
  Die Nachricht teilt dem LEU-Controller mit, daß der Geschäftsprozeß angehalten und aus der Datenbank gelöscht worden ist.

- **KSB_ACB_LC_SYSTEM_CHANNEL** (Agenden-Contr. → LEU-Contr.)
  Im eigenen Geschäftsprozeß ist ein $\xrightarrow{2.2}$Systemkanal mit einem Objekt gefüllt worden. Die Nachricht wird an den LEU-Controller übermittelt und von ihm an alle Agenden-Controller weitergeleitet.

- **KSB_ACB_LC_START** (Agenden-Contr. → LEU-Contr.)
  Mit dieser Nachricht sendet der Agenden-Controller dem LEU-Controller Informationen, die der LEU-Controller an den LEU-Kommunikationsmonitor weiterreicht.

Nachrichtenaustausch mit dem Geschäftsprozeß-Interpreter

- **KSB_VI_ACB_INIT** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht teilt der Geschäftsprozeß-Interpreter dem Agenden-Controller mit, daß er gestartet worden ist und sich korrekt initialisiert hat. Der Agenden-Controller übermittelt ihm danach die Nachricht ”KSB_ACB_VI_START_NEW_PROCESS” bzw. ”KSB_ACB_VI_CONTINUE_PROCESS”.

- **KSB_VI_ACB_PROCESS_STARTED** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht teilt der Geschäftsprozeß-Interpreter dem Agenden-Controller mit, daß er das Geschäftsprozeßmodell geladen und dazu einen neuen Geschäftsprozeß in der Datenbank angelegt hat. Nach der Übermittlung dieser Nachricht sendet er alle aktivierbaren manuellen Aktivitäten an den Agenden-Controller.

- **KSB_VI_ACB_PRED_AGENCY** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht übermittelt der Geschäftsprozeß-Interpreter dem Agenden-Controller entweder eine neue, prädikatbehaftete Aktivität ($\xrightarrow{2.2}$Instanzen) oder eine prädikatbehaftete Aktivität, deren Vorbereich ($\xrightarrow{2.2}$Instanzen) sich verändert hat und deswegen in den Agenden geändert werden muß.

- **KSB_VI_ACB_AGENCY** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht übermittelt der Geschäftsprozeß-Interpreter dem Agenden-Controller entweder eine neue, nicht prädikatbehaftete Aktivität ($\xrightarrow{2.2}$Instanzen) oder eine nicht prädikatbehaftete Aktivität, deren Vorbereich ($\xrightarrow{2.2}$Instanzen) sich verändert hat und deswegen in den Agenden geändert werden muß.

- **KSB_VI_ACB_AGENCY_DONE** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht benachrichtigt der Geschäftsprozeß-Interpreter den Agenden-Controller über die Beendigung einer manuellen Aktivität.

- **KSB_VI_ACB_ACTIVATION_FAILED** (G.-Int. → Agenden-Contr.)
  Wurde von einem Anwendungsbenutzer eine Aktivität zur Bearbeitung ausgewählt und über den Agenden-Controller dem Geschäftsprozeß-Interpreter übermittelt, so prüft dieser, ob die Aktivität noch immer aktivierbar ist. Ist die Aktivität nicht mehr aktivierbar, wird diese Nachricht an den Agenden-Controller übermittelt.

- **KSB_VI_ACB_DELETE_AGENCY** (G.-Int. → Agenden-Contr.)
  Sind manuelle Aktivitäten, zum Beispiel durch Konfliktlösung ($\xrightarrow{2.2}$Konflikte in FUNSOFT-Netzen), nicht mehr aktivierbar, werden diese dem Agenden-Controller mit dieser Nachricht übermittelt.

- **KSB_VI_ACB_DEAD_NET** (G.-Int. → Agenden-Contr.)
  Die Interpretation des dem Geschäftsprozeß zugrunde liegenden FUNSOFT-Netzes hat ergeben, daß sich keine Aktivität mehr in Bearbeitung befindet und keine neue Aktivität mehr aktivierbar ist (*Deadlock* in Petri-Netzen). Somit ist

der Geschäftsprozeß beendet. Dieser Sachverhalt wird dem Agenden-Controller über diese Nachricht mitgeteilt.

- **KSB_VI_ACB_SHUTDOWN_EXECUTED** (G.-Int. → Agenden-Contr.)
  Diese Nachricht wird vom Geschäftsprozeß-Interpreter dem Agenden-Controller übermittelt, nachdem der Geschäftsprozeß gestoppt und aus der Datenbank gelöscht worden ist.

- **KSB_VI_ACB_PROCESS_CONTINUED** (G.-Int. → Agenden-Contr.)
  Diese Nachricht wird vom Geschäftsprozeß-Interpreter dem Agenden-Controller übermittelt, nachdem ein vorher angehaltener Geschäftsprozeß erneut zur Ausführung gebracht worden ist.

- **KSB_VI_ACB_PROCESS_STOPPED** (G.-Int. → Agenden-Contr.)
  Diese Nachricht wird vom Geschäftsprozeß-Interpreter dem Agenden-Controller übermittelt, nachdem der Geschäftsprozeß angehalten worden ist.

- **KSB_VI_ACB_SYSTEM_CHANNEL** (G.-Int. → Agenden-Contr.)
  Diese Nachricht wird vom Geschäftsprozeß-Interpreter an den Agenden-Controller gesendet. Innerhalb des Geschäftsprozesses wurde ein $\xrightarrow{2.2}$Systemkanal gefüllt. Der Systemkanal muß über den LEU-Controller an alle Agenden-Controller der anderen Geschäftsprozesse übermittelt werden.

- **KSB_VI_ACB_START** (G.-Int. → Agenden-Contr.)
  Mit dieser Nachricht sendet der Geschäftsprozeß-Interpreter dem Agenden-Controller Informationen, die er an den LEU-Kommunikationsmonitor weiterreicht.

- **KSB_I_ACB_CLIENT_PING** (Agenden-Contr. → G.-Int.)
  Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" (Mit einem *"Aktiv-Check"* kann vom LEU-Kommunikationsmonitor aus jeder Clients innerhalb der LEU-Ausführung daraufhin überprüft werden, ob er noch korrekt arbeitet oder ob er sich "aufgehängt" hat.) auf den Geschäftsprozeß-Interpreter durchgeführt, so leitet der Agenden-Controller diese Nachricht an den Geschäftsprozeß-Interpreter weiter.

- **KSB_CLIENT_I_ACB_PONG** (Agenden-Contr. → G.-Int.)
  Diese Nachricht schickt der Geschäftsprozeß-Interpreter bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.

- **KSB_ACB_VI_START_AGENCY** (Agenden-Contr. → G.-Int.)
  Hat ein Anwendungsbenutzer eine nicht prädikatbehaftete Aktivität zur Bearbeitung ausgewählt, so wird sie dem Geschäftsprozeß-Interpreter mit dieser Nachricht übermittelt.

- **KSB_ACB_VI_START_PRED_AGENCY** (Agenden-Contr. → G.-Int.)
  Hat ein Anwendungsbenutzer eine prädikatbehaftete Aktivität zur Bearbeitung

ausgewählt, so wird sie dem Geschäftsprozeß-Interpreter mit dieser Nachricht übermittelt.

- **KSB_ACB_VI_RESET_AGENCY** (Agenden-Contr. → G.-Int.)
  Ein Anwendungsbenutzer möchte eine gestartete Aktivität nicht weiter bearbeiten und zurücksetzen. Mit dieser Nachricht wird die Aktivität vom Agenden-Controller an den Geschäftsprozeß-Interpreter weitergeleitet.
- **KSB_ACB_VI_SHUTDOWN** (Agenden-Contr. → G.-Int.)
  Diese Nachricht teilt dem Geschäftsprozeß-Interpreter mit, daß der Geschäftsprozeß gestoppt und aus der Datenbank gelöscht werden soll.
- **KSB_ACB_VI_STOP_PROCESS** (Agenden-Contr. → G.-Int.)
  Diese Nachricht teilt dem Geschäftsprozeß-Interpreter mit, daß der Geschäftsprozeß angehalten werden soll.
- **KSB_ACB_VI_CONTINUE_PROCESS** (Agenden-Contr. → G.-Int.)
  Mit dieser Nachricht wird dem Geschäftsprozeß-Interpreter vom Agenden-Controller mitgeteilt, daß ein vorher angehaltener Geschäftsprozeß erneut gestartet werden soll.
- **KSB_ACB_VI_START_NEW_PROCESS** (Agenden-Contr. → G.-Int.)
  Mit dieser Nachricht wird dem gerade gestarteten Geschäftsprozeß-Interpreter vom Agenden-Controller das Geschäftsprozeßmodell mitgeteilt, zu dem ein neuer Geschäftsprozeß gestartet werden soll.
- **KSB_ACB_VI_ACB_RESTART** (Agenden-Contr. → G.-Int.)
  Diese Nachricht sendet der nach einem Absturz vom LEU-Controller erneut gestartete Agenden-Controller dem Geschäftsprozeß-Interpreter, damit dieser ihm alle aktivierbaren manuellen Aktivitäten übermittelt.

Nachrichtenaustausch mit dem Agenda-Terminal

- **KSB_TI_ACB_START_AGENCY** (Terminal → Agenden-Contr.)
  Eine im Agenda-Terminal zur Bearbeitung ausgewählte nicht prädikatbehaftete Aktivität (siehe $\xrightarrow{2.2}$Instanzen) wird mit dieser Nachricht dem Agenden-Controller übermittelt.
- **KSB_TI_ACB_START_PRED_AGENCY** (Terminal → Agenden-Contr.)
  Eine im Agenda-Terminal zur Bearbeitung ausgewählte prädikatbehaftete Aktivität (siehe $\xrightarrow{2.2}$Instanzen) wird mit dieser Nachricht dem Agenden-Controller übermittelt.
- **KSB_TI_ACB_RESET_AGENCY** (Terminal → Agenden-Contr.)
  Eine im Agenda-Terminal zum Zurücksetzen ausgewählte Aktivität wird mit dieser Nachricht dem Agenden-Controller übermittelt.
- **KSB_TI_ACB_NEW_USER_LOGIN** (Terminal → Agenden-Contr.)
  Mit dieser Nachricht loggt sich ein neuer Benutzer in den Geschäftsprozeß ein.

- **KSB_TI_ACB_NEW_PROCESS_LOGIN** (Terminal → Agenden-Contr.)
Ein neuer Geschäftsprozeß ist gestartet worden. Mit dieser Nachricht melden sich alle Agenda-Terminals bei dem neuen Geschäftsprozeß an.
- **KSB_TI_ACB_END_TERM** (Terminal → Agenden-Contr.)
Mit dieser Nachricht meldet sich ein Agenda-Terminal beim Agenden-Controller ab.
- **KSB_TI_ACB_SHUTDOWN** (Terminal → Agenden-Contr.)
Wird im Agenda-Terminal ein Geschäftprozeß zum *Beenden* ausgewählt, so wird dem zu diesem Geschäftsprozeß gehörenden Agenden-Controller diese Nachricht übermittelt.
- **KSB_TI_ACB_STOP_PROCESS** (Terminal → Agenden-Contr.)
Wird im Agenda-Terminal ein Geschäftprozeß zum *Anhalten* ausgewählt, so wird dem zu diesem Geschäftsprozeß gehörenden Agenden-Controller diese Nachricht übermittelt.
- **KSB_TI_LACB_START** (Terminal → Agenden-Contr.)
Mit dieser Nachricht sendet ein Agenda-Terminal dem Agenden-Controller Informationen, die er an den LEU-Kommunikationsmonitor weiterreicht.
- **KSB_CLIENT_LACB_PONG** (Terminal → Agenden-Contr.)
Diese Nachricht schickt das Agenda-Terminal bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.
- **KSB_LACB_CLIENT_PING** (Agenden-Contr. → Terminal)
Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf ein Agenda-Terminal durchgeführt, so leitet der Agenden-Controller diese Nachricht an das zu überprüfende Agenda-Terminal weiter. Das Agenda-Terminal antwortet mit der Nachricht KSB_CLIENT_LACB_PONG.
- **KSB_ACB_TI_LOGIN_ANSWER** (Agenden-Contr. → Terminal)
Nachdem sich ein neuer Benutzer mit seinem Berechtigungsprofil beim Agenden-Controller angemeldet hat, überprüft dieser, ob der Benutzer zur Mitarbeit im Geschäftsprozeß berechtigt ist und teilt ihm das Ergebnis mit dieser Nachricht mit.
- **KSB_ACB_TI_AGENCY** (Agenden-Contr. → Terminal)
Mit dieser Nachricht übermittelt der Agenden-Controller dem Agenda-Terminal eine aktivierbare, nicht prädikatbehaftete Aktivität ($\xrightarrow{2.2}$Instanzen).
- **KSB_ACB_TI_PRED_AGENCY** (Agenden-Contr. → Terminal)
Mit dieser Nachricht übermittelt der Agenden-Controller dem Agenda-Terminal eine aktivierbare, prädikatbehaftete Aktivität ($\xrightarrow{2.2}$Instanzen).
- **KSB_ACB_TI_DELETE_AGENCY** (Agenden-Contr. → Terminal)
Mit dieser Nachricht übermittelt der Agenden-Controller dem Agenda-Terminal

eine nicht mehr aktivierbare Aktivität. Sie muß aus der Datenhaltung des Agenda-Terminals gelöscht werden.

- **KSB_ACB_TI_AGENCY_DONE** (Agenden-Contr. → Terminal)
  Mit dieser Nachricht unterrichtet der Agenden-Controller ein Agenda-Terminal über die Beendigung einer vom Benutzer bearbeiteten Aktivität. Sie kann aus der Datenhaltung gelöscht werden.

- **KSB_ACB_TI_AGENCY_ALREADY_WORKING** (Agenden-Contr. → Terminal)
  Ein Benutzer hat eine Aktivität zur Bearbeitung dem Agenden-Controller übermittelt. Die Aktivität wurde jedoch schon kurz vorher von einem anderen Benutzer zur Bearbeitung ausgewählt, oder kann aus anderen Gründen zur Zeit nicht aktiviert werden. Dieser Sachverhalt wird vom Agenden-Controller dem Agenda-Terminal mit dieser Nachricht übermittelt.

- **KSB_ACB_TI_NEW_INC** (Agenden-Contr. → Terminal)
  Das Starten eines neuen Geschäftsprozesses ist komplett abgeschlossen. Die Agenda-Terminals werden nun mit dieser Nachricht vom Agenden-Controller über den neuen Geschäftsprozeß informiert.

- **KSB_ACB_TI_END_OF_INC** (Agenden-Contr. → Terminal)
  Mit dieser Nachricht unterrichtet der Agenden-Controller die Agenda-Terminals über die Beendigung des Prozesses.

Nachrichtenaustausch mit dem Kommunikationsmonitor

- **KSB_CM_ACB_START** (Kom.-Monitor → Agenden-Contr.)
  Mit dieser Nachricht meldet sich der LEU-Kommunikationsmonitor als neuer Client beim Agenden-Controller an. Der Agenden-Controller übermittelt ihm danach mit der Nachricht KSB_ACB_CM_CLIENT_LIST Informationen zu allen Clients der Agenda-Controllerstruktur.

- **KSB_CM_LACB_PING** (Kom.-Monitor → Agenden-Contr.)
  Wenn im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf einen Client durchgeführt wird, so werden mit dieser Nachricht Informationen zum Client, der getestet werden soll, übermittelt. An den zu überprüfenden Client wird danach die Nachricht KSB_LACB_CLIENT_PING übermittelt.

- **KSB_LACB_CM_CLIENT_HELLO** (Agenden-Contr. → Kom.-Monitor)
  Meldet sich ein neuer Client an der Agenda-Controllerstruktur an, so wird überprüft, ob der LEU-Kommunikationsmonitor als Client an der Agenda-Controllerstruktur angemeldet ist. Wenn ja, so werden ihm mit dieser Nachricht Informationen über den neuen Client übermittelt.

- **KSB_LACB_CM_CLIENT_BY** (Agenden-Contr. → Kom.-Monitor)
  Meldet sich ein neuer Client an der Agenda-Controllerstruktur ab, so wird

überprüft, ob der LEU-Kommunikationsmonitor als Client an der Agenda-Controllerstruktur angemeldet ist. Wenn ja, so werden ihm mit dieser Nachricht Informationen über den Client, der sich abgemeldet hat, übermittelt.

- **KSB_LACB_CM_CLIENT_CRASH** (Agenden-Contr. → Kom.-Monitor)
  Stürzt ein Client innerhalb der Agenda-Controllerstruktur ab, so wird überprüft, ob der LEU-Kommunikationsmonitor als Client an der Agenda-Controllerstruktur angemeldet ist. Wenn ja, so werden ihm mit dieser Nachricht Informationen über den abgestürzten Client übermittelt.

- **KSB_LACB_CM_PONG** (Agenden-Contr. → Kom.-Monitor)
  Hat ein Client innerhalb der Agenda-Controllerstruktur durch das Zurücksenden der Nachricht KSB_CLIENT_LACB_PONG auf den "Aktiv-Check" des Kommunikationsmonitors reagiert, so schickt der Agenden-Controller diese Nachricht an den LEU-Kommunikationsmonitor.

Nachrichtenaustausch mit dem Dialog-Interpreter

- **KSB_DI_ACB_START** (Dialog-Int. → Agenden-Contr.)
  Mit dieser Nachricht sendet ein Dialog-Interpreter dem Agenden-Controller Informationen, die er an den LEU-Kommunikationsmonitor weiterreicht.

- **KSB_CLIENT_LACB_PONG** (Dialog-Int. → Agenden-Contr.)
  Diese Nachricht schickt der Dialog-Interpreter bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.

- **KSB_LACB_CLIENT_PING** (Agenden-Contr. → Dialog-Int.)
  Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf einen Dialog-Interpreter durchgeführt, so leitet der Agenden-Controller diese Nachricht an den zu überprüfenden Dialog-Interpreter weiter. Der Dialog-Interpreter antwortet mit der Nachricht KSB_CLIENT_LACB_PONG.

Nachrichtenaustausch mit dem C-Interpreter

- **KSB_CI_ACB_START** (C-Int. → Agenden-Contr.)
  Mit dieser Nachricht sendet ein C-Interpreter dem Agenden-Controller Informationen, die er an den LEU-Kommunikationsmonitor weiterreicht.

- **KSB_CLIENT_LACB_PONG** (C-Int. → Agenden-Contr.)
  Diese Nachricht schickt der C-Interpreter bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.

- **KSB_LACB_CLIENT_PING** (Agenden-Contr. → C-Int.)
  Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf einen C-Interpreter durchgeführt, so leitet der Agenden-Controller diese Nachricht an den zu überprüfenden C-Interpreter weiter. Der C-Interpreter antwortet mit der Nachricht KSB_CLIENT_LACB_PONG.

Nachrichtenaustausch mit dem Geschäftsprozeß-Monitor

- **KSB_PM_ACB_START** (GP-Monitor → Agenden-Contr.)
  Mit dieser Nachricht sendet der Geschäftsprozeß-Monitor dem Agenden-Controller Informationen, die er an den LEU-Kommunikationsmonitor weiterreicht.
- **KSB_CLIENT_LACB_PONG** (GP-Monitor → Agenden-Contr.)
  Diese Nachricht schickt der Geschäftsprozeß-Monitor bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.
- **KSB_LACB_CLIENT_PING** (Agenden-Contr. → GP-Monitor)
  Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf den Geschäftsprozeß-Monitor durchgeführt, so leitet der Agenden-Controller diese Nachricht an den Geschäftsprozeß-Monitor weiter. Der Geschäftsprozeß-Monitor antwortet mit der Nachricht KSB_CLIENT_LACB_PONG.

Nachdem nun alle Nachrichten, für deren Empfang oder deren Übermittlung der Agenden-Controller zuständig ist, kurz beschrieben worden sind, soll nun der Nachrichtenfluß, den die verschiedenen Ereignisse auslösen, erläutert werden.

- **Starten eines neuen Geschäftsprozesses**
  Wird im Agenda-Terminal von einem Anwendungsbenutzer ein neuer Geschäftsprozeß gestartet, so startet der LEU-Controller einen neuen Agenden-Controller. Nachdem der Agenden-Controller sich komplett initialisiert hat, sendet er die Nachricht *KSB_ACB_LC_LEUAC_HELLO* an den LEU-Controller. Dieser übermittelt dem Agenden-Controller hierauf mit der Nachricht *KSB_LC_ACB_START_INC* alle benötigten Informationen über das Geschäftsprozeß-Modell, zu dem ein neuer Geschäftsprozeß gestartet werden soll. Nach dem Erhalt dieser Nachricht startet der Agenden-Controller den Geschäftsprozeß-Interpreter. Nach dem Ende seiner Initialisierungsphase sendet der Geschäftsprozeß-Interpreter die Nachricht *KSB_VI_ACB_INIT* an den Agenden-Controller. Nach Erhalt dieser Nachricht übermittelt dieser dem Geschäftsprozeß-Interpreter mit der Nachricht *KSB_ACB_VI_START_NEW_PROCESS* alle Informationen, die er zum Starten des neuen Geschäftsprozesses benötigt. Nachdem der Geschäftsprozeß-Interpreter den neuen Geschäftsprozeß in die Datenbank und in seine interne Datenhaltung eingetragen hat, sendet er die Nachricht *KSB_VI_ACB_PROCESS_STARTED* an den

Agenden-Controller und beginnt danach mit der Interpretation des zum Geschäftsprozeß gehörenden FUNSOFT-Netzes. Der Agenden-Controller benachrichtigt mit der Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die Existenz des neuen Geschäftsprozesses. Danach melden sich alle Agenda-Terminals mit der Nachricht *KSB_TI_ACB_NEW_PROCESS_LOGIN* beim Agenden-Controller an. Diese Nachricht enthält unter anderem die Rollen, mit denen der jeweilige Sachbearbeiter in der LEU-Ausführung eingeloggt ist. Der Agenden-Controller überprüft für jeden Sachbearbeiter, ob er zur Teilnahme am Geschäftsprozeß berechtigt ist. Das Ergebnis der Berechtigungsüberprüfung wird über die Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an die Agenda-Terminals zurückgemeldet.

- **Beendigung eines Geschäftsprozesses**
  Enthält ein Geschäftsprozeß keinen $\xrightarrow{2.2}$Systemkanal, über den er erneut aktiviert werden kann, so erreicht er sein definiertes Ende, sobald sich keine Aktivitäten mehr in Bearbeitung befinden und auch keine mehr aktivierbar sind. In der Terminologie der FUNSOFT-Netze spricht man dann von einem "*toten Netz*". Ist dieser Zustand in einem Geschäftsprozeß erreicht, so wird er vom Geschäftsprozeß-Interpreter aus der Datenbank gelöscht. Außerdem übermittelt der Geschäftsprozeß-Interpreter die Nachricht *KSB_VI_ACB_DEAD_NET* an den Agenden-Controller und beendet sich danach selbst. Nach dem Erhalt dieser Nachricht löscht der Agenden-Controller über die Nachricht *KSB_ACB_TI_END_OF_INC* den Geschäftsprozeß aus allen Agenda-Terminals. Er benachrichtigt den LEU-Controller mit der Nachricht *KSB_ACB_LC_END_OF_INC* über die Beendigung des Geschäftsprozesses. Danach fährt er den IPC-Service herunter und terminiert hierauf als letzter Client der Controllerstruktur.

- **Beenden eines Geschäftsprozesses durch einen Benutzer**
  Wird im Agenda-Terminal ein nicht mehr benötigter Geschäftsprozeß zum Beenden ausgewählt, so wird dieser mit der Nachricht *KSB_TI_ACB_SHUTDOWN* an den zugehörigen Agenden-Controller übermittelt. Der Agenden-Controller löscht nach dem Emp-

fang dieser Nachricht über die Nachricht *KSB_ACB_TI_END_OF_INC* den Geschäftsprozeß aus allen Agenda-Terminals und leitet danach keine Aktivitäten mehr an die Agenda-Terminals weiter. Der Geschäftsprozeß-Interpreter wird über die Nachricht *KSB_ACB_VI_SHUTDOWN* benachrichtigt. Nach deren Empfang sendet der Geschäftsprozeß-Interpreter Nachrichten an alle laufenden Aktivitäts-Interpreter, auf deren Empfang hin sie sich beenden. Nachdem sich der letzte Aktivitäts-Interpreter beendet hat, löscht der Geschäftsprozeß-Interpreter den kompletten Geschäftsprozeß aus der Datenbank. Anschließend sendet er die Nachricht *KSB_VI_ACB_SHUTDOWN_EXECUTED* an den Agenden-Controller und beendet sich hierauf selbst. Wenn der Agenden-Controller die Nachricht *KSB_VI_ACB_SHUTDOWN_EXECUTED* empfangen hat, sendet er die Nachricht *KSB_ACB_LC_PROCESS_STOPPED* an den LEU-Controller, damit dieser den Geschäftsprozeß aus seiner Datenhaltung löschen kann, fährt den IPC-Service herunter und beendet sich als letzter Client der Controllerstruktur. Der Geschäftsprozeß ist danach komplett gelöscht.

- **Anhalten eines Geschäftsprozesses durch einen Benutzer**
  Soll im Agenda-Terminal ein später noch benötigter Geschäftsprozeß kurzzeitig heruntergefahren werden, so wird dieser mit der Nachricht *KSB_TI_ACB_STOP_PROCESS* an den zugehörigen Agenden-Controller übermittelt. Der Agenden-Controller löscht nach dem Empfang dieser Mitteilung über die Nachricht *KSB_ACB_TI_END_OF_INC* den Geschäftsprozeß aus allen Agenda-Terminals und leitet danach keine Aktivitäten mehr an die sie weiter. Der Geschäftsprozeß-Interpreter wird über die Nachricht *KSB_ACB_VI_STOP_PROCESS* informiert. Er wartet danach auf die Terminierung aller laufenden Aktivitäts-Interpreter, um alle Aktivitäten, die derzeit noch bearbeitet werden, korrekt terminieren zu lassen und schreibt danach den aktuellen Bearbeitungsstand des Geschäftsprozesses in die Datenbank. Anschließend sendet er die Nachricht *KSB_VI_ACB_PROCESS_STOPPED* an den Agenden-Controller und beendet sich hierauf selbst. Wenn der Agenden-Con-

troller die Nachricht *KSB_VI_ACB_PROCESS_STOPPED* empfangen hat, sendet er die Nachricht *KSB_ACB_LC_PROCESS_SUSPENDED* an den LEU-Controller, damit dieser die Controllerstruktur in seiner Datenhaltung entsprechend anpassen kann, fährt den IPC-Service herunter und terminiert als letzter Client der Controllerstruktur.

- **Erneuter Start eines angehaltenen Geschäftsprozesses**
  Im Agenda-Terminal können alle angehaltenen Geschäftsprozesse erneut zur Ausführung gebracht werden. Wird im Agenda-Terminal ein solcher Geschäftsprozeß erneut gestartet, so wird dieser Prozeß mit der Nachricht *KSB_TI_LC_RESTART_INC* an den LEU-Controller übermittelt. Vom LEU-Controller wird ein neuer Agenden-Controller gestartet. Nachdem der Agenden-Controller sich komplett initialisiert hat, sendet er die Nachricht *KSB_ACB_LC_LEUAC_HELLO* an den LEU-Controller. Nach dem Erhalt dieser Nachricht übermittelt dieser dem Agenden-Controller mit der Nachricht *KSB_LC_ACB_RESTART_INC* alle benötigten Informationen über den erneut zu startenden Geschäftsprozeß. Nach dem Erhalt dieser Nachricht startet der Agenden-Controller den Geschäftsprozeß-Interpreter. Nach dem Ende seiner Initialisierungsphase sendet der Geschäftsprozeß-Interpreter die Nachricht *KSB_VI_ACB_INIT* an den Agenden-Controller. Nach Erhalt dieser Mitteilung übermittelt der Agenden-Controller dem Geschäftsprozeß-Interpreter mit der Nachricht *KSB_ACB_VI_START_EXISTING_PROCESS* alle Informationen, die er zum erneuten Starten des Geschäftsprozesses benötigt. Nachdem der Geschäftsprozeß-Interpreter den Geschäftsprozeß aus der Datenbank ausgelesen und in seine interne Datenhaltung eingetragen hat, sendet er die Nachricht *KSB_VI_ACB_PROCESS_CONTINUED* und beginnt danach mit der weiteren Interpretation des zum Geschäftsprozeß gehörenden FUNSOFT-Netzes. Der Agenden-Controller benachrichtigt über die Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die erneute Existenz des Geschäftsprozesses. Danach melden sich alle Agenda-Terminals mit der Nachricht *KSB_TI_ACB_NEW_PROCESS_LOGIN* beim Agenden-Controller

an. Diese Nachricht enthält unter anderem die Rollen, mit denen der jeweilige Sachbearbeiter in der LEU-Ausführung eingeloggt ist. Der Agenden-Controller überprüft für jeden Sachbearbeiter, ob er zur Teilnahme am Geschäftsprozeß berechtigt ist. Das Ergebnis der Berechtigungsüberprüfung wird mit der Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an die Agenda-Terminals zurückgemeldet.

- **Anmeldung eines neuen Sachbearbeiters**
  Ein neuer Sachbearbeiter meldet sich mit der Nachricht *KSB_TI_ACB_NEW_USER_LOGIN* bei allen Agenden-Controller an. Diese Nachricht enthält unter anderem die Rollen, mit denen sich der jeweilige Sachbearbeiter in der LEU-Ausführung einloggt. Die Agenden-Controller überprüfen für den neuen Sachbearbeiter, ob er zur Teilnahme am jeweiligen Geschäftsprozeß berechtigt ist. Das Ergebnis wird über die Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an das Agenda-Terminals zurückgemeldet. Wenn der Benutzer zur Mitarbeit am Geschäftsprozeß berechtigt ist, werden ihm danach alle Aktivitäten, für deren Bearbeitung er zuständig ist, mit den Nachrichten *KSB_ACB_TI_AGENCY* und *KSB_ACB_TI_PRED_AGENCY* übermittelt.

- **Abmeldung eines Sachbearbeiters**
  Wenn ein Sachbearbeiter sein Agenda-Terminal verläßt, meldet es sich bei allen Agenden-Controllern mit der Nachricht *KSB_TI_ACB_END_TERM* ab. Die Agenden-Controller löschen daraufhin alle Daten, die für das Agenda-Terminal bzw. den Sachbearbeiter gespeichert wurden, aus ihrer internen Datenhaltung.

- **Ein Sachbearbeiter ändert sein Berechtigungsprofil**
  Verändert ein Sachbearbeiter während der Arbeit in den Geschäftsprozessen seine Rollenmenge (siehe Abb. 5.10 auf Seite 101), so meldet sich sein Agenda-Terminal zunächst über die Nachricht *KSB_TI_ACB_END_TERM* bei allen Geschäftsprozessen ab. Im Agenda-Terminal werden alle Geschäftsprozesse aus der internen Datenhaltung gelöscht. Direkt danach meldet sich das Agenda-Terminal

wieder bei allen Agenden-Controllern an (siehe →*Anmeldung eines neuen Sachbearbeiters*).

- **Benachrichtigung über eine neue Aktivität - Update einer Aktivität**
  Mit den Nachrichten *KSB_VI_ACB_PRED_AGENCY* bzw. *KSB_VI_ACB_AGENCY* übermittelt der Geschäftsprozeß-Interpreter neue prädikatbehaftete (*KSB_VI_ACB_PRED_AGENCY*) und nicht prädikatbehaftete (*KSB_VI_ACB_AGENCY*) Aktivitäten sowie Aktivitäten, deren Vorbereich ($\xrightarrow{2.2}$Kanäle) sich geändert hat, an den Agenden-Controller. Die beiden Nachrichten enthalten ein Update-Flag. Ist das Update-Flag gesetzt, hat sich der Vorbereich der Aktivität geändert und die Aktivität muß in den Agenden der Sachbearbeiter aktualisiert werden. Dazu löscht der Agenden-Controller über die Nachricht *KSB_ACB_TI_DELETE_AGENCY* die Aktivität aus den Agenden der Sachbearbeiter. Direkt danach übermittelt der Agenden-Controller die Nachricht *KSB_ACB_TI_PRED_AGENCY* (bei prädikatbehafteten Aktivitäten) bzw. *KSB_ACB_TI_AGENCY* (bei nicht prädikatbehafteten Aktivitäten) an alle Agenda-Terminals, deren Benutzer zur Bearbeitung dieser Aktivität berechtigt sind. Ist das Update-Flag in den Nachrichten *KSB_VI_ACB_PRED_AGENCY* bzw. *KSB_VI_ACB_AGENCY* nicht gesetzt, entfällt der Schritt zum Löschen der Aktivität aus den Agenden der Sachbearbeiter, da die Aktivität bisher noch nicht in den Agenden stand. Die Aktivität wird in die Datenhaltung des Agenden-Controllers eingetragen bzw. dort aktualisiert.

- **Benachrichtigung über eine zu löschende Aktivität**
  Ist eine Aktivität nicht mehr aktivierbar, so wird sie mit der Nachricht *KSB_VI_ACB_DELETE_AGENCY* vom Geschäftsprozeß-Interpreter an den Agenden-Controller übermittelt. Dieser übermittelt mit der Nachricht *KSB_ACB_TI_DELETE_AGENCY* die Aktivität zum Löschen an alle Agenda-Terminals, die die Aktivität in ihren Agenden stehen haben, und löscht sie aus seiner internen Datenhaltung.

- **Benachrichtigung über die Beendigung einer Aktivität**
  Wurde die Bearbeitung einer Aktivität von einem Anwendungsbenutzer abgeschlossen, so übermittelt der Geschäftsprozeß-Interpreter diese Aktivität mit der Nachricht *KSB_VI_ACB_AGENCY_DONE* an den Agenden-Controller. Dieser leitet die Aktivität über die Nachricht *KSB_ACB_TI_AGENCY_DONE* an das Agenda-Terminal des Anwendungsbenutzers weiter.

- **Ein Objekt wurde in einen $\xrightarrow{2.2}$Systemkanal geschrieben**
  Wurde in einem Geschäftsprozeß ein Systemkanal gefüllt, so müssen die Objekte des Kanals an alle Geschäftsprozesse, die diesen Systemkanal benutzen, übermittelt werden. Dazu übermittelt der Geschäftsprozeß-Interpreter dem Agenden-Controller die Nachricht *KSB_VI_ACB_SYSTEM_CHANNEL*. Diese Mitteilung enthält alle benötigten Informationen zu den Objekten im Systemkanal. Der Agenden-Controller reicht diese Informationen über die Nachricht *KSB_ACB_LC_SYSTEM_CHANNEL* an den LEU-Controller weiter. Von ihm wird die Nachricht nun an die Agenden-Controller aller laufenden Geschäftsprozesse mit der Nachricht *KSB_LC_ACB_SYSTEM_CHANNEL* übermittelt. Die so benachrichtigten Agenden-Controller reichen diese Informationen über die Nachricht *KSB_ACB_VI_SYSTEM_CHANNEL* an ihren Geschäftsprozeß-Interpreter durch. Diese überprüfen nach dem Erhalt dieser Nachricht, ob der Systemkanal in ihren Prozeß eingebunden ist, und schreiben gegebenenfalls die neuen Objekte in ihre Datenhaltung.

- **Auswahl einer Aktivität durch einen Benutzer**
  Hat ein Anwendungsbenutzer im Agenda-Terminal eine Aktivität zur Bearbeitung ausgewählt, so wird diese Aktivität mit der Nachricht *KSB_TI_ACB_START_PRED_AGENCY* bzw. *KSB_TI_ACB_START_AGENCY* an den zugehörigen Agenden-Controller übermittelt. Dieser überprüft in seiner internen Datenhaltung, ob diese Aktivität vielleicht zeitlich direkt zuvor schon von einem anderen Benutzer zur Bearbeitung ausgewählt worden ist.
  Ist das der Fall, so wird die Nachricht *KSB_ACB_TI_AGENCY_-*

ALREADY_WORKING an das Agenda-Terminal des zu spät gekommenden Anwendungsbenutzer übermittelt. Diese Nachricht öffnet im Agenda-Terminal eine Info-Box, in dem die Aktivität und der Benutzer, der ihm zuvorgekommen ist, angezeigt wird.
Wird im Agenden-Controller festgestellt, daß die Aktivität noch von keinem Benutzer bearbeitet wird, wird sie mit der Nachricht *KSB_ACB_VI_START_PRED_AGENCY* bzw. *KSB_ACB_VI_START_AGENCY* an den Geschäftsprozeß-Interpreter weitergereicht. Der Geschäftsprozeß-Interpreter führt eine Aktivierbarkeitsprüfung dieser Aktivität durch und stellt somit sicher, daß die Aktivität noch immer mit den aus dem Vorbereich ($\xrightarrow{2.2}$Instanzen) ausgewählten Objekten aktivierbar ist.
Verläuft die Aktivierbarkeitsprüfung positiv, so wird für den zu dieser Aktivität gehörenden Dialog ein Dialog-Interpreter gestartet und der Dialog erscheint auf dem Bildschirm des Anwendungsbenutzers. Verläuft die Aktivierbarkeitsprüfung negativ, so wird die Nachricht *KSB_VI_ACB_ACTIVATION_FAILED* an den Agenden-Controller zurückgegeben und diese Information von ihm mit der Nachricht *KSB_ACB_TI_AGENCY_ALREADY_WORKING* an das Agenda-Terminal weitergereicht. In der Verarbeitung dieser Mitteilung wird auf dem Bildschirm des Anwendungsbenutzers eine Info-Box geöffnet, welche diesen Sachverhalt verdeutlicht.

- **Zurücksetzen einer Aktivität durch einen Benutzer**
  Hat ein Anwendungsbenutzer im Agenda-Terminal eine manuelle Aktivität zum Zurücksetzen ausgewählt, so wird sie mit der Nachricht *KSB_TI_ACB_RESET_AGENCY* an den zugehörigen Agenden-Controller übermittelt. Dieser leitet die Aktivität über die Nachricht *KSB_ACB_VI_RESET_AGENCY* an den Geschäftsprozeß-Interpreter weiter. Der Geschäftsprozeß-Interpreter schickt dem zugehörigen Dialog-Interpreter eine Nachricht, so daß dieser terminiert und stellt im FUNSOFT-Netz den Zustand der Aktivität vor ihrer Aktivierung wieder her (Rollback in der Datenbank). Danach wird die Aktivität über den Agenden-Controller wieder in die Agenden aller berechtigten Sachbearbeiter geschrieben.

- **Anmeldung des LEU-Kommunikationsmonitors**
  Nach seinem Start meldet sich der Kommunikationsmonitor (Beschreibung siehe Kapitel 2.5.7) bei allen Agenden-Controllern über die Nachricht *KSB_CM_ACB_START* an. Die Agenden-Controller übermitteln dem Kommunikationsmonitor danach mit der Nachricht *KSB_ACB_CM_CLIENT_LIST* Informationen zu allen Clients ihrer Agenda-Controllerstruktur.

- **Anmeldung eines neuen Clients**
  Neue Clients der Agenda-Controllerstruktur melden sich mit der Nachricht *KSB_xx_ACB_START* beim Agenden-Controller an. Das *"xx"* steht hierbei für das Kürzel des Clienttyps (z.B. *"TI"* für ein Agenda-Terminal). Die Mitteilung enthält Informationen, die an den LEU-Kommunikationsmonitor weiterzureichen sind. Diese Informationen werden in einer internen Datenhaltung des Agenden-Controllers abgespeichert und, falls der Kommunikationsmonitor sich als Client in der Controllerstruktur befindet, direkt über die Nachricht *KSB_LACB_CM_CLIENT_HELLO* an den Kommunikationsmonitor weitergeleitet.

- **Korrekte Terminierung eines Clients**
  Hat ein Client seine Aufgabe beendet, so meldet er sich über die Controllernachricht *CLIENT_BY* bei allen Clients der Agenda-Controllerstruktur ab. Empfängt der Agenden-Controller eine solche Mitteilung, so löscht er den Client aus seiner internen Datenstruktur und übermittelt dem LEU-Kommunikationsmonitor, falls dieser sich als Client in der Controllerstruktur befindet, die Nachricht *KSB_LACB_CM_CLIENT_BY*.

- **Unkorrekte Terminierung (Absturz) eines Clients**
  Terminiert ein Client innerhalb der Controllerstruktur unkontrolliert (beim Absturz eines Clients wird die Controllernachricht *CLIENT_CRASHED* an alle Clients der Controllerstruktur übermittelt), so wird er von einem anderen Client der gleichen Controllerstruktur erneut wieder hochgefahren. Die Aktivitäts-Interpreter werden hierbei vom Geschäftsprozeß-Interpreter erneut gestartet. Der Absturz eines

Agenden-Controllers oder eines Geschäftsprozeß-Interpreters werden weiter unten gesondert behandelt. Befindet sich der Kommunikationsmonitor als Client in der Controllerstruktur, so wird ihm der Absturz eines Clients über die Nachricht *KSB_LACB_CM_CLIENT_CRASH* mitgeteilt.

- **Unkorrekte Terminierung (Absturz) des Agenden-Controllers**
  Terminiert der Agenden-Controller während der Ausführung eines Geschäftsprozesses unkontrolliert, so wird vom IPC-Service die Controllernachricht *CLIENT_CRASHED* an alle Clients der LEU- und Agenda-Controllerstruktur übermittelt. Die Agenda-Terminals reagieren auf den Empfang dieser Mitteilung dadurch, daß sie den Geschäftsprozeß vorübergehend aus ihrer Datenhaltung löschen. Der LEU-Controller startet nach dem Erhalt dieser Nachricht den Agenden-Controller neu. Nach seiner Initialisierungsphase meldet sich der Agenden-Controller mit der Nachricht *KSB_ACB_LC_LEUAC_HELLO* beim LEU-Controller zurück. Dieser übermittelt ihm darauf mit der Nachricht *KSB_LC_ACB_RESTART_ACB* alle Informationen, die er benötigt, um seine interne Datenhaltung wieder aufzubauen. Nach der Verarbeitung dieser Mitteilung meldet sich der Agenden-Controller über die Nachricht *KSB_ACB_VI_ACB_RESTARTED* beim Geschäftsprozeß-Interpreter zurück, der ihm nach dem Empfang dieser Nachricht erneut alle aktivierbaren manuellen Aktivitäten übermittelt. Der Agenden-Controller benachrichtigt mit der Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die erneute Existenz des Geschäftsprozesses. Danach melden sich alle Agenda-Terminals mit der Nachricht *KSB_TI_ACB_NEW_PROCESS_LOGIN* beim Agenden-Controller an. Diese Mitteilung enthält unter anderem die Rollen, mit denen die jeweiligen Sachbearbeiter in der LEU-Ausführung eingeloggt sind. Der Agenden-Controller überprüft für jeden Sachbearbeiter, ob er zur Teilnahme am Geschäftsprozeß berechtigt ist. Das Ergebnis der Berechtigungsüberprüfung wird mit der Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an die Agenda-Terminals

zurückgemeldet. Danach werden alle vom Geschäftsprozeß-Interpreter übermittelten Aktivitäten wieder an die Agenda-Terminals der für ihre Bearbeitung berechtigten Sachbearbeiter weitergeleitet.

- **Unkorrekte Terminierung (Absturz) des Geschäftsprozeß-Interpreters**
  Terminiert der Geschäftsprozeß-Interpreter während der Ausführung eines Geschäftsprozesses unkontrolliert, so wird vom IPC-Service die Controllernachricht *CLIENT_CRASHED* an alle anderen Clients der Agenda-Controllerstruktur übermittelt. Nach dem Erhalt dieser Nachricht übermittelt der Agenden-Controller die Nachricht *KSB_ACB_TI_END_OF_INC* an alle an diesem Geschäftsprozeß beteiligten Agenda-Terminals. Nach Erhalt dieser Mitteilung wird der Geschäftsprozeß vorübergehend aus der Datenhaltung der Agenda-Terminals gelöscht. Daraufhin startet der Agenden-Controller den Geschäftsprozeß-Interpreter erneut. Nach seiner Initialisierungsphase meldet sich dieser beim Agenden-Controller mit der Nachricht *KSB_VI_ACB_INIT* zurück. Der Agenden-Controller übermittelt ihm hierauf über die Nachricht *KSB_ACB_VI_RESTART_VI* den Datenbankschlüssel des Geschäftsprozesses, dessen FUNSOFT-Netz er weiter interpretieren soll. Der Geschäftsprozeß-Interpreter liest daraufhin den zuletzt in der Datenbank festgeschriebenen Stand des Geschäftsprozesses aus, füllt seine interne Datenhaltung und sendet dann die Nachricht *KSB_VI_ACB_RESTARTED* an den Agenden-Controller zurück. Danach übernimmt er die weitere Interpretation des FUNSOFT-Netzes und übermittelt alle ausführbaren manuellen Aktivitäten an den Agenden-Controller. Nachdem dieser die Nachricht *KSB_VI_ACB_RESTARTED* vom Geschäftsprozeß-Interpreter erhalten hat, unterrichtet er mit der Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die erneute Existenz des Geschäftsprozesses. Danach melden sich alle Agenda-Terminals mit der Nachricht *KSB_TI_ACB_NEW_PROCESS_LOGIN* beim Agenden-Controller an. Diese Mitteilung enthält unter anderem die Rollen, mit denen der jeweilige Sachbearbeiter in der LEU-Ausführung eingeloggt ist. Der Agenden-Controller überprüft für je-

den Sachbearbeiter, ob er zur Teilnahme am Geschäftsprozeß berechtigt ist. Das Ergebnis der Berechtigungsüberprüfung wird mit der Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an die Agenda-Terminals zurückgemeldet. Danach werden alle vom Geschäftsprozeß-Interpreter übermittelten Aktivitäten wieder an die Agenda-Terminals der für ihre Bearbeitung berechtigten Sachbearbeiter weitergeleitet.

3.2.3 Die Modularisierung des Agenden-Controllers

Während der Entwurfsphase der LEU-Ausführungskomponente wurde die Entscheidung getroffen, die einzelnen Ausführungswerkzeuge nicht mit einer Ablaufsteuerung zu einem Prozeß zusammenzubinden, sondern die einzelnen Ausführungswerkzeuge als eigenständige Betriebssytem-Prozesse zu realisieren. Der Weg über eine Ablaufsteuerung wäre nicht flexibel und performant genug gewesen, um mehrere Geschäftsprozesse gleichzeitig ausführen zu können und mehrere Sachbearbeiter an einem Geschäftsprozeß teilnehmen zu lassen. Die Ablaufsteuerung, mit der dann zum Beispiel alle Agenda-Terminals hätten kommunizieren müssen, wäre zum *"Bottle Neck"* der LEU-Ausführungskomponente geworden, da alle Informationen, die bei der Ausführung der Geschäftsprozesse zwischen den einzelnen Werkzeugen auszutauschen sind, zentral über die Ablaufsteuerung hätten laufen müssen. Durch die Verteilung der LEU-Ausführungskomponente auf mehrere eigenständige Prozesse wurde eine Interprozeß-Kommunikation nötig, die den Datenaustausch zwischen den Prozessen realisiert.
Abbildung 3.1 verdeutlicht diese Entwurfsentscheidung. Links in der Abbildung wird die Prozeß-Sicht auf die LEU-Ausführungskomponente dargestellt, wenn die Entscheidung zugunsten einer Ablaufsteuerung getroffen worden wäre. Der rechte Teil der Abbildung zeigt die Prozeß-Sicht der realisierten LEU-Ausführungskomponente.

Diese Betrachtung hilft beim Verständnis der im weiteren Verlauf benutzten Begriffe *Prozeß*, *Komponente*, *Modul* und *Funktion*. Mit diesen Begriffen ist im weiteren Text folgendes gemeint:

- **Prozeß**
  Unter einem Prozeß wird im weiteren ein als eigenständiger

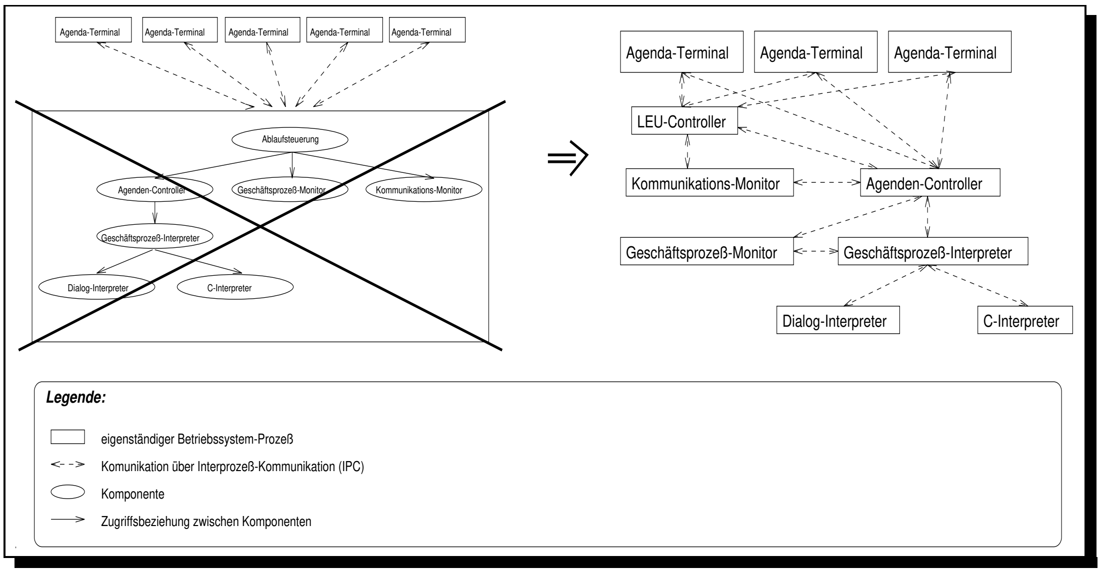


Abbildung 3.1: Prozeß-Sicht der LEU-Ausführungskomponente

Betriebssystem-Prozeß realisiertes Werkzeug verstanden. Die Prozesse unterteilen sich in *Komponenten*.

- **Komponente**
  Eine Komponente besteht aus einem oder mehreren *lokalen und globalen Modulen*, die funktional zusammengehören. Die globalen Module stellen in ihrer Exportschnittstelle Funktionen zur Verfügung, die auch von anderen Komponenten aufgerufen werden können. Die lokalen Module stellen keine Funktionen in der Exportschnittstelle ihres Moduls zur Verfügung. Sie enthalten nur lokale und komponentenglobale Funktionen, die nur innerhalb des Moduls bzw. nur innerhalb der eigenen Komponente benutzt werden dürfen.

- **Modul**
  Module bestehen aus einer Ansammlung von Funktionen und lassen sich nach mehreren Kriterien einteilen. Es gibt beispielsweise *Funktions-* und *Datenmodule*. Die Funktionsmodule haben keine eigene Datenhaltung (kein eigenes Gedächtnis) und stellen Algorithmen zur Verwendung in anderen Modulen zur Verfügung. Datenmo-

dule dienen der Verkapselung abstrakter Datentypen. Sie bilden das Gedächtnis einer Komponente. In einem Datenmodul werden alle Objekte des zu dem Modul gehörenden abstrakten Datentyps gespeichert. Datenmodule stellen in ihrer Exportschnittstelle alle für die Objekte des Datentyps benötigten Zugriffsfunktionen zur Verfügung.
Enthält ein Modul *globale Funktionen*, so wird es als globales Modul bezeichnet. Enthält es keine, so ist es ein lokales Modul. Die globalen Funktionen der globalen Module bilden die Exportschnittstelle der zugehörigen Komponente.

- **Funktion**
  Funktionen können global, komponentenglobal und lokal sein. Die globalen Funktionen stehen in der Exportschnittstelle ihres Moduls und bilden gleichzeitig die Exportschnittstelle der zugehörigen Komponente. Die komponentenglobalen Funktionen stehen ebenfalls in der Exportschnittstelle ihres Moduls, können aber nur von Modulen aus der gleichen Komponente benutzt werden. Die lokalen Funktionen können nur innerhalb des Moduls, in dem sie implementiert sind, von den Funktionen des gleichen Moduls benutzt werden. Sie stehen nicht in der Exportschnittstelle des Moduls.

3.2.3.1 Die Komponenten des LEU-Agenden-Controllers

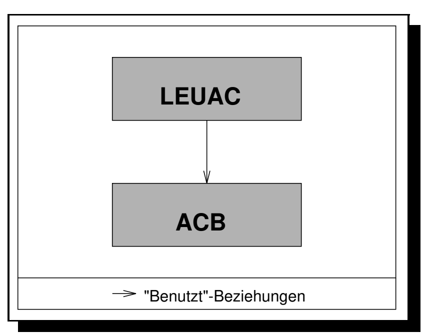


Abbildung 3.2: Komponenten des LEU-Agenden-Controllers

Wie in Abbildung 3.2 zu sehen, besteht der LEU-Agenden-Controller aus den Komponenten **LEUAC** und **ACB**. Die Komponente *LEUAC* besteht aus dem Modul *leuac*. Das Modul enthält die Main-Routine und die Exit-Funktion des Agenden-Controllers. Innerhalb dieses Moduls wird bei der Initialisierung ein neuer IPC-Service gestartet und gegebenenfalls der LEU-Controller von der korrekten Initialisierung benachrichtigt. Die Main-Routine enthält die *"Messageloop"* für die eingehenden IPC-Nachrichten. Sie ist so realisiert, daß der Prozeß in einen *"IDLE"-Modus* wechselt, wenn keine weiteren Mitteilungen am IPC-Socket anliegen. Auf diese Weise verbraucht der Prozeß keine Rechenzeit. Liegt dann am IPC-Socket eine neue Nachricht an, so wird der Prozeß über ein hierdurch ausgelöstes *"SIG_IO"-Signal* wieder aktiviert. Alle ankommenden IPC-Nachrichten werden in eine *"Message-Queue"* geschrieben und dann in der *"Messageloop"* verarbeitet. Die Komponente *LEUAC* stützt sich hierzu auf die Komponente *ACB* ab. Die Komponente *ACB* realisiert die Datenhaltung des Agenden-Controllers. Sie enthält Funktionen zur Verarbeitung der Nachrichten und zur Kommunikation mit den anderen LEU-Ausführungswerkzeugen.

3.2.3.2 Die Komponente ACB

Die Funktionalität der einzelnen Module des Agenden-Controllers wird im folgenden kurz beschrieben.

Die Module *ACBin, ACBvii, ACBtii, ACBlci* und *ACBcmi* dienen dem Empfang und der Verarbeitung von IPC-Nachrichten. Sie bilden eine Datenkapsel für den Objekttyp *KSBme_Message*. Der abstrakte Datentyp *KSBme_Message* verkapselt die Nachrichten, die zwischen den LEU-Ausführungswerkzeugen mittels IPC ausgetauscht werden müssen.

- **ACBin**
  Dieses Modul enthält den MessageHandler, der angesprungen wird, sobald am Socket eine neue Nachricht anliegt. Zusätzlich übernimmt dieses Modul über die Funktion *ACBin_Init* die komplette Initialisierung der Komponente ACB.

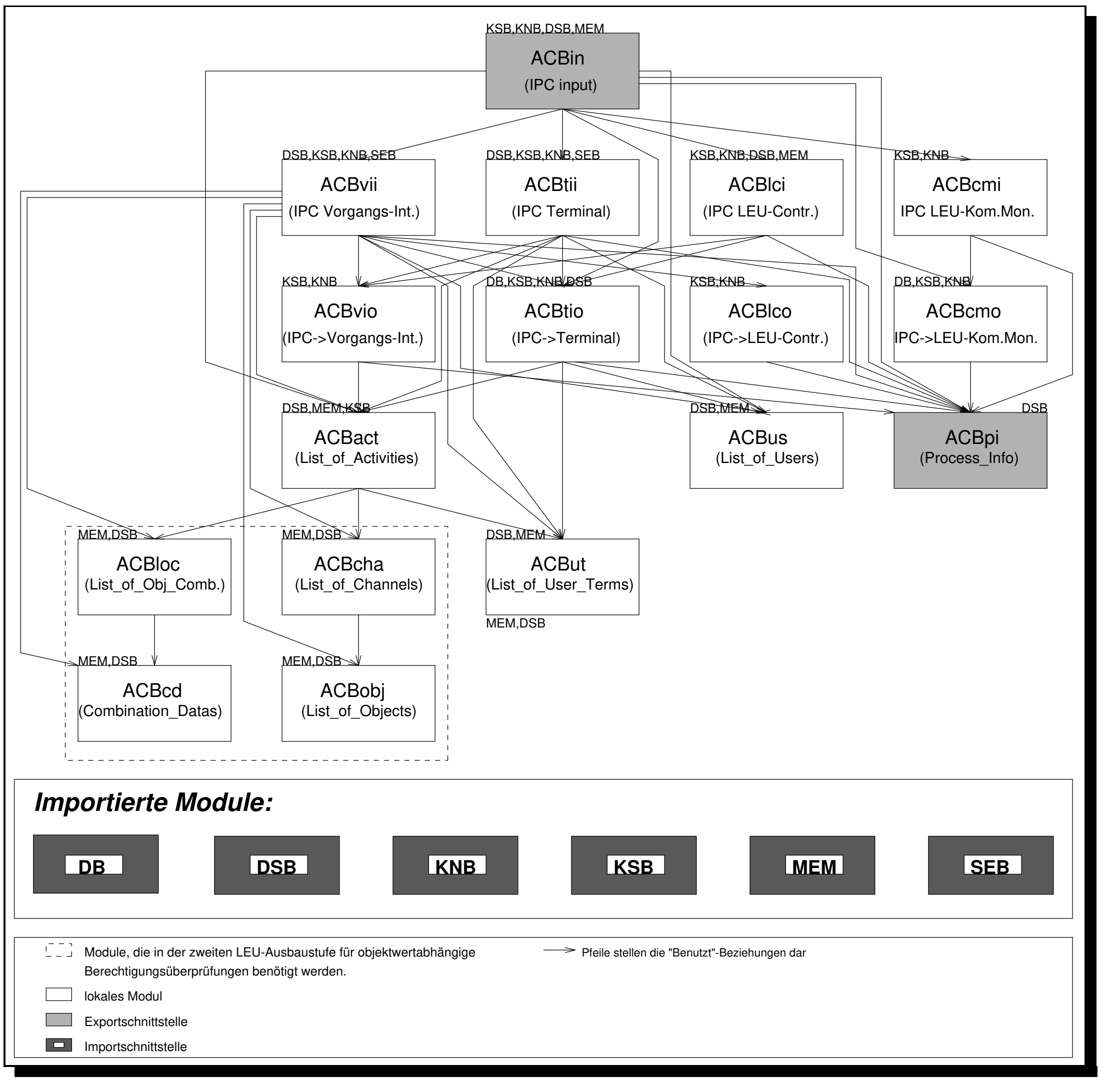


Abbildung 3.3: Modulhierarchie des Agenden-Controllers

- **ACBvii**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Geschäftsprozeß-Interpreter (VI) an den Agenden-Controller (ACB) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Nachricht übernimmt.

- **ACBtii**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenda-Terminal (TI) an den Agenden-Controller (ACB) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

- **ACBlci**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom LEU-Controller (LC) an den Agenden-Controller (ACB) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

- **ACBcmi**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Kommunikationsmonitor (CM) an den Agenden-Controller (ACB) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

Die Module *ACBvio, ACBtio, ACBlco* und *ACBcmo* bilden ebenfalls eine Datenkapsel für den Objekttyp *KSBme_Message*. Sie enthalten die Verarbeitungsfunktionen, die Nachrichten an die anderen Clients der LEU-Ausführungswerkzeuge verschicken.

- **ACBvio**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenden-Controller (ACB) zum Geschäftsprozeß-Interpreter (VI) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

- **ACBtio**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenden-Controller (ACB) zum Agenda-Terminal (TI) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

- **ACBlco**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenden-Controller (ACB) zum LEU-Controller (LC) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

- **ACBcmo**
  Dieses Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenden-Controller (ACB) zum LEU-Kommunikationsmonitor (CM) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Message übernimmt.

Die Module *ACBact, ACBus, ACBpi* und *ACBut* bilden die Datenmodule der Komponente. In ihnen werden alle Daten gespeichert, die der Agenden-Controller benötigt, um die Aktivitäten unter Berücksichtigung der individuellen Benutzerberechtigungen an die Agenda-Terminals weiterzuleiten.

- **ACBact**
  Dieses Modul verkapselt den abstrakten Datentyp **Activity**. Mit dem abstrakte Datentyp *Activity* werden alle Informationen über die vom Geschäftsprozeß-Interpreter übermittelten Aktivitäten gespeichert, die der Agenden-Controller zur Berechtigungsüberprüfung und zum Weiterleiten an die Agenda-Terminals benötigt. Das Modul stellt alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

- **ACBus**
  Dieses Modul verkapselt den abstrakten Datentyp **User**. Mit dem abstrakte Datentyp *User* werden alle Informationen über die an diesem

Geschäftsprozeß beteiligten Personen gespeichert, die der Agenden-Controller zur Berechtigungsüberprüfung und zur Übermittlung von Daten an die Benutzerterminals benötigt. Das Modul stellt alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

- **ACBpi**
  In diesem Modul werden alle Informationen abgespeichert, die über den entsprechenden Gechäftsprozeß in der Datenbank abgelegt sind und von den Clients innerhalb der Agenda-Controllerstruktur erfragt werden müssen. Das Modul stellt Funktionen zum Setzen und Auslesen der gespeicherten Informationen zur Verfügung.

- **ACBut**
  Dieses Modul verkapselt den abstrakten Datentyp **User_Terminals**. Hier werden Informationen zu allen Agenda-Terminals abgelegt, an die eine bestimmte Aktivität übermittelt worden ist. Auf diese Weise kann bei der Auswahl einer Aktivität durch einen Benutzer die Aktivität einfach aus allen anderen Agenda-Terminals gelöscht werden. Das Modul stellt alle Zugriffsoperationen, die zum Einfügen, Löschen und Auslesen der Objekte dieses Datentyps benötigt werden, zur Verfügung.

Die Module *ACBloc, ACBcha, ACBcd* und *ACBobj* werden erst in der zweiten LEU-Ausbaustufe für objektwertabhängige Berechtigungsüberprüfungen benötigt. Im Rahmen dieser Diplomarbeit wurden die Datenstrukturen und benötigten Funktionen schon spezifiziert, die Funktionen jedoch noch nicht implementiert.

- **ACBloc**
  Dieses Modul verkapselt den abstrakten Datentyp **ObjectCombination**. Es wird benötigt, um die Objektkombinationen von prädikatbehafteten Aktivitäten zu speichern. Es speichert nur die direkt die Kombination betreffenden Daten. Die Objekte, mit denen die zugehörige Aktivität gestartet werden kann und die zu einer Kombination gehören, werden im Modul ACBcd abgelegt. Das Modul

stellt alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

- **ACBcha**
  Dieses Modul verkapselt den abstrakten Datentyp **Channel**. Es speichert die Daten der im Vorbereich einer Instanz / Aktivität stehenden Kanäle. Das Modul stellt alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

- **ACBcd**
  Dieses Modul verkapselt den abstrakten Datentyp **Combination_Datas**. Hier werden die Daten, die zu einer Objektkombination einer prädikatbehafteten Aktivität gehören, gespeichert. Das Modul stellt alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

- **ACBobj**
  Dieses Modul verkapselt den abstrakten Datentyp **Object**. Hier werden die Informationen, die zu einem Objekt gehören, gespeichert. Außerdem stellt das Modul alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte dieses Datentyps benötigt werden, zur Verfügung.

Bis auf die Module *ACBin* und *ACBpi* besitzen alle Module keine Exportschnittstelle, da alle Funktionen nur innerhalb der jeweiligen Module oder innerhalb der Komponente ACB benutzt werden. Das Modul *ACBin* stellt in seiner Exportschnittstelle den MessageHandler, der angesprungen wird, sobald eine neue IPC-Nachricht an das Werkzeug gesendet worden ist, Funktionen zur Initialisierung der Komponente und Funktionen zur Speicherfreigabe zur Verfügung. Die Exportschnittstelle des Moduls *ACBpi* besteht aus Funktionen, die zum Setzen und Auslesen von Geschäftsprozeß-Informationen benötigt werden.

3.2.3.3 Die Importschnittstelle

In der Importschnittstelle [1] des Agenden-Controllers liegen die Module **DB**, **DSB**, **KNB**, **KSB**, **MEM**, und **SEB**. Sie stellen folgende Funktionalität zur Verfügung.

- **Die Komponente DB**
  Die Komponente DB kapselt alle Datenbankfunktionen.
- **Die Komponente DSB**
  Die Komponente DSB ist ein Datenstrukturen-Basismodul. Es stellt Datenstrukturen und Zugriffsfunktionen zur Verfügung, auf die sich der Agenden-Controller in seinen Datenmodulen abstützt.
- **Die Komponenten KNB und KSB**
  Sie stellen Funktionen für die IPC-Kommunikation zur Verfügung.
- **Die Komponente MEM**
  In der Komponente MEM sind alle Speicherverwaltungs-Funktionen gekapselt.
- **Die Komponente SEB**
  Sie stellt Funktionen für Berechtigungsüberprüfungen zur Verfügung.

[1] Die Module der Importschnittstelle wurden nicht innerhalb dieser Diplomarbeit entwickelt.

Kapitel 4

Das Agenda-Terminal

Das Agenda-Terminal bildet die Benutzeroberfläche des Geschäftsprozeß-Management-Systems LEU. Es wurde während dieser Diplomarbeit entwickelt und in die Ausführungskomponente von LEU integriert. Der erstellte Programmcode wurde bei der Abgabe am Lehrstuhl Softwaretechnologie hinterlegt. In diesem Kapitel werden die Anforderungen (Abschnitt 4.1) und die Realisierung (Abschnitt 4.2) des Agenda-Terminals beschrieben.

4.1 Anforderungen

In Kapitel 5 wird die Entwicklung der Oberfläche des LEU-Agendaterminals erläutert. In diesem Rahmen wird auch auf die funktionalen und ergonomischen Anforderungen, denen das Agenda-Terminal zu genügen hat, eingegangen.

Das Agenda-Terminal stellt alle von den Anwendungsbenutzern benötigten Operationen auf den abstrakten Datentypen *"Geschäftsprozessen"* und *"Aktivitäten"* zur Verfügung.
Ziel der Arbeit ist es, eine für beliebige LEU-Anwendungen geeignete Oberfläche zu entwickeln. Eine weitere Anforderung besteht darin, daß das Agenda-Terminal keine Abhängigkeit von der Datenbank haben darf, um die ortsunabhängige Benutzung des Werkzeugs zu gewährleisten. Auf diese Weise kann beispielsweise ein Sachbearbeiter in Dortmund an Geschäftspro-

zessen teilnehmen, die in Bochum ausgeführt werden. Diese Anforderung bedeutete einen hohen zusätzlichen Aufwand bei der Implementierung, da alle zur Anzeige an der Oberfläche des Werkzeugs benötigten Daten über die Interprozeß-Kommunikation der LEU-Ausführungswerkzeuge in die interne Datenhaltung des Werkzeugs gelangen müssen.

4.2 Grobentwurf und Realisierung

4.2.1 Grundlegende Entwurfsentscheidungen

Dieser Abschnitt befaßt sich mit einigen in der Entwurfsphase des Agenda-Terminals getroffenen Entwurfsentscheidungen.

- **Entwicklung einer allgemeingültigen Oberfläche**
  Eine, die gesamte Entwicklung prägende Entwurfsentscheidung war, daß das Agenda-Terminal für die Ausführung von allen mit LEU zu realisierenden Anwendungen geeignet sein muß. Die Möglichkeit, eine Oberfläche zu entwickeln, die dieser Anforderung gerecht wird, setzt voraus, daß die benötigte Funktionalität, die das Agenda-Terminal zur Verfügung stellen muß, für die verschiedenen Anwendungen dieselbe ist. Bei abstrahierender Betrachtung bestehen alle mit LEU realisierbaren Anwendungen aus Geschäftsprozessen und ihren Aktivitäten. Das Agenda-Terminal arbeitet also auf den abstrakten Datentypen *"Geschäftsprozeß"* und *"Aktivität"*. An seiner Oberfläche muß es deshalb alle für die Bearbeitung dieser beiden Objekttypen benötigte Funktionalität den Sachbearbeitern zur Verfügung stellen. In Abschnitt 5.1 wird ein Funktionalitätenkatalog zusammengestellt, der, unabhängig von der Art der mit LEU modellierten und ausgeführten Anwendung, den Anwendungsbenutzern alle benötigte Funktionalität anbietet. Wenn des weiteren in Betracht gezogen wird, daß die Zielgruppe der LEU-Ausführungswerkzeuge, also die Sachbearbeiter in den Unternehmen, alle ungefähr dieselben Erfahrungen im Arbeiten mit Computern haben, so scheint es möglich zu sein ein Werkzeug zu entwickeln, das von der Oberflächengestaltung und der angebotenen Funktionalität für die Ausführung aller LEU-Anwendungen geeignet

ist.

- **Keine Datenbankzugriffe**
  Eine weitere wichtige Entwurfsentscheidung war, daß im Agenda-Terminal keine Datenbankzugriffe stattfinden dürfen. Diese Entscheidung hatte weitreichende Konsequenzen für die Realisierung, da alle an der Oberfläche anzuzeigenen Daten in Form von Nachrichten über eine Interprozeß-Kommunikation in die Datenhaltung des Werkzeugs einfließen müssen. Diese Entscheidung birgt den großen Vorteil, daß die Benutzung des Agenda-Terminals ortsunabhängig bleibt. Das heißt, daß man sein Agenda-Terminal beispielsweise in Dortmund starten und von dort an der Bearbeitung von in Bochum laufenden Geschäftsprozessen teilnehmen kann. Die Unabhängigkeit von lokalen Datenbanken ermöglicht einen ortsunabhängigen Einsatz des Werkzeugs.

4.2.2 Die Benutzeroberfläche

Aufgrund der Tatsache, daß einige der späteren Benutzer des Agenda-Terminals zur Zeit der Durchführung dieser Arbeit als Berater im Hause der Firma LION tätig waren, konnte die Oberfläche des Agenda-Terminals nach einem Prototyping-Verfahren unter häufiger Rücksprache mit ihnen entwickelt werden. Wegen dieser besonderen Umstände wird die Entwicklung der Oberfläche in einem eigenen Kapitel (Kapitel 5) beschrieben. Abschnitt 5.2.4 enthält auch die Endversion der erstellten Oberfläche.

4.2.3 Einbettung in die LEU-Ausführungskomponente

Das Agenda-Terminal wurde im Rahmen dieser Diplomarbeit implementiert und in die Ausführungskomponente des Geschäftsprozeß-Managementsystems LEU integriert. Wie in Abbildung 2.6 auf Seite 25 zu sehen, ist das Agenda-Terminal sowohl Client in der LEU-Controllerstruktur als auch in der Agenda-Controllerstruktur.

Innerhalb der LEU-Controllerstruktur kommuniziert das Agenda-Terminal über den LEU-Server (in Abbildung 2.6 auf Seite 25 an der Beschriftung des Kommunikationspfades mit "LS" zu erkennen) mit dem LEU-Controller. Von ihm wird es mit Informationen zu allen laufenden Agenda-Controllerstrukturen (Geschäftsprozessen) versorgt, die es benötigt um sich bei den Agenden-Controllern der Geschäftsprozesse anzumelden. Desweiteren wird das Starten, Neustarten, Beenden und Anhalten von Geschäftsprozessen über die Interprozeß-Kommunikation mit dem LEU-Controller realisiert.
Mit den Agenden-Controllern der LEU-Ausführungskomponente kommunizieren die Agenda-Terminals über die Agenda-Server. Über diese Server werden alle Aktivitäten, zwischen den Agenda-Terminals und den Agenden-Controllern übermittelt. Im folgenden werden die zum Datenaustausch zwischen dem Agenda-Terminal und dem LEU-Controller bzw. dem Agenden-Controller spezifizierten Nachrichten kurz beschrieben. In diesem Zusammenhang werden nur die "NORMAL-MESSAGES" beschrieben, die zum Datenaustausch zwischen den Prozessen dienen. Auf die "CONTROL-" und "SYSTEM-MESSAGES", die den Auf- und Abbau der Kommunikationsstruktur innerhalb der Controllerstrukturen realisieren, soll hier nicht weiter eingegangen werden. Für die "NORMAL-MESSAGES" wurde mit Hilfe der erweiterten BNF (EBNF) ein Protokoll spezifiziert (siehe Anhang B).

Nachrichtenaustausch mit dem LEU-Controller

- **KSB_LC_TI_START_MESSAGE** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, sobald sich dieses nach der Beendigung seiner Initialisierungsphase mit der Nachricht *KSB_TI_LC_LACB_START* beim LEU-Controller angemeldet hat. Mit dieser Mitteilung informiert der LEU-Controller das Terminal über alle ausführbaren Geschäftsprozeßmodelle, die aus dem Agenda-Terminal heraus gestartet werden können. Die Geschäftsprozeßmodelle werden in die Datenhaltung des Agenda-Terminals eingetragen und an der Oberfläche des Werkzeugs angezeigt.
- **KSB_LC_TI_NEW_MODEL** (LEU-Contr. → Terminal)
  Diese Nachricht wird vom LEU-Controller an das Agenda-Terminal übermittelt,

sobald ein neues ausführbares Geschäftsprozeßmodell existiert, zu dem ein neuer Geschäftsprozeß aus dem Agenda-Terminal heraus gestartet werden kann. Das Geschäftsprozeßmodell wird in die Datenhaltung des Agenda-Terminals eingefügt und die Datenstruktur der Oberfläche aktualisiert.

- **KSB_LC_TI_SUSPENDED_PROCESSES** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, sobald sich dieses, nach der Beendigung seiner Initialisierungsphase, mit der Nachricht *KSB_TI_LC_LACB_START* beim LEU-Controller angemeldet hat. Mit dieser Mitteilung informiert der LEU-Controller das Terminal über alle angehaltenen Geschäftsprozesse, die aus dem Agenda-Terminal erneut gestartet werden können. Die Geschäftsprozesse werden in die Datenhaltung des Agenda-Terminals eingetragen und an der Oberfläche des Werkzeugs angezeigt.

- **KSB_LC_TI_NEW_SUSPENDED_PROCESS** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, sobald ein weiterer Geschäftsprozeß angehalten wurde und erneut gestartet werden kann. Der Geschäftsprozeß wird in die Datenhaltung des Agenda-Terminals eingefügt und die Datenstruktur der Oberfläche aktualisiert.

- **KSB_LC_TI_DELETE_SUSPENDED_PROCESS** (LEU-Contr. → Terminal)
  Diese Nachricht wird vom LEU-Controller an das Agenda-Terminal übermittelt, sobald ein angehaltener Geschäftsprozeß erneut zur Ausführung gebracht wurde. Der Geschäftsprozeß wird aus der Datenhaltung des Agenda-Terminals entfernt und die Datenstruktur der Oberfläche aktualisiert.

- **KSB_LC_TI_MODEL_LOCKED** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, nachdem der Anwendungsbenutzer versucht hat, zu einem derzeit gesperrten Geschäftsprozeßmodell einen neuen Geschäftsprozeß zu starten. Nach dem Erhalt dieser Nachricht wird im Agenda-Terminal eine Info-Box geöffnet, mit der der Anwendungsbenutzer über den Sachverhalt informiert wird.

- **KSB_LC_TI_NO_FURTHER_INC** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, nachdem der Anwendungsbenutzer versucht hat, zu einem Geschäftsprozeßmodell, zu dem sich immer nur ein Geschäftsprozeß in Ausführung befinden darf, einen zweiten Geschäftsprozeß zu starten. Nach dem Erhalt dieser Nachricht wird im Agenda-Terminal eine Info-Box geöffnet, mit welcher der Anwendungsbenutzer über den Sachverhalt informiert wird.

- **KSB_LC_TI_NO_FREE_SERVICE** (LEU-Contr. → Terminal)
  Diese Mitteilung wird vom LEU-Controller an das Agenda-Terminal übermittelt, nachdem der Anwendungsbenutzer versucht hat, einen neuen Geschäftsprozeß zu

starten, aber installationsbedingt kein freies IPC-Service für die neue Agenda-Controllerstruktur mehr existiert. Nach dem Erhalt dieser Nachricht wird im Agenda-Terminal eine Info-Box geöffnet, mit der der Anwendungsbenutzer über den Sachverhalt informiert wird.

- **KSB_TI_LC_TERMINAL_HELLO** (Terminal → LEU-Contr.)
  Mit dieser Mitteilung meldet sich ein neues Agenda-Terminal beim LEU-Controller an. Nach dem Erhalt dieser Nachricht übermittelt der LEU-Controller alle Geschäftsprozeßmodelle, zu denen neue Geschäftsprozesse aus dem Agenda-Terminal heraus gestartet werden können und alle angehaltenen Geschäftsprozesse, die erneut zur Ausführung gebracht werden können, mit Hilfe der Nachrichten *KSB_LC_TI_START_MESSAGE* und *KSB_LC_TI_SUSPENDED_PROCESSES* an das Agenda-Terminal zurück.
- **KSB_TI_LC_START_INC** (Terminal → LEU-Contr.)
  Wird vom Anwendungsbenutzer an der Oberfläche des Agenda-Terminals ein neuer Geschäftsprozeß gestartet, so werden mit dieser Mitteilung alle Informationen zum zugehörigen Geschäftsprozeßmodell an den LEU-Controller übermittelt. Der LEU-Controller startet nach dem Erhalt dieser Nachricht einen neuen Agenden-Controller. Das weitere Vorgehen zum Starten eines neuen Geschäftsprozesses wurde in Abschnitt 3.2.2 unter →*Starten eines neuen Geschäftsprozesses* ausführlich beschrieben.
- **KSB_TI_LC_RESTART_PROCESS** (Terminal → LEU-Contr.)
  Wird vom Anwendungsbenutzer an der Oberfläche des Agenda-Terminals ein angehaltener Geschäftsprozeß erneut gestartet, so werden mit dieser Mitteilung alle Informationen über den Geschäftsprozeß an den LEU-Controller übermittelt. Das erneute Starten eines angehaltenen Geschäftsprozesses wurde in Kapitel 3.2.2 unter → *Erneuter Start eines angehaltenen Geschäftsprozesses* ausführlich beschrieben.

Nachrichtenaustausch mit dem Agenden-Controller

- **KSB_ACB_TI_LOGIN_ANSWER** (Agenden-Contr. → Terminal)
  Nachdem sich ein neuer Benutzer mit seinem Berechtigungsprofil beim Agenden-Controller angemeldet hat, überprüft dieser, ob der Benutzer zur Mitarbeit im Geschäftsprozeß berechtigt ist und teilt ihm das Ergebnis mit dieser Mitteilung mit. Wenn der Benutzer zur Mitarbeit in dem Geschäftsprozeß berechtigt ist, wird der Geschäftsprozeß in die Datenhaltung des Agenda-Terminals eingetragen und die Datenstruktur der Oberfläche aktualisiert.
- **KSB_ACB_TI_AGENCY** (Agenden-Contr. → Terminal)
  Mit dieser Nachricht übermittelt der Agenden-Controller dem Agenda-Terminal eine aktivierbare nicht prädikatbehaftete Aktivität (→Instanzen). Die Aktivität wird in die Datenhaltung des Agenda-Terminals eingetragen und die Datenstruktur der Oberfläche zum Anzeigen der ausführbaren Aktivitäten aktualisiert.

- **KSB_ACB_TI_PRED_AGENCY** (Agenden-Contr. → Terminal)
Mit dieser Mitteilung übermittelt der Agenden-Controller dem Agenda-Terminal eine aktivierbare prädikatbehaftete Aktivität (→Instanzen). Die Aktivität wird in die Datenhaltung des Agenda-Terminals eingetragen und die Datenstruktur der Oberfläche zum Anzeigen der ausführbaren Aktivitäten aktualisiert.

- **KSB_ACB_TI_DELETE_AGENCY** (Agenden-Contr. → Terminal)
Mit dieser Nachricht übermittelt der Agenden-Controller dem Agenda-Terminal eine nicht mehr aktivierbare Aktivität. Die Aktivität wird aus der Datenhaltung des Agenda-Terminals gelöscht und die Datenstruktur der Oberfläche zum Anzeigen der ausführbaren Aktivitäten aktualisiert.

- **KSB_ACB_TI_AGENCY_DONE** (Agenden-Contr. → Terminal)
Mit dieser Mitteilung unterrichtet der Agenden-Controller ein Agenda-Terminal über die Beendigung einer vom Benutzer bearbeiteten Aktivität. Sie kann aus der Datenhaltung des Agenda-Terminals gelöscht werden. Die Oberfläche zum Anzeigen der in Bearbeitung befindlichen Aktivitäten wird danach aktualisiert.

- **KSB_ACB_TI_AGENCY_ALREADY_WORKING** (Agenden-Contr. → Terminal)
Ein Benutzer hat eine Aktivität zur Bearbeitung dem Agenden-Controller übermittelt. Die Aktivität wurde jedoch schon kurz vorher von einem anderen Benutzer zur Bearbeitung ausgewählt oder kann aus anderen Gründen zur Zeit nicht aktiviert werden. Dieser Sachverhalt wird vom Agenden-Controller dem Agenda-Terminal mit dieser Nachricht übermittelt. Nach dem Empfang dieser Mitteilung wird im Agenda-Terminal eine Info-Box geöffnet, die dem Anwendungsbenutzer diesen Sachverhalt anzeigt.

- **KSB_ACB_TI_NEW_INC** (Agenden-Contr. → Terminal)
Das Starten eines neuen Geschäftsprozesses ist komplett abgeschlossen. Die Agenda-Terminals werden nun mit dieser Nachricht vom Agenden-Controller über den neuen Geschäftsprozeß informiert. Das Agenda-Terminal meldet sich hierauf über die Nachricht *KSB_TI_ACB_NEW_PROCESS_LOGIN* mit den Rollen des Anwendungsbenutzers beim neuen Agenden-Controller an.

- **KSB_ACB_TI_END_OF_INC** (Agenden-Contr. → Terminal)
Mit dieser Mitteilung unterrichtet der Agenden-Controller die Agenda-Terminals über die Beendigung des Geschäftsprozesses. Der Prozeß wird aus der Datenhaltung des Terminals gelöscht und die Datenstruktur der Oberfläche aktualisiert.

- **KSB_TI_ACB_START_AGENCY** (Terminal → Agenden-Contr.)
Eine im Agenda-Terminal zur Bearbeitung ausgewählte nicht prädikatbehaftete Aktivität wird mit dieser Nachricht dem Agenden-Controller übermittelt. Die Aktivität wird in der internen Datenstruktur als *"in Bearbeitung befindlich"* markiert und die Datenstruktur der Oberfläche des Agenda-Terminals aktualisiert.

Das Starten einer Aktivität wurde in Kapitel 3.2.2 unter → *Auswahl einer Aktivität durch einen Benutzer* ausführlich beschrieben.

- **KSB_TI_ACB_START_PRED_AGENCY** (Terminal → Agenden-Contr.)
  Eine im Agenda-Terminal zur Bearbeitung ausgewählte prädikatbehaftete Aktivität wird mit dieser Nachricht dem Agenden-Controller übermittelt. Die Aktivität wird in der internen Datenstruktur als *"in Bearbeitung befindlich"* markiert und die Datenstruktur der Oberfläche des Agenda-Terminals aktualisiert. Das Starten einer Aktivität wurde in Kapitel 3.2.2 unter → *Auswahl einer Aktivität durch einen Benutzer* ausführlich beschrieben.

- **KSB_TI_ACB_RESET_AGENCY** (Terminal → Agenden-Contr.)
  Eine im Agenda-Terminal zum Zurücksetzen ausgewählte Aktivität wird mit dieser Mitteilung dem Agenden-Controller übermittelt. Die in der internen Datenhaltung des Agenda-Terminals als *"in Bearbeitung befindlich"* markierte Aktivität wird gelöscht und die Datenstruktur der Oberfläche aktualisiert. Das Zurücksetzen von Aktivitäten wurde schon in Kapitel 3.2.2 unter → *Zurücksetzen einer Aktivität durch einen Benutzer* ausführlich beschrieben.

- **KSB_TI_ACB_NEW_USER_LOGIN** (Terminal → Agenden-Contr.)
  Mit dieser Mitteilung loggt sich ein neuer Benutzer unter Angabe seines aktuellen Berechtigungsprofils in den Geschäftsprozeß ein. Das Einloggen eines Benutzers in einen Geschäftsprozeß wurde schon in Kapitel 3.2.2 unter → *Anmeldung eines neuen Sachbearbeiters* ausführlich beschrieben.

- **KSB_TI_ACB_NEW_PROCESS_LOGIN** (Terminal → Agenden-Contr.)
  Nachdem das Agenda-Terminal über die Existenz eines neuen Geschäftsprozesses informiert worden ist, meldet es sich mit dieser Nachricht beim Agenden-Controller des neuen Geschäftsprozesses an. Diese Mitteilung enthält die Rollen des Anwendungsbenutzers, die der Agenden-Controller für die Berechtigungsüberprüfungen benötigt. Das Einloggen in einen neuen Geschäftsprozeß wurde schon in Kapitel 3.2.2 unter → *Anmeldung eines neuen Sachbearbeiters* ausführlich beschrieben.

- **KSB_TI_ACB_END_TERM** (Terminal → Agenden-Contr.)
  Mit dieser Nachricht meldet sich ein Agenda-Terminal bei einem Agenden-Controller ab. Das passiert, wenn ein Sachbearbeiter sein Berechtigungsprofil verändert (siehe Kapitel 3.2.2, → *Ein Sachbearbeiter ändert sein Berechtigungsprofil*) oder wenn er sich aus dem Agenda-Terminal ausloggt.

- **KSB_TI_ACB_STOP_PROCESS** (Terminal → Agenden-Contr.)
  Wird im Agenda-Terminal ein Geschäftprozeß zum *Anhalten* ausgewählt, so wird dem zu diesem Geschäftsprozeß gehörenden Agenden-Controller diese Nachricht übermittelt. Das Anhalten eines Geschäftsprozesses wurde schon in Kapitel 3.2.2 unter →*Anhalten eines Geschäftsprozesses durch einen Benutzer* ausführlich beschrieben.

- **KSB_TI_ACB_SHUTDOWN** (Terminal → Agenden-Contr.)
  Wird im Agenda-Terminal ein Geschäftprozeß zum *Beenden* ausgewählt, so wird dem zu diesem Geschäftsprozeß gehörenden Agenden-Controller diese Nachricht übermittelt. Das Beenden eines Geschäftsprozesses wurde schon in Kapitel 3.2.2 unter →*Beenden eines Geschäftsprozesses durch einen Benutzer* ausführlich beschrieben.

- **KSB_TI_LACB_START** (Terminal → Agenden-Contr.)
  Diese Mitteilung wird direkt nach der Initialisierungsphase des Agenda-Terminals an die Agenden-Controller gesendet. Die Mitteilung enthält Informationen, die der Agenden-Controller an den LEU-Kommunikationsmonitor weiterreicht.

- **KSB_LACB_CLIENT_PING** (Agenden-Contr. → Terminal)
  Wird im LEU-Kommunikationsmonitor ein "Aktiv-Check" auf ein Agenda-Terminal gemacht, so leitet der Agenden-Controller diese Nachricht an das zu überprüfende Agenda-Terminal weiter. Das Agenda-Terminal antwortet mit der Nachricht KSB_CLIENT_LACB_PONG.

- **KSB_CLIENT_LACB_PONG** (Terminal → Agenden-Contr.)
  Diese Mitteilung schickt das Agenda-Terminal bei einem "Aktiv-Check" vom LEU-Kommunikationsmonitor an den Agenden-Controller zurück.

4.2.4 Die Modularisierung des Agenda-Terminals

Im folgenden wird die Modularisierung des Agenda-Terminals beschrieben. Die in diesem Abschnitt verwendeten Begriffe *Prozeß*, *Komponente*, *Modul* und *Funktion* verstehen sich hierbei wie in Abschnitt 3.2.3 erläutert.

4.2.4.1 Die Komponenten des LEU-Agenda-Terminals

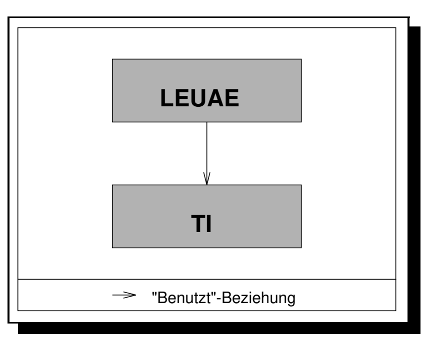


Abbildung 4.1: Komponenten des LEU-Agenda-Terminals

Wie in Abbildung 4.1 zu sehen, besteht das LEU-Agendaterminal aus den Komponenten **LEUAE** und **TI**. Die Komponente *LEUAE* besteht aus dem Modul *leuaedi*. Das Modul enthält die Main-Routine und die Exit-Funktion des Agenda-Terminals. Innerhalb dieses Moduls wird bei der Initialisierung die Berechtigungsüberprüfung zum Arbeiten mit dem Agenda-Terminal, das Starten des Oberflächendialogs und die Anmeldung als Client beim LEU-Controller und allen Agenden-Controllern vorgenommen. Die Main-Routine enthält die *"Eventloop"* der Oberfläche. Alle durch den Benutzer ausgelösten *Oberflächenevents* werden in eine Warteschlange geschrieben und dann in der *"Eventloop"* verarbeitet. Die Komponente *LEUAE* stützt sich hierzu auf die Komponente *TI* ab. Die Komponente *TI* realisiert die Datenhaltung des Agenda-Terminals. Sie enthält Funktionen zur Verarbeitung der *Oberflächenevents* und zur Kommunikation mit den anderen LEU-Ausführungswerkzeugen.

4.2.4.2 Die Komponente TI

Die Funktionalität der einzelnen Module des Agenda-Terminals wird im folgenden kurz beschrieben.

Die Module *TIin, TIaci* und *TIlcii* dienen dem Empfang und der Verarbeitung von IPC-Nachrichten. Sie bilden eine Datenkapsel für den Objekttyp *KSBme_Message*. Der abstrakte Datentyp *KSBme_Message* verkapselt die Nachrichten, die zwischen den LEU-Ausführungswerkzeugen mittels IPC ausgetauscht werden müssen.

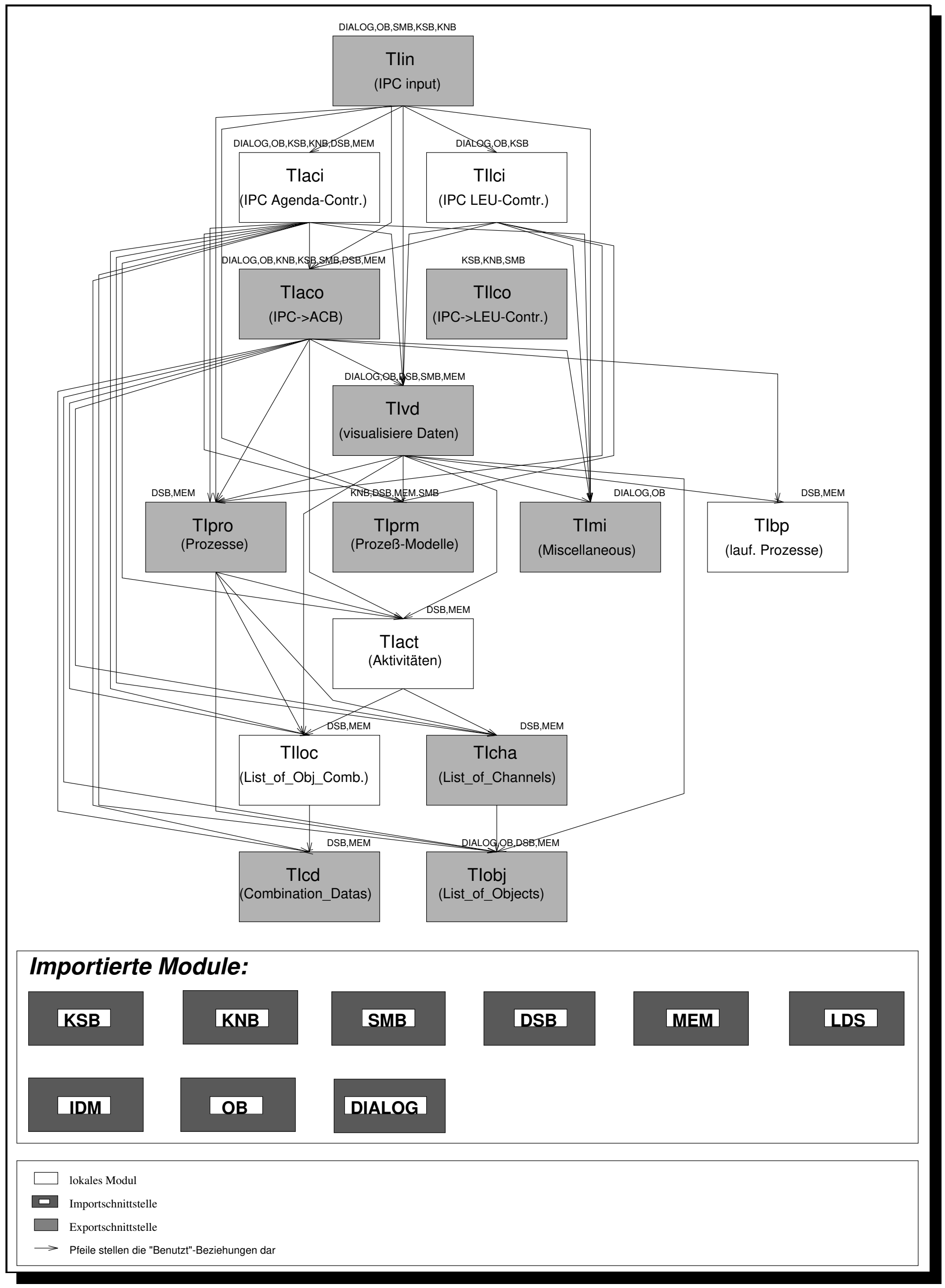


Abbildung 4.2: Modulhierarchie der Komponente TI

- **Modul TIin**
  Das Modul enthält den MessageHandler, der angesprungen wird, sobald am Socket eine neue Nachricht anliegt. Zusätzlich übernimmt dieses Modul durch die Funktion *TIin_Init* die komplette Initialisierung des Terminals. Mit der Funktion *TIin_Exit* wird der Speicher aller innerhalb der Komponente TI gespeicherten Daten freigegeben, und das Terminal loggt sich bei allen Agenden-Controllern aus. Der MessageHandler und die Funktionen zur Initialisierung und Terminierung der Komponente bilden die Exportschnittstelle des Moduls.

- **Modul TIaci**
  Das Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Agenden-Controller (ACB) an das Terminal (TI) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Nachricht übernimmt. Das Modul besitzt keine Exportschnittstelle, da alle Funktionen nur innerhalb des Moduls oder innerhalb der Komponente TI benutzt werden.

- **Modul TIlci**
  Das Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom LEU-Controller (LC) an das Terminal (TI) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, welche die Verarbeitung der Nachricht übernimmt. Das Modul besitzt keine Exportschnittstelle, da alle Funktionen nur innerhalb des Moduls oder innerhalb der Komponente TI benutzt werden.

Die Module *TIaco* und *TIlco* bilden ebenfalls eine Datenkapsel für den Objekttyp *KSBme_Message*. Sie enthalten die Verarbeitungsfunktionen, die Nachrichten an die anderen Clients der LEU-Ausführungswerkzeuge verschicken.

- **Modul TIaco**
  Das Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Terminal (TI) zum Agenden-Controller (ACB) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine

Funktion, die die Verarbeitung der Nachricht übernimmt. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TIlco**
  Das Modul enthält die Verarbeitungsfunktionen für die Nachrichten, die vom Terminal (TI) zum LEU-Controller (LC) übermittelt werden. Hierzu existiert für jeden vereinbarten Nachrichtentyp eine Funktion, die die Verarbeitung der Nachricht übernimmt. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TIvd**
  Dieses Modul ist ein Funktionsmodul und dient der Visualisierung der in den internen Datenstrukturen gespeicherten Daten an der Oberfläche. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

Die Module *TIpro, TIprm, TImi, TIbp, TIact, TIloc, TIcha, TIcd* und *TI-obj* bilden die Datenmodule der Komponente. In ihnen werden alle Daten gespeichert, die zur Visualisierung an der Oberfläche des Agenda-Terminals und zur Kommunikation mit den anderen Clients der LEU-Ausführungswerkzeuge benötigt werden. Die Module stellen alle Zugriffsoperationen, die zum Einfügen, Löschen, Auslesen und Modifizieren der Objekte des jeweiligen Datentyps benötigt werden, zur Verfügung.

- **Modul TIpro**
  Dieses Modul verkapselt den abstrakten Datentyp **Process**. Mit dem abstrakten Datentyp *Process* werden alle Informationen zu einem Geschäftsprozeß gespeichert, die zur Visualisierung an der Oberfläche und zur Kommunikation mit dem dazugehörenden Agenden-Controller benötigt werden. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TIprm**
  Dieses Modul verkapselt den abstrakten Datentyp **Process_Model**. Mit dem abstrakten Datentyp *Process_Model* werden alle Informationen zu einem Geschäftsprozeßmodell gespeichert, die zur Visualisierung an der Oberfläche benötigt werden. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TImi**
  Dieses Modul speichert verschiedene Daten, die sowohl im Regelwerk der Oberfläche, als auch im C-Code des Prozesses benötigt werden. Es stellt in seiner Exportschnittstelle Funktionen zum Setzen und Auslesen dieser Informationen zur Verfügung.

- **Modul TIbp**
  Dieses Modul verkapselt den abstrakten Datentyp **BusinessProcess**. Mit dem abstrakten Datentyp *BusinessProcess* werden alle Informationen über die laufenden Geschäftsprozesse gespeichert, die der Benutzer anhalten und herunterfahren darf und die zur Visualisierung an der Oberfläche benötigt werden. Das Modul besitzt keine Exportschnittstelle, da alle Funktionen nur innerhalb des Moduls oder innerhalb der Komponente TI benutzt werden.

- **Modul TIact**
  Dieses Modul verkapselt den abstrakten Datentyp **Activity**. Mit dem abstrakten Datentyp *Activity* werden alle Informationen über die in den Geschäftsprozessen zu bearbeitenden Aktivitäten gespeichert, die zur Visualisierung an der Oberfläche benötigt werden. Das Modul besitzt keine Exportschnittstelle, da alle Funktionen nur innerhalb des Moduls oder innerhalb der Komponente TI benutzt werden.

- **Modul TIloc**
  Dieses Modul verkapselt den abstrakten Datentyp **ObjectCombination**. Es wird benötigt, um Objektkombinationen von prädikatbehafteten Aktivitäten zu speichern. Es speichert nur die direkt zur Kombination gehörenden Daten. Die Objekte, mit denen die zugehörige Aktivität gestartet werden kann und die zu einer Kombination gehören,

werden im Modul TIcd abgelegt. Das Modul besitzt keine Exportschnittstelle, da alle Funktionen nur innerhalb des Moduls oder innerhalb der Komponente TI benutzt werden.

- **Modul TIcha**
  Dieses Modul verkapselt den abstrakten Datentyp **Channel**. Es speichert die Daten der im Vorbereich einer Aktivität stehenden Kanäle. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TIcd**
  Dieses Modul verkapselt den abstrakten Datentyp **Combination_Datas**. Hier werden die Informationen, die zu einer Objektkombination einer prädikatbehafteten Aktivität gehören, gespeichert. Es stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

- **Modul TIobj**
  Dieses Modul verkapselt den abstrakten Datentyp **Object**. Hier werden die Informationen zu den Objekten, die in den Vorbereichskanälen der Aktivitäten liegen, gespeichert. Das Modul stellt in seiner Exportschnittstelle Funktionen zur Verwendung im Oberflächendialog des Agenda-Terminals zur Verfügung.

4.2.4.3 Die Importschnittstelle

In der Importschnittstelle[1] des Agenda-Terminals liegen die Module **DIALOG**, **DSB**, **IDM**, **KNB**, **KSB**, **LDS**, **MEM**, **OB** und **SMB**. Sie stellen folgende Funktionalität zur Verfügung.

- **Die Komponenten DIALOG, IDM, LDS und OB**
  Sie stellen Funktionen für den Oberflächendialog zur Verfügung.

- **Die Komponente DSB**
  Die Komponente DSB ist ein Datenstrukturen-Basismodul. Es stellt Datenstrukturen und Zugriffsfunktionen zur Verfügung, auf die sich das Agenda-Terminal abstützt.

[1]Die Module der Importschnittstelle wurden nicht innerhalb dieser Diplomarbeit entwickelt.

- **Die Komponenten KNB und KSB**
  Sie stellen Funktionen für die IPC-Kommunikation zur Verfügung.
- **Die Komponente MEM**
  In der Komponente MEM sind alle Speicherverwaltungs-Funktionen gekapselt.
- **Die Komponente SMB**
  Sie stellt Funktionen für Berechtigungsüberprüfungen zur Verfügung.

Kapitel 5

Die Benutzeroberfläche des Agenda-Terminals

Innerhalb dieser Arbeit wurde auch die Benutzeroberfläche des LEU-Agenda-Terminals entwickelt und in das LEU-System integriert. Diese Oberfläche sollte zum einen allgemeingültig, das heißt für alle erstellbaren LEU-Anwendungen verwendbar sein, und zum anderen alle für die Anwendungsbenutzer notwendige Funktionalität enthalten. In diesem Kapitel wird der Entwicklungsprozeß der Oberfläche näher erläutert. In Abschnitt 5.2.2 wird ein früher Prototyp, der die Diskussionsgrundlage für weitere Änderungen bildete, beschrieben. Abschnitt 5.2.4 enthält die Beschreibung der Endversion der Oberfläche des Agenda-Terminals.

5.1 Ergonomie in der Benutzerführung

Um einen Überblick über die notwendige Funktionalität und die Gestaltungsmöglichkeiten der Oberfläche im allgemeinen zu erhalten, wurde zunächst der Kontakt zu Firmen gesucht, die Systeme zur Geschäftsprozeß-Ausführung anbieten. Leider gestaltete sich die Gewinnung brauchbarer Informationen, zumindest was die Oberflächengestaltung anbelangt, schwieriger als erwartet. In vielen Fällen war nicht mehr als die Standardprospekte zur Produktinformation zu bekommen. Jedoch konnten oft auch aus ihnen Informationen über die angebotene Funktionalität gewonnen werden.

Weitere wichtige Anregungen über die anzubietene Funktionalität konnte ich beim Besuch eines Seminares bei *Hewlett-Packard* zum Thema *"Dokumentmanagement- und Workflow-Konzepte als wichtige Instrumente für Business-Process Re-Engineering"* sammeln. In diesem Seminar wurde unter anderem auch eine kurze Marktübersicht über bestehende Geschäftsprozeß-Management-Systeme gegeben. Die Nachbetrachtung der Seminarunterlagen [HP94] hatte für mich ebenfalls hohen Informationswert.

Eindrücke über die Oberflächengestaltung konnte ich dann auf der *CEBIT*-Messe '94 in Hannover gewinnen, die ich zu diesem Zweck für drei Tage besucht habe. Hierbei konnte ich feststellen, daß mehrere Anbieter für jede mit ihrem Produkt erstellte Anwendung eine eigene Oberfläche entwickeln mußten.

Tabelle 5.1 gibt eine Übersicht über die im Rahmen dieser unvollständigen Marktübersicht betrachteten Produkte.

Hersteller	**Produkt**
AIC	STAFFWARE
Bull	FlowPATH [DK94]
COI	BusinessFlow
DIALOGIKA	multiDESK [DK94]
FileNet	WorkFlo / Visual Workflo [DK94]
IABG	ProMInanD [DK94]
IBM	FlowMark [DK94]
Ley	COSA [DK94]
VW-GEDAS	synergie für windows [DK94]

Tabelle 5.1: Andere Geschäftsprozeß-Management-Systeme

Auf dieser, auf den oben genannten Quellen basierenden, Wissensgrundlage kam folgender Funktionalitätenkatalog zustande:

- **Starten von Geschäftsprozessen**
  In vielen Anwendungsgebieten lösen eingegangener Schriftverkehr oder Anrufe von Kunden den Start neuer Geschäftsprozesse aus. Diese Geschäftsprozesse müssen von den Anwendungsbenutzern aus dem Agenda-Terminal heraus gestartet werden können. Welcher Anwendungsbenutzer welche Geschäftsprozesse starten darf, kann im LEU-Berechtigungssystem festgelegt werden.

- **Beenden und Anhalten laufender Geschäftsprozesse**
  Auch das Beenden und Anhalten von laufenden Geschäftsprozessen geschieht unter Berücksichtigung der jeweiligen Benutzerberechtigungen im Agenda-Terminal. Die Sachbearbeiter haben damit die Möglichkeit, nicht mehr benötigte Geschäftsprozesse zu beenden oder laufende Geschäftsprozesse, zum Beispiel zur Datensicherung, kurzzeitig herunterzufahren.

- **Neustart der angehaltenen Geschäftsprozesse**
  Heruntergefahrene Geschäftsprozesse können im Agenda-Terminal erneut aktiviert werden. Die angehaltenen Geschäftsprozesse können danach an dem Punkt weiterbearbeitet werden, an dem sie heruntergefahren wurden.

- **Anzeigen von Aktivitäten**
  Die Aktivitäten werden im Agenda-Terminal, nach Geschäftsprozessen geordnet, den Anwendungsbenutzern in ihrer persönlichen Agenda angezeigt.

- **Weiterleiten von Aktivitäten**
  Während der Berechtigungsmodellierung wurden alle Aktivitäten mit Rollen verbunden. Die Aktivitäten können zur Ausführungszeit von allen Anwendungsbenutzern bearbeitet werden, die zumindest eine der Rollen besitzen, mit denen die Aktivität verbunden ist. Das Weiterleiten von Aktivitäten würde ein Umgehen der modellierten Benutzerberechtigungen bedeuten und ist daher nicht realisiert worden.

- **Anzeigen aller in Bearbeitung befindlicher Aktivitäten**
  Damit alle Benutzer zu jeder Zeit eine Übersicht über die von Ihnen derzeit bearbeiteten Aktivitäten bekommen können, wurde dieser Punkt im Agenda-Terminal realisiert.

- **Zurückstellen von in Bearbeitung befindlichen Aktivitäten**
  Wurde eine Aktivität fälschlicherweise ausgewählt, oder soll sie aus anderen Gründen erst später bzw. besser von anderen Anwendungsbenutzern bearbeitet werden, so können bereits ausgewählte Aktivitäten zurückgestellt werden.

- **Sortieren der ausführbaren Aktivitäten nach verschiedenen Kriterien**
  Als Sortierkriterien für die Aktivitäten wurden in anderen Systemen eine Priorisierung und die alphabetische Anordnung verwendet. Im Agenda-Terminal kann neben der alphabetischen Anordnung nach "Eingangsdatum" (→ Priorisierung von Aktivitäten) sortiert werden.

- **Objektabhängige Aktivitätenauswahl**
  Eine wichtige Funktionalität ist die objektabhängige Aktivitätenauswahl. Durch die Verwendung ungeeigneter Modellierungssprachen fehlt diese Funktionalität in vielen anderen Geschäftsprozeß-Management-Systemen. Durch die objektabhängige Aktivitätenauswahl erhalten die Anwendungsbenutzer die Möglichkeit, schon vor der Auswahl einer Aktivität, die durch die Abarbeitung dieser Aktivität zu verändernden Datenobjekte anzusehen und auszuwählen und somit die Ausführung der Aktivitäten gezielt mit den von ihnen ausgewählten Objekten zu starten.

- **Berücksichtigung von Urlaubs- und Krankheitsvertretungen**
  Diese Funktionalität wird durch das LEU-Berechtigungssytem gegeben. Soll ein Mitarbeiter die Tätigkeiten eines Kollegen für eine gewisse Zeit mit übernehmen, so werden ihm für diese Zeit die Rollen seines Kollegen zugewiesen.

- **Priorisierung von Aktivitäten**
  Zunächst wurden alle ausführbaren Aktivitäten in Prioritätsklassen

eingeteilt. Da diese Einteilung jedoch nur in Abhängigkeit von dem Zeitpunkt, seit dem die Aktivität ausführbar ist, vorgenommen wurde, wurde die Unterteilung in Prioritätsklassen verworfen und statt dessen zu jeder Aktivität das Datum, seit wann sie zur Bearbeitung ansteht, mit angezeigt.
Außerdem wurden bei der Implementierung schon einige Vorbereitungen getroffen, um Aktivitäten, die in der Animation der Geschäftsprozesse (siehe Abschnitt 2.5.8 – Der Geschäftsprozeß-Monitor) als zeitkritisch beobachtet werden konnten, direkt zur Ausführung zu bringen. Hierzu muß die LEU-Animationskomponente dahingehend erweitert werden, daß sie dem zugehörigen Geschäftsprozeß-Interpreter die zeitkritischen Aktivitäten übermitteln kann. Die Aktivitäten werden dann, unter Berücksichtigung der Benutzerberechtigungen, in die Agenden der Anwendungsbenutzer geschrieben und sperren die Auswahl aller anderen in dem Geschäftsprozeß anstehenden Aktivitäten.

- **Einschränkung und Erweiterung von Benutzerberechtigungen**
  Der Benutzer kann im Agenda-Terminal seine aktuelle Rollenmenge, innerhalb der ihm zugewiesenen Rollen, einschränken oder ausweiten.

- **Anzeige der Bearbeitungsstände laufender Geschäftsprozesse**
  Für diese Funktionalität existiert in LEU eine eigene Komponente, die die Animation von laufenden Geschäftsprozessen mit Hilfe einer graphischen Oberfläche visualisiert (siehe Abschnitt 2.5.8 – Der Geschäftsprozeß-Monitor).

Das Agenda-Terminal stellt alle benötigten, auf den Objekten der abstrakten Datentypen *"Geschäftsprozeß"* und *"Aktivität"* arbeitenden Operationen, den Anwendungsbenutzern der erstellten LEU-Anwendungen zur Verfügung. Die Operationen auf den Geschäftsprozessen unterliegen hierbei den für die Sachbearbeiter individuell modellierten Berechtigungen, die im Agenda-Terminal Berücksichtigung finden müssen.

5.2 Entwicklung im Prototyping-Verfahren

Die Oberfläche des LEU Agenda-Terminals wurde in einem Prototyping-Verfahren unter häufiger Rücksprache mit den WIS-Entwicklern, die bei der Firma LION LEU-Anwendungen für die Bau- und Wohnungswirtschaft modellieren, entwickelt.

In Abschnitt 5.2.1 wird der *ISA-Dialogmanager*, mit dem die Oberfläche des Agenda-Terminals entwickelt wurde, beschrieben. Hierbei werden auch die Dialogobjekte die im Oberflächendialog des Agenda-Terminals verwendet werden, kurz erläutert. In Abschnitt 5.2.2 wird ein früher Prototyp der Oberfläche vorgestellt. Der Vergleich dieses Prototypen mit der Endversion der Oberfläche, die in Abschnitt 5.2.4 vorgestellt wird, und die in Kapitel 5.2.3 vorgestellten Erkenntnisse aus den Rücksprachen mit den WIS-Beratern und -Entwicklern sollen die Schwierigkeiten und Probleme bei der Oberflächengestaltung verdeutlichen, die ohne den direkten Kontakt zu den Endanwendern möglicherweise überhaupt nicht erkannt worden wären. Dieses Problem wird leicht verständlich, wenn man sich vor Augen führt, daß ein Informatiker oder Software-Entwickler oft schon jahrelang jeden Tag mit dem Medium Computer arbeitet, sich in Multi-Window-Anwendungen "zuhause fühlt" und viele EDV-Fachbegriffen mittlerweile nicht mehr zu diesen rechnet, da man sie häufig im täglichen Sprachgebrauch verwendet. Das trifft selbstverständlich auf die Endanwender der LEU-Anwendungen, also auf die Sachbearbeiter in den Unternehmen, nicht zu. Um so begründete Fehler in der Oberfläche des LEU Agenda-Terminals möglichst zu vermeiden, sind mir die Besprechungen mit den WIS-Beratern eine große Hilfe gewesen.

5.2.1 Der ISA-Dialogmanager und seine Dialogobjekte

Die Oberfläche des Agenda-Terminals wurde mit dem *ISA-Dialogmanager* [ISA92] entwickelt. Der *ISA-Dialogmanager* bietet ein User Interface Management System, mit dem Benutzeroberflächen für Anwendungen auf Standardfenstersysteme wie *OSF/Motif*, *MS-Windows* und *Presentation Mana-*

ger einfach entwickelt und getestet werden können. Hierbei bleibt auf sämtlichen Plattformen das typische "Look and Feel" des jeweiligen Fenstersystems erhalten. Die mit dem *ISA-Dialogmanager* erstellten Oberflächen bestehen aus einem Definitions- und einem Regelteil. Der Definitionsteil bestimmt das Aussehen der Oberfläche. Im Regelteil werden die Regeln implementiert, die den Ablauf des Dialogs bestimmen. Der Regelteil der Oberfläche bildet gleichzeitig die Schnittstelle zum Programmcode, der in der Programmiersprache *C* erstellt wird. Die in der Exportschnittstelle der erstellten C-Module liegenden Funktionen können aus den Regeln der Oberfläche heraus aufgerufen werden. Im nun folgenden Abschnitt werden die im weiteren Text benutzten Dialogobjekte des *ISA-Dialogmanagers* kurz erläutert.

- **Dialogboxen**
  Dialogboxen sind Fenster, die die Bearbeitung (Selektion von Dialogobjekten) in allen anderen Fenstern sperren.

- **Edittext**
  Edittexte dienen der Eingabe und Darstellung von vom Benutzer editierbaren oder anzuzeigenden Texten. Sie können ein- und mehrzeilig sein.

- **Fenster**
  Innerhalb eines Fensters können weitere Dialogobjekte angelegt werden. Das Fenster dient somit als optischer sowie sinngemäßer Rahmen für andere Dialogobjekte.

- **Groupboxen**
  Groupboxen sind ein Hilfsmittel, um eine logische Struktur innerhalb von Dialogobjekten definieren zu können. Innerhalb einer Groupbox können logisch zusammengehörige Objekte zu einer Gruppe zusammengefaßt werden.

- **Listboxen**
  In eine Listbox können Informationen in Form von schriftlichen Einträgen untereinander aufgereiht und zur Auswahl (Selektion) bereitgestellt werden.

- **Menü**
  Menüs werden zur Dialogführung verwendet. Mit ihrer Hilfe können zum Beispiel Applikationen ausgeführt und Befehle aufgerufen werden.

- **Menüeintrag**
  Ein Menüeintrag ist ein Dialogobjekt innerhalb eines Menüs, das vom Benutzer angewählt werden kann.

- **Poptext**
  Poptexte dienen der Darstellung und Selektion von Aufzählungstypen. Im Primärfenster des Agenda-Terminals werden sie dazu benutzt, die Funktionalität des Fensters zu verändern.

- **Pushbuttons**
  Pushbuttons, im weiteren Text auch oft nur *Buttons* genannt, dienen dazu, bestimmte Aktionen innerhalb eines Dialogs zu bestätigen oder auszuführen.

- **Tablefields**
  Mit diesem Dialogobjekt können beliebig formatierte Listen in Form von Tabellen ausgegeben werden.

5.2.2 Ein früher Prototyp der Oberfläche

Die in den Abbildungen dieses Abschnittes dargestellten Fenster und Dialogboxen enthalten in ihrer Oberfläche oft keine Daten, da die zum damaligen Zeitpunkt erstellten Snapshots nur als Diskussionsgrundlage für die Besprechungen mit den WIS-Beratern dienten.

Abbildung 5.1 zeigt das Primärfenster des Agenda-Terminals. Im dargestellten Tablefield des Fensters werden die verschiedenen Aktivitäten angezeigt, die der Benutzer innerhalb der Geschäftsprozesse, in denen er zur Teilnahme berechtigt ist, ausführen darf. Die Aktivitäten sind den in der ersten Spalte des Tablefields auf gleicher Höhe befindlichen Geschäftsprozessen zugeordnet.

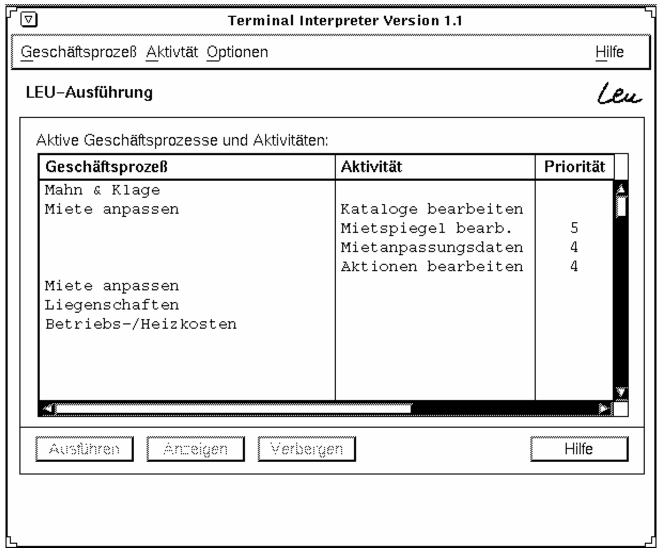


Abbildung 5.1: Primärfenster des Terminalprozesses

Die in Abbildung 5.2 dargestellten Fenster dienen zum Starten neuer Geschäftsprozesse. Im größeren Fenster werden alle Geschäftsprozeßmodelle angezeigt, für die der Benutzer einen neuen Geschäftsprozeß starten darf. Durch Selektion eines Geschäftsprozeßmodells und Anklicken des OK-Buttons mit der linken Maustaste wird das kleine Fenster geöffnet. Hier kann der Benutzer entscheiden, ob der neue Geschäftsprozeß mehrbenutzerfähig (Default) oder im Single-User-Mode gestartet werden soll. Die mehrbenutzerfähigen Geschäftsprozesse erscheinen nach ihrem Start in den Agenden aller zur Teilnahme an diesem Geschäftsprozeß berechtigten Benutzern. Ein im Single-User-Mode gestarteter Geschäftsprozeß erscheint nur in der Agenda des Anwendungsbenutzers, der ihn gestartet hat. Außerdem kann hier eine Kurzinformation zum neuen Geschäftsprozeß eingegeben werden, die den Benutzern im Primärfenster angezeigt werden kann. Als Kurzinformation eignen sich hierbei Informationen zum "warum" und "wozu" des neuen Geschäftsprozesses.

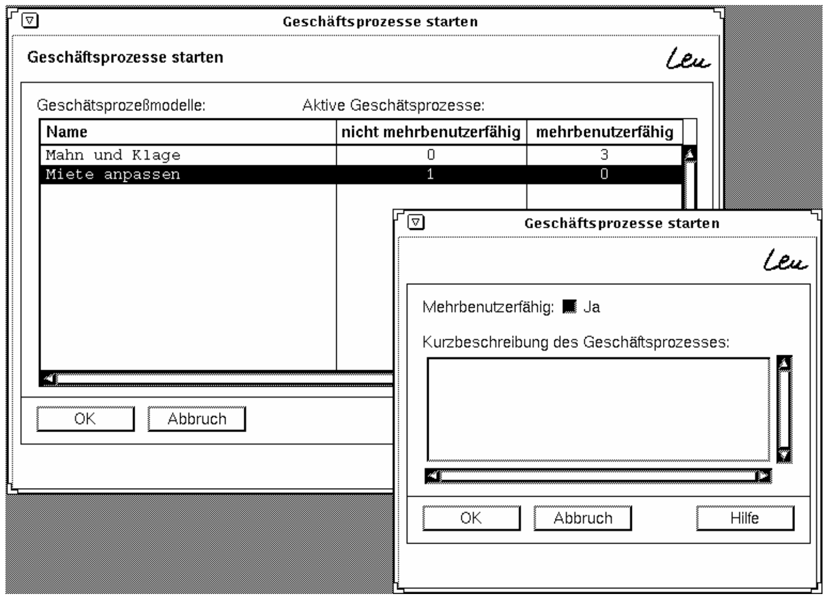


Abbildung 5.2: Dialogboxen zum Starten eines neuen Geschäftsprozesses

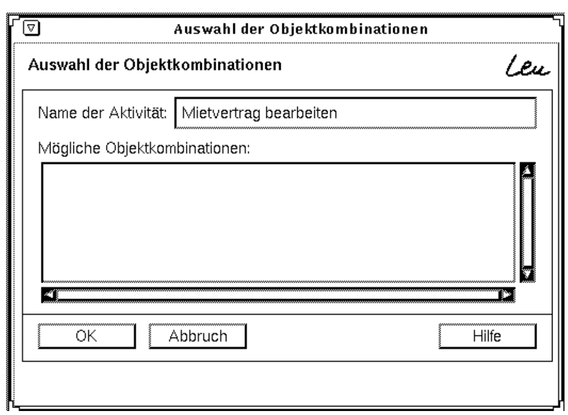


Abbildung 5.3: Dialogbox zur Objektauswahl bei prädikatbehafteten Instanzen

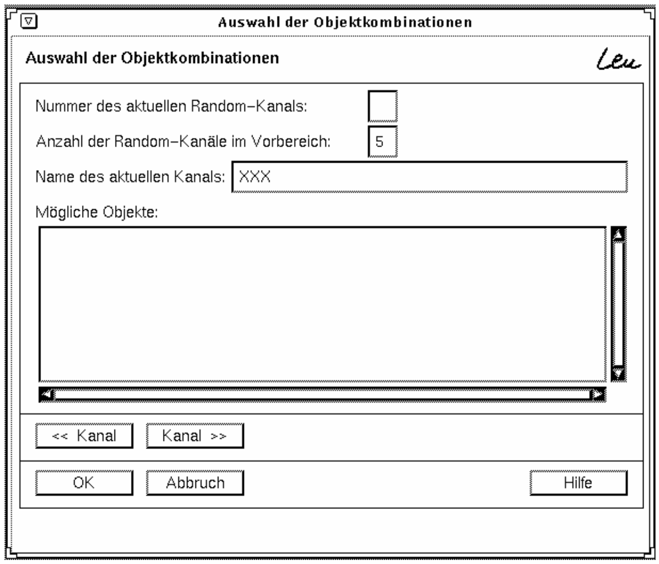


Abbildung 5.4: Dialogbox zur Objektauswahl bei nicht prädikatbehafteten Instanzen

In der Listbox des Fensters in Abbildung 5.3 werden die verschiedenen Objektkombinationen, mit denen eine Aktivität gestartet werden kann, dem Benutzer zur Auswahl angezeigt. Dieses Fenster wird geöffnet, wenn der Benutzer eine Aktivität, an die eine prädikatbehaftete Instanz (siehe $\xrightarrow{2.2}$Instanzen) angebunden ist, ausgewählt hat. Die zu einer Objektkombination dargestellten Daten entsprechen den Schlüsselfeldern (während der Datenmodellierung könnnen bestimmte Felder von Objekttypen als Schlüsselfelder gekennzeichnet werden) der im Vorbereich der Instanz liegenden Objekte.

Das Fenster in Abbildung 5.4 dient ebenfalls der Auswahl einer Objektkombination. Es wird geöffnet, wenn der Benutzer eine Aktivität, an die eine nicht prädikatbehaftete Instanz (siehe $\xrightarrow{2.2}$Instanzen) angebunden ist, ausgewählt hat. In diesem Fenster werden alle Objekte, die in einem Vorbereichskanal (siehe $\xrightarrow{2.2}$Kanäle) liegen, dem Benutzer zur Auswahl angezeigt. Existieren im Vorbereich der Instanz mehrere Kanäle, können diese durch Anklicken der Buttons [<< *Kanal*] oder [*Kanal* >>] mit der linken Maustaste oder durch die Eingabe des Kanalnamens im Feld < *Name des aktuellen Kanals* :> selektiert werden. In der Listbox werden immer die Objekte des aktuell ausgewählten Vorbereichskanals angezeigt. Wurde in jedem der Vorbereichskanäle ein Objekt ausgewählt, kann die Aktivität über den [*OK*]-Button gestartet werden. Über den Button [*Abbruch*] verläßt man die Dialogbox und gelangt wieder in das Primärfenster des Agenda-Terminals.

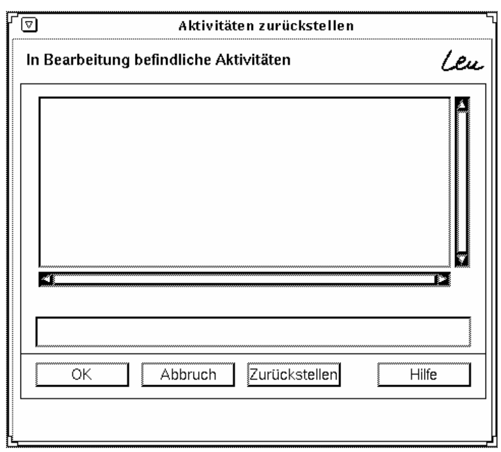


Abbildung 5.5: Dialogbox zum Zurückstellen von Aktivitäten

Die Dialogbox "*Aktivität zurückstellen*" in Abbildung 5.5 hat zwei Funktionalitäten. Zum einen ist es möglich, sich alle derzeit vom Benutzer bearbeiteten Aktivitäten anzeigen zu lassen, und zum anderen kann der Benutzer durch Auswahl einer Aktivität und Drücken des Buttons [*Zurückstellen*] die Bearbeitung dieser Aktivität abbrechen. Die Aktivität erscheint daraufhin wieder in den Agenden aller Benutzer, die die Berechtigung zu ihrer Bearbeitung haben.

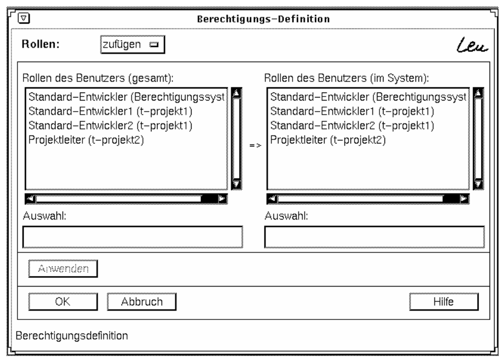


Abbildung 5.6: Dialogbox zum Ändern der Rollen

Mit dem *Berechtigungs-Definitions-Fenster* in Abbildung 5.6 hat der Benutzer die Möglichkeit, zur Laufzeit seine Berechtigungen zu verändern. In der linken Dialogbox werden immer alle Rollen, die dem Benutzer zugeordnet sind, angezeigt und in der rechten Dialogbox die Rollen, mit denen der Benutzer aktuell im System eingeloggt ist. Je nach Funktionalität, die über den Button [*zufügen/entfernen*] eingestellt werden kann, hat der Benutzer die Möglichkeit, seine Rollenmenge zu erweitern und so eventuell auch an weiteren Geschäftsprozessen teilnehmen zu können oder seine Rollenmenge einzuschränken und somit seine Teilnahme an einigen Geschäftsprozessen zu beenden.

In diesem Entwurf der Oberfläche waren die Funktionalitäten zum Herunterfahren und Stoppen laufender Geschäftsprozesse und das erneute Starten von heruntergefahrenen Geschäftsprozessen noch nicht enthalten. Im Gespräch mit den WIS-Beratern konnten einige Unzulänglichkeiten entdeckt und nach und nach beseitigt werden. Dieser Prozeß wird nun im Abschnitt 5.2.3 näher beschrieben.

5.2.3 Erkenntnisse aus dem WIS-Projekt

Bei der Firma **LION** wird derzeit mit LEU das Projekt *WIS*, ein Wohnungswirtschaftliches Informationssystem entwickelt. Dieses Projekt wird für die **WIS GmbH**, einem Zusammenschluß mehrerer großer Unternehmen aus dem Bereich der Bau- und Wohnungswirtschaft, von 20 Modellierern, 10 Programmierern und 80 externen Beratern aus der *WIS GmbH*, die das nötige Fachwissen mitbringen, entwickelt.

Die Entwicklung und Weiterentwicklung aller LEU-Komponenten konnte aus dem Umstand, einige Anwendungsentwickler und -benutzer direkt im Hause zu haben, stark profitieren. So konnte auch die Oberflächenentwicklung des Agenda-Terminals, von den positiven und negativen fachlichen Kritiken und Anregungen der *WIS*-Berater profitieren. Viele Dinge, deren Verwendung im Vorhinein problemlos zu sein schien, wie zum Beispiel das Oberflächenhandling mehrerer Fenster oder die Benutzung von Begriffen aus der FUNSOFT-Netz-Terminologie (siehe Abb. 5.4 - Kanal, Random-Kanal), erwiesen sich erst bei einem gemeinsamen Oberflächenreview und einer anschließenden Diskussion als kritisch und ungeschickt.

Viele der späteren Anwendungsbenutzer haben bisher noch nie am Computer oder höchstens auf DOS-Rechnern in "Single-Window-Anwendungen" gearbeitet. Da die Anwendungsbenutzer durch die innerhalb der Geschäftsprozesse angebundenen Dialoge ohnehin schon die Schwierigkeit haben werden, sich an Multi-Window-Anwendungen gewöhnen zu müssen, sollte zumindest das Agenda-Terminal mit möglichst wenigen Fenstern auskommen, die nur die tatsächlich benötigte Funktionalität enthalten.

Im Verlauf der Entwicklung weiterer Oberflächenprototypen fielen die (nicht dargestellten) Fenster zum benutzer- bzw. rollenbezogenem Weiterleiten von Aktivitäten heraus, da auf diese Weise die im LEU-Berechtigungssystem modellierten individuellen Benutzerberechtigungen umgangen und in Frage gestellt worden wären. Möchte der Benutzer während der Ausführung der

Geschäftsprozesse einige Aktivitäten auch durch andere, bisher nicht berechtigte Mitarbeiter ausführen lassen, so können für diese Mitarbeiter im Berechtigungssystem ihre Berechtigung auf die jeweiligen Aktivitäten ausgedehnt werden. Die Fenster zum Auswählen (Starten) von Aktivitäten, zum Starten von Geschäftsprozessen und zum Zurücksetzen von Aktivitäten wurden in ein Primärfenster integriert (siehe Abb. 5.8 auf Seite 98). Die Funktionalität des Primärfensters kann durch einen Poptext umgeschaltet werden. Später wurde auch das Anhalten und Beenden von Geschäftsprozessen, sowie der Neustart von angehaltenen Geschäftsprozessen ebenfalls in das Primärfenster des Agenda-Terminals integriert.

Im nächsten Abschnitt wird nun die Endversion der Oberfläche des LEU-Agendaterminals vorgestellt.

5.2.4 Die Endversion der Oberfläche

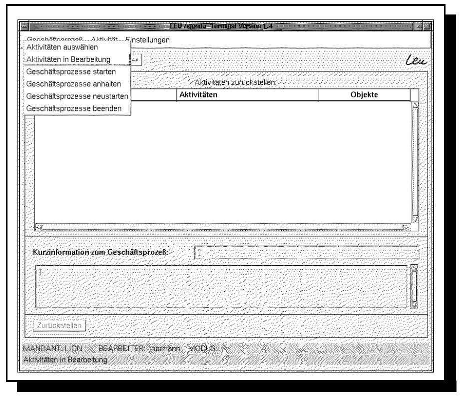
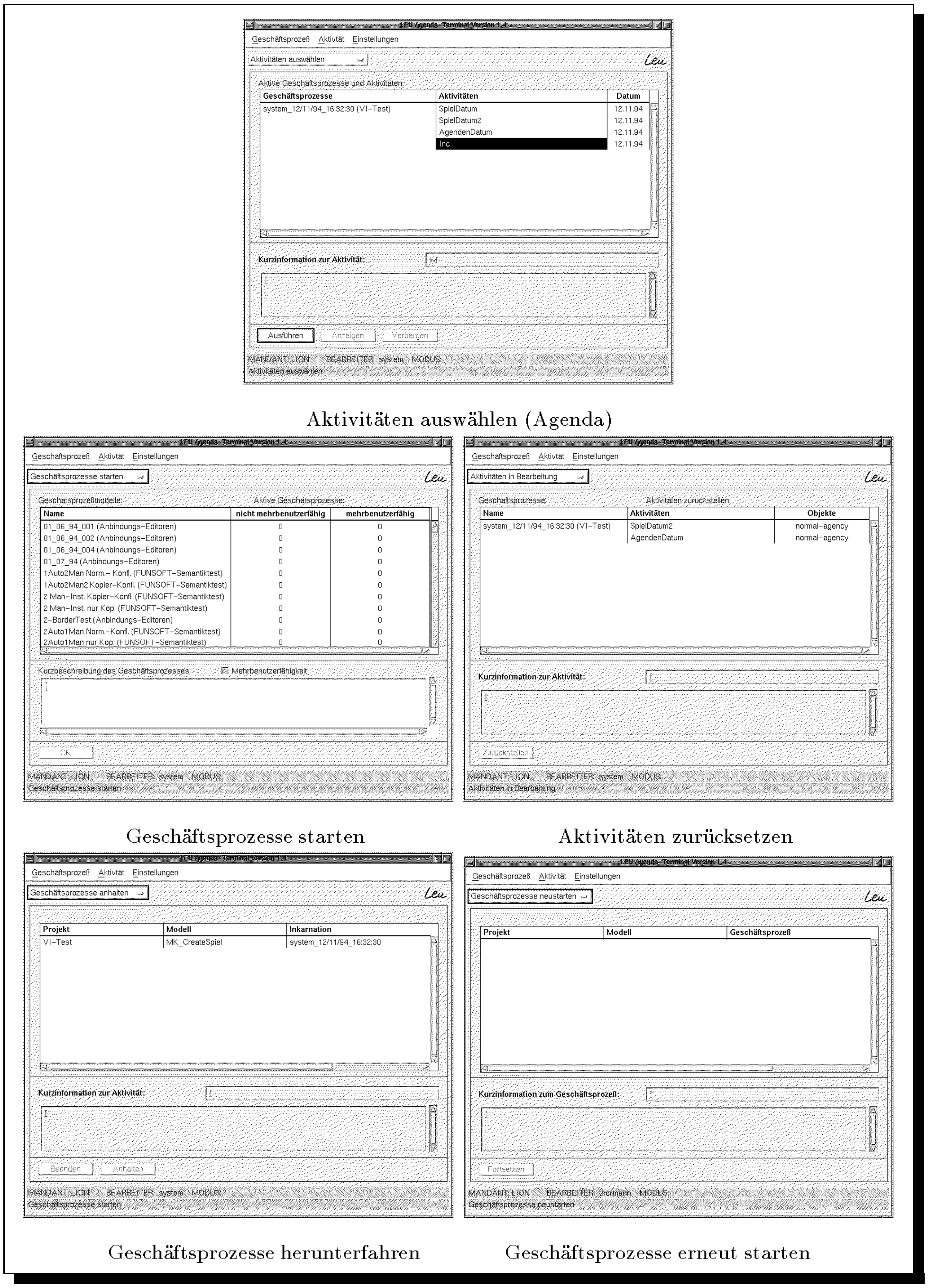


Abbildung 5.7: Endversion des Primärfensters vom Agenda-Terminal

Abbildung 5.7 zeigt die Endversion des Primärfensters vom LEU-Agenda-Terminal. Die in früheren Versionen noch auf verschiedene Fenster aufgeteilten Funktionalitäten, *Aktivitäten auswählen*, *Geschäftsprozesse starten*, *Aktivitäten zurücksetzen*, *Geschäftsprozesse herunterfahren / beenden* und *heruntergefahrene Geschäftsprozesse erneut starten* wurden in das Primärfenster integriert. Die verschiedenen Funktionalitäten können kontextsensitiv über einen Poptext (siehe Abb. 5.7) umgeschaltet werden. Beim Umschalten der Funktionalität werden die aktuell im Fenster angezeigten *Groupboxen* unsichtbar geschaltet und gegen andere ausgetauscht. Abbildung 5.8 visualisiert die verschiedenen Funktionalitäten des Agenda-Terminal-Primärfensters.

Steht die Funktionalität des Fensters auf *Aktivitäten auswählen*, so werden im Tablefield die verschiedenen Aktivitäten angezeigt, die der Benutzer innerhalb der Geschäftsprozesse, in denen er zur Teilnahme berechtigt ist, ausführen darf. Die Aktivitäten sind hierbei den in der ersten Spalte des Tablefields auf gleicher Höhe befindlichen Geschäftsprozessen zugeordnet. Die Geschäftsprozesse und die Aktivitäten bilden die selektierbaren Objekte des Fensters.



Aktivitäten auswählen (Agenda)

Geschäftsprozesse starten

Aktivitäten zurücksetzen

Geschäftsprozesse herunterfahren

Geschäftsprozesse erneut starten

Abbildung 5.8: Die verschiedenen Funktionalitäten des Agenda-Terminal-Primärfensters

Wird ein Geschäftsprozeß ausgewählt, so werden in dem mehrzeiligen *Edittext* unter der *Groupbox* des *Tablefields* dem Benutzer einige Informationen zum selektierten Geschäftsprozeß angezeigt. Hierzu zählen z.B. das Startdatum des Prozesses, der startende Benutzer und die beim Starten durch den Benutzer angegebene Kurzinformation zum Geschäftsprozeß. Über die *Pushbuttons Anzeigen* und *Verbergen* kann die Anzeige der zu dem selektierten Geschäftsprozeß gehörenden Aktivitäten unterbunden oder zugelassen werden.
Wird eine Aktivität ausgewählt, so wird eine zu dieser Aktivität hinterlegte Kurzinformation im mehrzeiligen Edittext des Fensters angezeigt, und die Ausführung der Aktivität kann durch Anklicken des *Pushbuttons Ausführen* gestartet werden.

Wurde die Funktionalität des Fensters auf *Geschäftsprozesse starten* eingestellt, so werden im *Tablefield* des Fensters alle Geschäftsprozeßmodelle, zu denen der jeweilige Benutzer einen neuen Geschäftsprozeß starten darf, angezeigt. In der zweiten und dritten Spalte des *Tablefields* wird dem Benutzer die Anzahl der zu diesem Geschäftsprozeßmodell laufenden Geschäftsprozesse im *Single-* und *Multi-User-Mode* angezeigt. Die selektierbaren Objekte sind hier die Geschäftsprozeßmodelle. Nach dem Auswählen eines Modells kann im unter dem *Tablefield* liegenden, mehrzeiligen *Edittext* eine Kurzinformation angegeben werden, die den Benutzern, die in dem Geschäftsprozeß mitarbeiten, angezeigt wird. Ein neuer Geschäftsprozeß wird durch Selektion eines Geschäftsprozeßmodells und anschließender Betätigung des *OK-Buttons* gestartet.

Steht die Funktionalität des Fensters auf *Aktivitäten zurücksetzen*, so werden im Tablefield des Primärfensters alle vom Benutzer in den verschiedenen Geschäftsprozessen derzeit bearbeiteten Aktivitäten angezeigt. Zu jeder Aktivität wird hierbei das zugehörige Geschäftsprozeßmodell und die Objekte mit denen sie gestartet worden ist, angezeigt. Durch Selektion einer Aktivität und anschließender Betätigung des *Zurückstellen-Buttons* kann die Bearbeitung der Aktivität abgebrochen werden. Der zugehörige Geschäftsprozeß wird in den Zustand vor Aktivierung der Aktivität zurückgesetzt, und die Aktivität erscheint danach wieder in den Agenden der für ihre Bearbeitung zuständigen Benutzer.

Wurde die Funktionalität des Fensters auf *Geschäftsprozesse herunterfahren* eingestellt, so werden im *Tablefield* des Fensters alle laufenden und durch den Benutzer des jeweiligen Agenda-Terminals gestarteten Geschäftsprozesse angezeigt. Die Geschäftsprozesse sind die selektierbaren Objekte des Fensters. Wird ein Geschäftsprozeß vom Benutzer ausgewählt, so werden ihm im unter dem *Tablefield* liegenden, mehrzeiligen *Edittext* Informationen zum selektierten Geschäftsprozeß angezeigt. Durch die Selektion der *Pushbuttons Beenden* oder *Anhalten* kann der ausgewählte Geschäftsprozeß beendet, das heißt heruntergefahren, und aus der Datenbank gelöscht oder angehalten, das heißt für kurze Zeit heruntergefahren werden.

Steht die Funktionalität des Fensters auf *Geschäftsprozesse erneut starten*, so werden im Tablefield des Primärfensters alle angehaltenen Geschäftsprozesse angezeigt, für die der jeweilige Benutzer die Berechtigung zum erneuten Starten hat. Nach der Selektion eines Geschäftsprozesses werden im unter dem *Tablefield* liegenden, mehrzeiligen *Edittext* Informationen zum selektierten Geschäftsprozeß angezeigt. Hierzu zählt beispielsweise, wann der Geschäftsprozeß angehalten worden ist. Durch die Selektion des *Pushbuttons Neustart* wird der selektierte Geschäftsprozeß erneut zur Ausführung gebracht.

Abbildung 5.9 zeigt die Endversionen der *Dialogboxen*, die beim Starten von Aktivitäten zur Selektion der Objekte dienen. Die linke *Dialogbox* dient der Auswahl einer Objektkombination bei prädikatbehafteten $\xrightarrow{2.2}$Instanzen. Jede in der *Listbox* der *Dialogbox* dargestellte Zeile entspricht einer möglichen Objektkombination, mit der die Ausführung der Aktivität gestartet werden kann. Durch Auswahl einer Objektkombination und Selektion des *OK-Buttons* wird die Ausführung der Aktivität gestartet.
Die rechte *Dialogbox* dient der Auswahl einer Objektkombination bei nicht prädikatbehafteten $\xrightarrow{2.2}$Instanzen. In der *Listbox* der *Dialogbox* werden alle

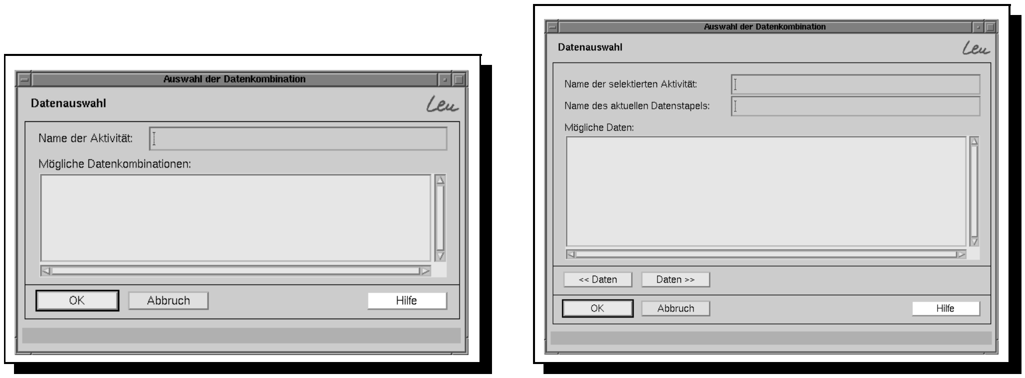


Abbildung 5.9: Endversionen der Dialogboxen zur Objektauswahl

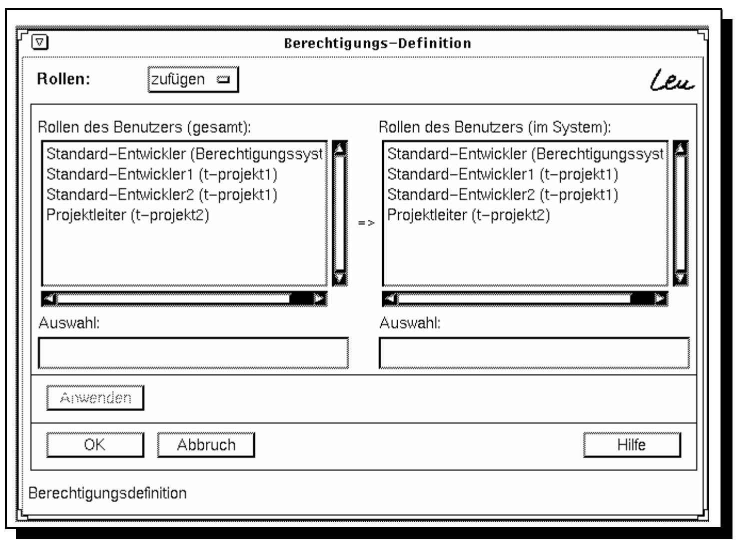


Abbildung 5.10: Endversion der Dialogbox zum Ändern der Rollen

Objekte, die im aktuell ausgewählten Vorbereichskanal ($\xrightarrow{2.2}$Kanäle) der Aktivität liegen, angezeigt. Durch Selektion der *Pushbuttons [<< Daten]* und *[Daten >>]* kann zwischen den verschiedenen Vorbereichskanälen ($\xrightarrow{2.2}$Kanäle) einer Aktivität gewechselt werden. Nachdem in jedem Vorbereichskanal ein Objekt ausgewählt worden ist, kann die Ausführung der Aktivität über den *OK-Button* gestartet werden.

Abbildung 5.10 zeigt die Endversion des *Berechtigungsdefinitions-Fensters*.

Mit diesem Fenster hat der Benutzer die Möglichkeit, zur Laufzeit seine Berechtigungen zu verändern. In der linken Dialogbox werden immer alle Rollen, die dem Benutzer zugeordnet sind, angezeigt und in der rechten Dialogbox die Rollen, mit denen der Benutzer aktuell im System eingeloggt ist. Je nach Funktionalität, die über den Button [$zufügen/entfernen$] eingestellt werden kann, hat der Benutzer die Möglichkeit, seine Rollenmenge zu erweitern, und so auch an weiteren Geschäftsprozessen teilnehmen zu können, oder seine Rollenmenge einzuschränken und somit seine Teilnahme an einigen Geschäftsprozessen zu beenden.

	Prototypen	**Endversion**
Anzahl der Fenster:	11	4
	*1 * Aktivitäten auswählen*	*in das Primärfenster integriert*
	*1 * Aktivitäten zurücksetzen*	*in das Primärfenster integriert*
	*2 * Geschäftsprozesse starten*	*in das Primärfenster integriert*
	*1 * Geschäftsprozesse herunterfahren*	*in das Primärfenster integriert*
	*1 * Geschäftsprozesse erneut starten*	*in das Primärfenster integriert*
	*2 * Aktivitäten weiterleiten*	*nicht mehr vorhanden*
	*2 * Objektauswahl*	*geblieben*
	*1 * Berechtigungs-Definitions-Fenster*	*geblieben*
gleichzeitig sichtbar:	3	2
	Geschäftsprozesse starten	*Objektauswahl*
		Berechtigungs-Definition

Tabelle 5.2: Gegenüberstellung Prototypen ↔ Endversion der Oberfläche

Die Oberflächen-Endversion besteht somit nur noch aus vier verschiedenen Fenstern und Dialogboxen, von denen maximal zwei Fenster gleichzeitig sichtbar sind. Bei den in den Fenstern verwendeten Begriffen wurde von den Begrifflichkeiten der FUNSOFT-Netz-Welt abstrahiert und der Gebrauch von Fachausdrücken vermieden. Durch die einfach gehaltene Dialogführung konnte der Schulungsaufwand für das Werkzeug gering gehalten und eine intuitive Bedienung gewährleistet werden. Dies dürfte bei den im Umgang mit Computern unerfahrenen Anwendungsbenutzern zur Akzeptanz der für sie erstellten LEU-Anwendungen beitragen und ihnen den Einstieg in die *EDV-unterstützte Durchführung von Geschäftsprozessen* erleichtern.

Kapitel 6

Testen der Werkzeuge

Dieses Kapitel beschreibt das Vorgehen beim Testen des Agenden-Controllers und des Agenda-Terminals. In diesem Zusammenhang wird nur auf die Testmethoden, nicht die einzelnen Testfälle, eingegangen, da dieses den Rahmen der Arbeit sprengen würde. Unter dem Begriff "Testen" wird hierbei das Sicherstellen der Funktionalität, also die Behebung der Fehler bei der Implementierung der Werkzeuge, verstanden. Das Kapitel unterteilt sich in drei Abschnitte. In Abschnitt 6.1 wird das Testen der einzelnen Module eines Werkzeugs, in Abschnitt 6.2 das Testen eines Werkzeugs und in Abschnitt 6.3 die Integrationstests der beiden Werkzeuge beschrieben. Diese Aufteilung entspricht der Vorgehensweise beim Testen der Werkzeuge. Zunächst wurde durch die Modultests die Korrektheit aller Datenmodule des Werkzeugs sichergestellt. Danach wurden die Werkzeuge einzelnd, daß heißt ohne die anderen Werkzeuge der LEU-Ausführungskomponente getestet, und schließlich in mehreren Schritten miteinander integriert.
Am Ende dieses Abschnitts wird das Testen der Oberfläche der Agenda-Terminals beschrieben. Die Oberflächentests wurden während der Komponententests durchgeführt.

6.1 Modultests

Mit Hilfe des Modultests wurde die Funktionalität der Datenmodule des Agenden-Controllers und des Agenda-Terminals sichergestellt. Zu den Datenmodulen des Agenden-Controllers gehören die Module *ACBact*, *ACBus* und *ACBut* (siehe Abb. 3.3 auf Seite 60). Die Datenmodule des Agenda-Terminals sind *TIbp*, *TIpro*, *TIprm*, *TIact*, *TIloc*, *TIcha*, *TIcd* und *TIobj* (siehe Abb. 4.2 auf Seite 76). Für die Modultests wurde für jedes Datenmodul ein Testrahmen geschrieben, durch die die einzelnen Funktionen der Module getestet und ihre Funktionalität sichergestellt werden konnte. Bei den Tests wurde nicht, wie oft in der Literatur [CMD91] empfohlen, ein Testrahmen für alle *"Benutzt-Beziehungen"*, das heißt für alle aus anderen Modulen benutzen Funktionen, geschrieben. Statt dessen wurde beim Testen *"Buttom-Up"* vorgegangen und somit zunächst die Module getestet, die keine Importschnittstelle derselben Komponente mehr besitzen. Beim Testen der Komponente *TI* (siehe Abb. 4.2 auf Seite 76) wurden also zunächst die Module *TIcd* und *TIobj* getestet. Für die Importschnittstelle der Datenmodule aus anderen Komponenten wurde ebenfalls kein Testrahmen erzeugt, da diese Komponenten schon über längere Zeit in den anderen LEU-Werkzeugen verwendet werden.

Da die Datenmodule von vornherein relativ fehlerlos (Fehler in allen Datenmodulen < 10 !!) waren, konnten bei den Modultests nur sehr wenige Fehler gefunden und behoben werden. Trotzdem halte ich den Aufwand, den die Modultests mit sich brachten, für gerechtfertigt, da hierdurch die Fehler, die in den späteren Tests entdeckt wurden, besser eingegrenzt werden konnten.

6.2 Komponententests

Innerhalb der Komponententests wurden die beiden Komponenten TI und ACB, also das Agenda-Terminal und der Agenden-Controller, auf funktionale Korrektheit getestet. Da beide Werkzeuge über eine Interprozeß-Kommunikation mit anderen Werkzeugen Nachrichten austauschen müssen, wurde ein *universeller IPC-Client* entwickelt, mit dem man beliebige Nachrichten an andere Clients versenden und von ihnen empfangen kann. Auf diese Weise war es möglich, den Datenaustausch über die Interprozeß-Kommunikation mit anderen Werkzeugen zu simulieren und jedes Werkzeug getrennt voneinander auszutesten. Beim Verschicken der Nachrichten, die sich an die für den Datenaustausch abgesprochenen Protokolle halten mußten, wurde darauf geachtet, die Daten der Nachrichten so aufzubauen, daß jede Zeile Programmcode der Nachrichtenverarbeitungsfunktionen mindestens einmal durchlaufen wurde. Auf diese Weise konnte die funktionale Korrektheit der einzelnen Verarbeitungsfunktionen sichergestellt werden.

Diese Art des Testens gestaltete sich als sehr mühselig, bildete aber die einzige Möglichkeit, Testdaten in die Datenhaltung der Werkzeuge einzufügen und im Agenda-Terminal an der Oberfläche anzuzeigen und zu verarbeiten. Auch diese Art des Testens halte ich im Nachhinein für notwendig und nützlich, da jeder so gefundene und beseitigte Fehler die nachfolgene Integration der einzelnen Werkzeuge der LEU-Ausführungskomponente wesentlich vereinfachte. Während der Komponententests konnte jedoch nur der korrekte Empfang und das korrekte Versenden einzelner Nachrichten sichergestellt werden. Im realen Betrieb der Werkzeuge müssen diese jedoch eine Vielzahl quasi gleichzeitig ankommender und zu verschickender Nachrichten verarbeiten können. Diese Funktionalität wurde während der anschließenden Integrationstests sichergestellt.

6.3 Integrationstests

Die im Anschluß an die Komponententests durchgeführten Integrationstests unterteilten sich in mehrere Abschnitte. Zunächst wurden die Ausführungswerkzeuge innerhalb der LEU-Controllerstruktur (siehe Abb. 2.6 auf Seite 25) miteinander integriert. Hierzu zählen der LEU-Controller, der Kommunikations-Monitor, die Agenden-Controller und die Agenda-Terminals. Die anderen Werkzeuge wurden über den für die Komponententests entwickelten *universellen IPC-Client* simuliert. Nach erfolgreichem Abschluß dieser ersten Integrationsstufe wurden dann die anderen Werkzeuge der Agenda-Controllerstrukturen (siehe Seite 25ff.) nach und nach integriert. Zu diesen Werkzeugen zählen der Agenden-Controller, der Geschäftsprozeß-Interpreter und -Monitor sowie die Dialog- und C-Interpreter. Am Ende dieser zweiten Integrationsphase konnte zum ersten Mal ein realer Geschäftsprozeß zur Ausführung gebracht werden. Beide

Phasen gestalteten sich, trotz der vorausgegangenen Komponententests, in denen die Funktionalität aller Werkzeuge im Stand-Alone-Betrieb sichergestellt worden war, sehr problembehaftet und zeitintensiv, da durch den parallelen Datenaustausch zwischen den Werkzeugen und das dadurch wachsende Datenaufkommen noch größere Probleme, besonders bei der Kommunikation und Nachrichtenverarbeitung, aufgedeckt wurden.

Nach Beendigung dieser zweiten Integrationsphase wurde, mit speziell für diesen Zeck ausgelegten Geschäftsprozeßmodellen, *Last-Tests* gefahren. Hierbei wurde von mehreren Personen bis zu zehn Geschäftsprozesse gleichzeitig ausgeführt und die LEU-Ausführungskomponente unter höherer Last im realitätsnahen Betrieb getestet.

6.4 Testen der Oberfläche

Die Oberflächentests sind Bestandteil der Komponententests und wurden während dieser Zeit durchgeführt. Dabei wurde für jedes Fenster, jeden Pushbutton und jeden Menüeintrag eine Testfallspezifikation aufgestellt, um so die Oberflächensteuerung anhand dieser Spezifikation überprüfen und ihre Funktionalität sicherstellen zu können. In den Testfallspezifikationen wurden alle durch eine Benutzeraktion auszulösenden Veränderungen der Oberfläche (Aktivieren bzw. Deaktivieren von Menüeinträgen und Pushbuttons, Öffnen bzw. Schließen von Dialogboxen, Füllen der Listboxen und Tablefields) festgehalten.
Da die Erstellung der Testfallspezifikationen nur einen geringen Arbeitsaufwand bedeutete und diese bei Änderungen und Erweiterungen der Oberflächenfunktionalität als Testmenge wiederverwendet werden konnten, halte ich die Erstellung von Testfallspezifikationen für eine geeignete Möglichkeit, Oberflächentests durchzuführen.
Für die Testfallspezifikation wurde hierbei das in Abbildung 6.1 zu sehende Formblatt verwendet. Das Beispiel beschreibt die Testfallspezifikation für das Starten eines neuen Geschäftsprozesses.

leu	Testfälle	Seite
Testobjekt-Nr.: 2		Datum : 06.07.94 Autor : RTh

LEU-Werkzeug:

(Werkzeugmanager, Onlinehilfe-System, Dialog-Ausführung, Administrations-Modellierung, Geschäftsprozeß-Modellierung, Dialog-Modellierung, Daten-Modellierung, Druck-System, Funktions-Verarbeitung, Report-Verarbeitung, Anfrage-Verarbeitung, Geschäftsprozeß-Ausführung, Agenda-Terminal, Funktions-Testumgebung, Dokumentations-System, Import-Export-System, Druck-Utility, Controller, Agenden-Controller, Kommunikationsmonitor, Ausführung)

LEU-Teilwerkzeug:

(Projektverwaltung, Rollen-Editor, Benutzerverwaltung, Aufbau-Editor, Datenmodell-Editor, Datenmodell-Analyse, Feld-Editor, Feldtyp-Editor, Funktions-Editor, Archivierungswerkzeug, Geschäftsprozeß-Editor, Geschäftsprozeß-Simulator, Dokumentations-Editor, Dokumentations-Browser, Hilfesystem-Editor, Hilfesystem-Browser, Dialog-Editor, Dialog-Simulation, Anbindungs-Editor, Anfrage-Editor, Anfrage-Interpreter, C-Test-Werkzeug, Hilfetextsystem)

Beschreibung des zu testenden Objektes:

Dialogmaske : Geschäftsprozesse starten

Beschreibung der Testfälle:
(Nummerierung: <Testobjekt-Nr.>.<Testfall-Nr.>)

2.1 Versuch aus einem leeren Tabellenfeld etwas zu selektieren
→ OK-Button muß insensitiv bleiben

2.2 Versuch mehrere GP's zu selektieren ↯

2.3 Zu einem nicht mehrbenutzerfähigen GPM einen mehrbenutzerfähigen GP starten ↯

2.4 Eingabe eines 'maximalen' Strings im Edittext

2.5 GPM, das nicht instanziierbar ist mehrfach starten ↯

2.6 OK-Button immer insensitiv, wenn kein GPM selektiert ?

2.7 Selektion eines GPM's => OK-Button sensitiv ?

2.8 Funktionalität über Poptext und Menüitems anwählbar

2.9 Beim Starten eines GP's keine 'Kurzinfo' im Edittext ausgeben

LEU-FB-002

Abbildung 6.1: Formblatt für die Testfallspezifikationen der Oberflächentests

Kapitel 7

Zusammenfassung und Ausblick

Im Implementierungsteil dieser Arbeit ist die Benutzeroberfläche für den Ausführungsteil des Geschäftsprozeß-Management-Systems LEU und der Agenden-Controller, der die innerhalb der Geschäftsprozesse zur Bearbeitung anstehenden Aktivitäten unter Berücksichtigung von Zuständigkeiten in die Agenden der Sachbearbeiter schreibt, entwickelt worden.

Der schriftliche Teil dieser Arbeit beschreibt die Entwicklung dieser beiden Werkzeuge. Hierzu wurde im Kapitel 1 zunächst die Motivation für die Arbeit gegeben und der derzeitige Stand des Geschäftsprozeß-Management und des dem Geschäftsprozeß-Management nah verwandten Software-Prozeß-Management aufgezeigt (Abschnitt 1.4). In Kapitel 2 wurde das Geschäftsprozeß-Management-System LEU beschrieben, zu dessem Ausführungsteil die innerhalb dieser Arbeit implementierten Werkzeuge gehören. In diesem Kapitel wurde auch der Aufbau und die anderen Werkzeuge der LEU-Ausführungskomponente beschrieben. Durch die Realisierung der LEU-Ausführungskomponente als verteiltes System mußte der Datenaustausch zwischen den einzelnen Werkzeugen über eine Interprozeß-Kommunikation geleistet werden. Es mußten die Kommunikationsbedürfnisse der einzelnen Werkzeuge untereinander geklärt und eine geeignete Kommunikationsstruktur gefunden werden. Der Auf- und Abbau der verschiedenen Kommunikationspfade und das über die IPC-Service zu übermittelnde hohe Datenaufkommen stellten auch bei der Implementierung hohe Anforderungen an die einzelnen Entwickler und gestalteten die Integration der Werkzeuge als schwierig und zeitraubend. Kapitel 3 und 4 beschreiben die Entwicklung des Agenden-Controllers und des Agenda-Terminals. Hier werden die Anforderungen an die beiden Werkzeuge und ihre Realisierung, deren Beschreibung auch die Modulhierarchie und den Grobentwurf der beiden Werkzeuge enthält, näher erläutert. Kapitel 5 beschreibt den Entwicklungsprozeß der Benutzeroberfläche des Agenda-Terminals. In die Entwicklung der Oberfläche konnten die späteren Benutzer des Agenda-Terminals einbezogen werden. So konnte die Oberfläche nach einem Prototyping-Verfahren in enger Zusammenarbeit mit den späteren Anwendern erstellt werden. Aus diesem Grund wurde die Entstehung der Benutzeroberfläche aus Kapitel 4 ausgelagert und in einem gesonderten Kapitel beschrieben. Auf das Testen der einzelnen Werkzeuge und ihre Integration wird in Kapitel 6 näher eingegangen.

Abschließend sollen einige Erweiterungsmöglichkeiten des Agenda-Terminals und des Agenden-Controllers aufgezeigt werden. Beispielsweise erscheint es wünschenswert, den Geschäftsprozeß-Monitor (siehe Abschnitt 2.5.8), der mit seiner Oberfläche den derzeitigen Bearbeitungsstand aller ausgeführten Geschäftsprozesse visualisiert, direkt aus dem Agenda-Terminal starten zu können. Des weiteren müßte auf die, im Geschäftsprozeß-Monitor erkannten Engpässe (Datenstau, Termindruck, usw.) bei der Ausführung der Geschäftsprozesse aktiv reagiert werden können. So müßten zum Beispiel die Aktivitäten, in deren Vorbereich es zu einem Datenstau gekommen ist oder deren Bearbeitung aufgrund festgesetzter Termine Vorrang hat, die Bearbeitung aller anderen Aktivitäten des jeweiligen Geschäftsprozesses sperren können. Ebenso sollte in den Dialogboxen, die der Objektauswahl dienen (siehe Abb. 5.9 auf Seite 101), ein Suchstring eingegeben und somit eine Vorauswahl bei den Objekten getroffen werden können, was besonders die Objektauswahl bei großen Objektmengen erheblich erleichtern würde.

Im Agenden-Controller könnten die Berechtigungsüberprüfungen auf die objektwertabhängigen Berechtigungen ausgeweitet werden. Mit den objektwertabhängigen Berechtigungen kann beispielsweise festgelegt werden, daß die Anwendungsbenutzer mit der Rolle *Sachbearbeiter* nur die erfaßten Personen, deren Anfangsbuchstabe des Nachnamens zwischen "A" und "E" liegt, bearbeiten dürfen. Bisher wird diese Art der Berechtigungsüberprüfungen im Dialog-Interpreter (siehe Abschnitt 2.5.5) vor dem Füllen der Dialoge geleistet. Auf diese Weise kann es derzeit noch vorkommen, daß die Anwendungsbenutzer Aktivitäten in ihren Agenden stehen haben, zu deren Ausführung sie nicht berechtigt sind und die Bearbeitung dieser Instanzen vom Dialog-Interpreter abgelehnt wird, nachdem diese sie zur Bearbeitung ausgewählt haben.

Kapitel 8

Anhang A

8.1 Ein beispielhaftes Ausführungsszenario

Im folgenden wird in einem beispielhaften Ausführungsszenario das Agenda-Terminal und der Agenden-Controller im praktischen Einsatz gezeigt. Es werden die Geschäftsprozesse *"Auftragserteilung"* und *"Rechnungsbearbeitung"*, deren Ablaufmodelle die Abbildungen 8.1 und 8.4 zeigen, zur Ausführung gebracht. Diese beiden Geschäftsprozesse sind über $\xrightarrow{2.2}$Systemkanäle miteinander verbunden.

Im Geschäftsprozeß *"Auftragserteilung"* wird ein zu erteilender Auftrag zunächst mit seinen Daten erfaßt (Instanz *Auftragserteilung*) und als Bestellung oder Beschaffung klassifiziert (mit einer Bestellung werden unter anderem kurzfristig benötigte Verbrauchsmaterialien geordert, während mit einer Beschaffung langfristige Abschreibegüter geordert werden). Danach durchläuft der Auftrag (abhängig von der Klassifizierung) die entsprechenden Prüfungs- und Genehmigungsinstanzen (*Auftragsprüfung*, *Auftragssonderprüfung*, *Beschaffung genehmigen*, *Bestellung genehmigen*). Zu jeder Prüfung bzw. Genehmigung wird ein Prüfungsprotokoll (Instanz *Protokollarchivierung*) erzeugt, das anschließend auch archiviert wird. Wurde ein Auftrag geprüft und genehmigt, so muß der entsprechende Lieferant, der für die Auftragsabwicklung zuständig sein soll, ausgewählt werden.

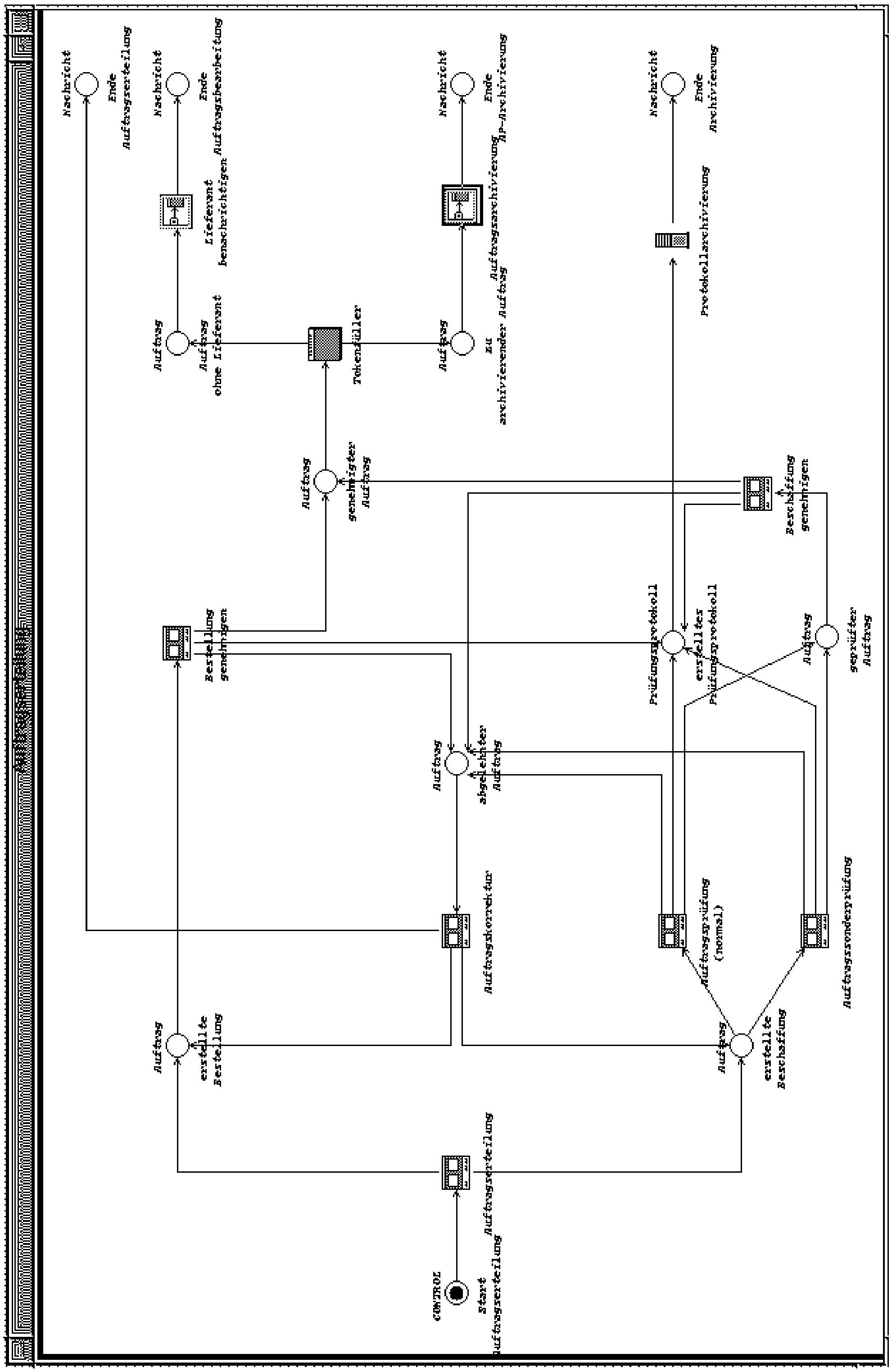


Abbildung 8.1: Ablaufmodell des Geschäftsprozesses *Auftragserteilung*

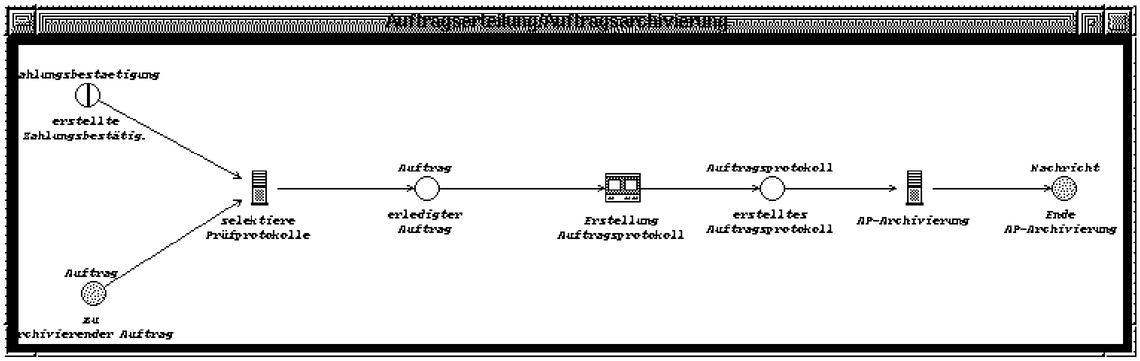


Abbildung 8.2: Subnetz des Ablaufmodells vom Geschäftsprozeß *Auftragserteilung* (verfeinerte Instanz *Auftragsarchivierung*)

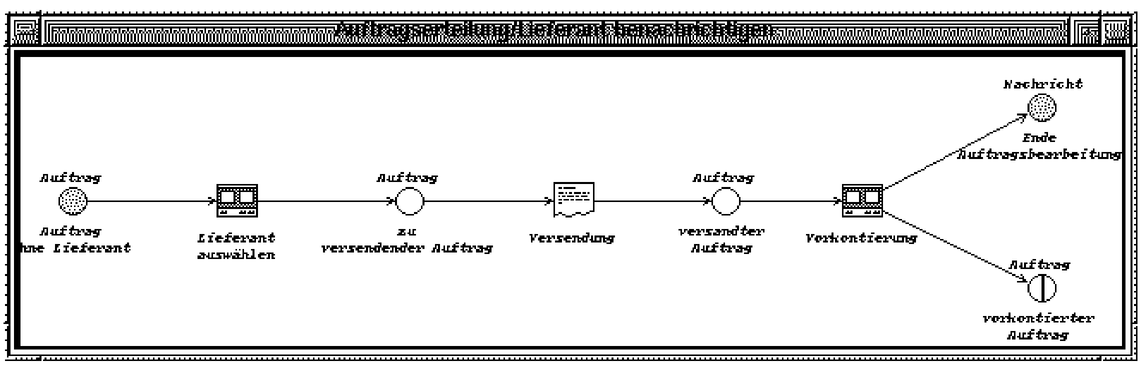


Abbildung 8.3: Subnetz des Ablaufmodells vom Geschäftsprozeß *Auftragserteilung* (verfeinerte Instanz *Lieferant benachrichtigen*)

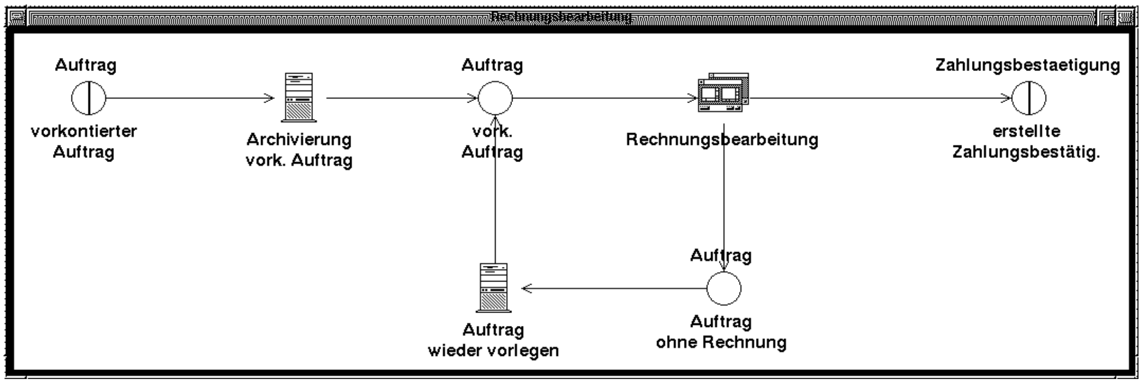


Abbildung 8.4: Ablaufmodell des Geschäftsprozesses *Rechnungsbearbeitung*

Gleichzeitig muß der Geschäftsprozeß *"Rechnungsbearbeitung"* gestartet werden. In diesem Geschäftsprozeß wird die Zahlungsbestätigung für den Auftrag erstellt. Konnte sie erstellt werden, so wird der Auftrag zusammen mit einem Auftragsprotokoll endgültig archiviert (*Instanz AP-Archivierung* in Abb. 8.2).
Die Abbildungen 8.2 und 8.3 sind Subnetze, die zum Ablaufmodell des Geschäftsprozesses *Auftragserteilung* gehören. Das in Abbildung 8.2 dargestellte Netz gehört zur verfeinerten Instanz *Auftragsarchivierung*. Das in Abbildung 8.3 dargestellte Netz gehört zur verfeinerten Instanz *Lieferant benachrichtigen*.

Während der Berechtigungsmodellierung wurden für die Ausführung dieser beiden Geschäftsprozesse folgende $\xrightarrow{1.2}$Rollen modelliert:

- **Buchhalter**
  Diese Rolle definiert, daß in dem Geschäftsprozeß *"Auftragserteilung"* nur die Aktivität *"Vorkontierung"* und der Geschäftsprozeß *"Rechnungsbearbeitung"* komplett ausgeführt werden darf. Mit dieser Rolle dürfen Geschäftsprozesse vom Typ *"Rechnungsbearbeitung"* gestartet werden.

- **Einkäufer**
  Diese Rolle definiert, daß in dem Geschäftsprozeß *"Auftragserteilung"* die Aktivitäten *"Auftragsprüfung (normal)"*, *"Auftragssonderprüfung"*, *"Lieferant auswählen"* und *"Erstellung Auftragsprotokoll"* durchgeführt werden dürfen. Zur Mitarbeit im Geschäftsprozeß *"Rechnungsbearbeitung"* besteht keine Berechtigung. Mit dieser Rolle dürfen keine Geschäftsprozesse gestartet werden.

- **Leiter**
  Diese Rolle definiert, daß in dem Geschäftsprozeß *"Auftragserteilung"* die Aktivitäten *"Auftragsprüfung (normal)"*, *"Auftragssonderprüfung"*, *"Auftragserteilung"*, *"Auftragskorrektur"*, *"Bestellung genehmigen"* sowie *"Beschaffung genehmigen"* durchgeführt werden dürfen. Zur Mitarbeit im Geschäftsprozeß *"Rechnungsbearbeitung"* besteht keine Berechtigung. Mit dieser Rolle dürfen Geschäftsprozesse vom Typ *"Rechnungsbearbeitung"* und *"Auftragserteilung"* gestartet werden.

- **Mitarbeiter**
  Diese Rolle definiert, daß in dem Geschäftsprozeß *"Auftragserteilung"* die Aktivität *"Auftragserteilung"*durchgeführt werden darf. Zur Mitarbeit im Geschäftsprozeß *"Rechnungsbearbeitung"* besteht keine Berechtigung. Mit dieser Rolle dürfen Geschäftsprozesse vom Typ *"Auftragserteilung"* gestartet werden.

Im folgenden Szenario werden diese Rollen von folgenden Personen gespielt.

Name	**Rollen**
Hr. Riehl	Buchhalter
Hr. Nowak	Einkäufer
Fr. Neumann	Leiter
Fr. Hofmann	Mitarbeiter

Tabelle 8.1: Die Personen des Szenarios und ihre Rollen

Zu Beginn unseres Szenarios ist kein Mitarbeiter in die LEU-Ausführung eingeloggt, und es wird kein Geschäftsprozeß ausgeführt. Frau Neumann startet ihr Agenda-Terminal und loggt sich mit ihrer *Leiter*-Roller in die LEU-Ausführung ein. Sie startet zum Geschäftsprozeßmodell *"Auftragserteilung"* einen mehrbenutzerfähigen Geschäftsprozeß (siehe Abb. 8.5).

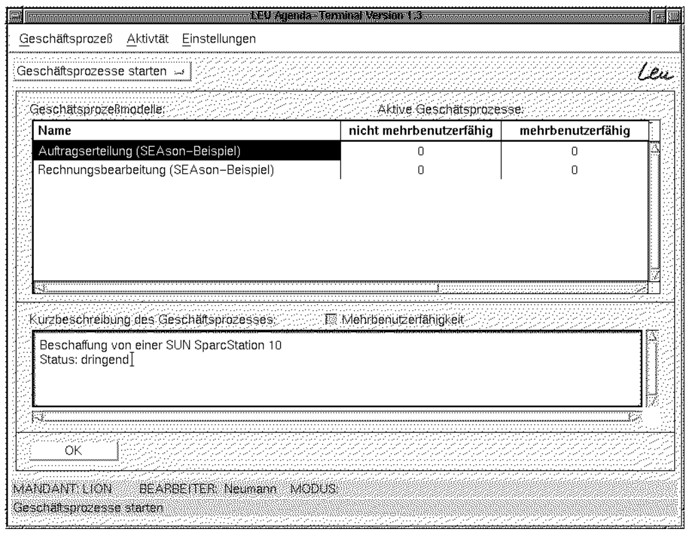


Abbildung 8.5: Frau Neumann startet zum Geschäftsprozeßmodell *"Auftragserteilung"* einen neuen Geschäftsprozeß

Hierbei gibt sie als Kurzinformation an, daß eine *SUN SparcStation 10* beschafft werden muß und kennzeichnet den Auftrag als dringend.
Der LEU-Controller wird von Frau Neumanns Agenda-Terminal mit der Nachricht *KSB_TI_LC_START_INC* über den neu zu startenden Geschäftsprozeß informiert. Nach dem Erhalt dieser Nachricht startet der LEU-Controller einen Agenden-Controller und teilt diesem über die IPC-Nachricht *KSB_LC_ACB_START_INC* alle benötigten Informationen mit. Der Agenden-Controller startet daraufhin den Geschäftsprozeß-Interpreter und übermittelt ihm mit der IPC-Nachricht *KSB_ACB_VI_START_NEW_-PROCESS* die Informationen, die er benötigt um den Geschäftsprozeß zu starten. Nachdem der Geschäftsprozeß-Interpreter den neuen Geschäftsprozeß erfolgreich gestartet hat, sendet er die Nachricht *KSB_VI_ACB_-PROCESS_STARTED* an den Agenden-Controller und beginnt danach mit der Interpretation des Ablaufmodells. Er stellt fest, daß die manuelle Akti-

vität *"Auftragserteilung"* aktivierbar ist und übermittelt sie dem Agenden-Controller. Nachdem der Agenden-Controller vom Geschäftsprozeß-Interpreter die Nachricht *KSB_VI_ACB_PROCESS_STARTED* empfangen hat, unterrichtet er mit der Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die Existenz des neuen Geschäftsprozesses. Da Frau Neumann derzeit die einzige ist, die in die LEU-Ausführung eingeloggt ist, wurde nur ihr Agenda-Terminal über den neuen Geschäftsprozeß informiert. Ihr Agenda-Terminal meldet sich daraufhin unter Angabe aller von Frau Neumann gespielten Rollen (derzeit nur die Rolle *Leiter*) beim Agenden-Controller des neuen Geschäftsprozesses an. Dieser überprüft, ob Frau Neumann zur Mitarbeit am Geschäftsprozeß *"Auftragserteilung"* berechtigt ist, und stellt fest, daß sie mit ihrer Rolle einige Aktivitäten des Geschäftsprozesses ausführen darf. Dieses Ergebnis der Berechtigungüberprüfung meldet der Agenden-Controller an das Agenda-Terminal von Frau Neumann zurück und trägt sie in seine Datenhaltung ein.

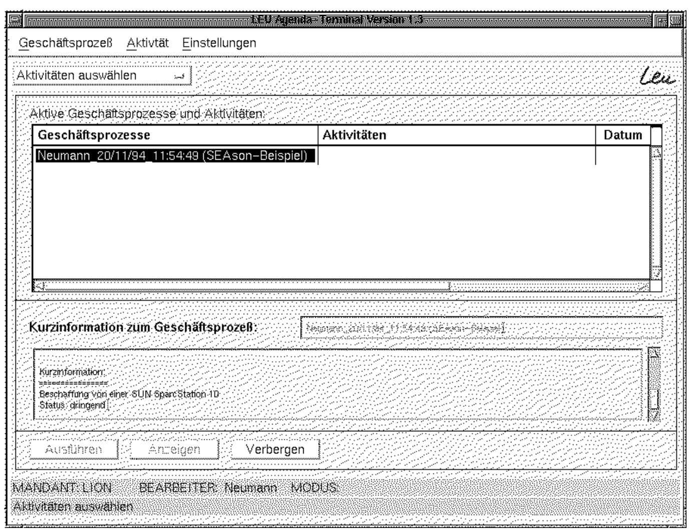
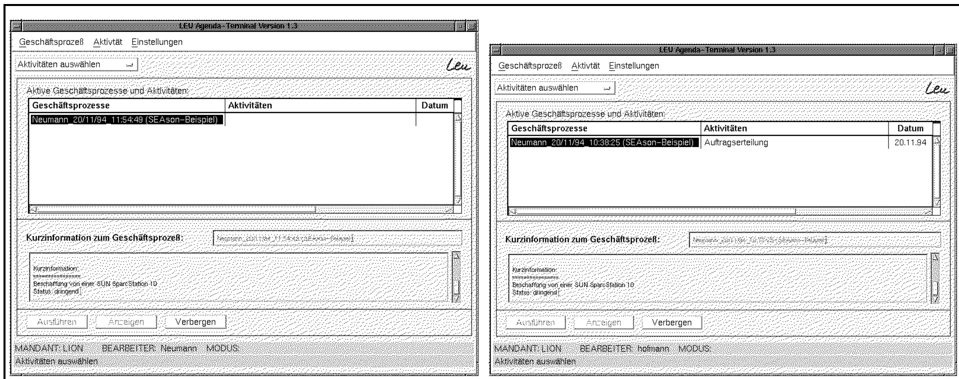


Abbildung 8.6: Das Agenda-Terminal von Frau Neumann nach dem Start des Geschäftsprozesses *"Auftragserteilung"*

Da Frau Neumann jedoch keine Berechtigung auf die einzige derzeit aktivierbare manuelle Aktivität (*"Auftragserteilung"*) hat, werden vom Agenden-Controller keine Aktivitäten an das Agenda-Terminal von Frau Neumann übermittelt. Nachdem das Agenda-Terminal vom Agenden-Controller darüber informiert worden ist, daß Frau Neumann zur Mitarbeit am von ihr gestarteten Geschäftsprozeß berechtigt ist, wurde der Geschäftsprozeß (der Name eines Geschäftsprozesses setzt sich aus dem Benutzer, der den Geschäftsprozeß gestartet hat, dem Datum und der Uhrzeit, wann der Geschäftsprozeß gestartet worden ist, und dem Namen des Projekts, dem der Geschäftsprozeß zugehört, zusammen) in die Datenhaltung des Agenda-Terminals eingetragen und an der Oberfläche angezeigt (siehe Abb. 8.6).

Nun startet Frau Hofmann ihr Agenda-Terminal und loggt sich mit ihrer Rolle *Mitarbeiter* in die LEU-Ausführung ein. Sie wird mit dem Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* (In den Agenda-Terminals als *"Neumann_20/11/94_10:38:25(SEAson-Beispiel)"* angezeigt) verbunden, und ihr Agenda-Terminal übermittelt dem Agenden-Controller mit der Nachricht *KSB_TI_ACB_NEW_USER_LOGIN* die Rollen (Rolle *Mitarbeiter*), mit denen Frau Hofmann in der LEU-Ausführung eingeloggt ist. Der Agenden-Controller überprüft, ob Frau Hofmann zur Mitarbeit am Geschäftsprozeß berechtigt ist und stellt fest, daß sie die Aktivität *"Auftragserteilung"* ausführen darf. Er trägt Frau Hofmann in seine Datenhaltung ein und übermittelt das Ergebnis der Berechtigungsüberprüfung mit der Nachricht *KSB_ACB_TI_LOGIN_ANSWER* an das Agenda-Terminal von Frau Hofmann. Hierauf überprüft der Agenden-Controller alle derzeit aktivierbaren manuellen Aktivitäten daraufhin, ob sie von Frau Hofmann ausgeführt werden dürfen. Er stellt fest, daß die einzige derzeit aktivierbare Aktivität *"Auftragserteilung"* von ihr ausgeführt werden darf und sendet sie mittels der Nachricht *KSB_ACB_TI_AGENCY* an ihr Agenda-Terminal. Das Agenda-Terminal von Frau Hofmann empfängt diese Nachrichten, trägt den Geschäftsprozeß *"Neumann_20/11/94_10:38:25(SEAson-Beispiel)"* und die Aktivität *"Auftragserteilung"* in seine Datenhaltung ein und aktualisiert die Oberfläche des Agenda-Terminals.
Abbildung 8.7 gibt den Stand der Agenden von allen derzeit an der Ausführung beteiligten Personen wider. Frau Hofmann wählt nun in ihrer Agenda die Aktivität *"Auftragserteilung"* zur Bearbeitung aus. Die Aktivität wird über die Nachricht *KSB_TI_ACB_START_AGENCY* an den



Agenda Fr. Neumann Agenda Fr. Hofmann

Abbildung 8.7: Stand der Agenden (1)

Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* übermittelt. Er überprüft, ob diese Aktivität an mehrere Benutzer zur Auswahl übermittelt worden ist und somit nun aus den anderen Agenden gelöscht werden muß. Da Frau Hofmann die Aktivität *"Auftragserteilung"* jedoch als einzige in ihrer Agenda stehen hatte, kann er die Aktivität direkt an den Geschäftsprozeß-Interpreter über die Nachricht *KSB_ACB_VI_START_-AGENCY* weiterreichen. Der Geschäftsprozeß-Interpreter startet für die Ausführung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm die Daten der von Frau Hofmann ausgewählten Aktivität. Der Dialog-Interpreter lädt den zu dieser Aktivität gehörenden Dialog und stellt ihn auf dem Rechner von Frau Hofmann dar (siehe Abb. 8.8).
In dem an die Aktivität *"Auftragserteilung"* angebundenen Dialogfenster (siehe Abb. 8.8) gibt Frau Hofmann alle für den Auftrag benötigten Daten, wie die *Auftragsnummer*, das *Auftragsdatum*, die *Auftragsposition*, die Frau Hofmann der beim Starten des Geschäftsprozesses von Frau Neumann eingegebenen Kurzinformation entnehmen konnte, die *Lieferungsart*, den *Auftraggeber/Bereich*, die *Auftragsart* und den *Auftragswert* im Feld *"Betrag"* ein. Im Feld *"Erläuterung"* kann sie zusätzlich noch eine textuelle Erläuterung zum erfaßten Auftrag angeben. Nachdem Frau Hofmann den Auftrag komplett erfaßt hat, drückt sie den OK-Button der Dialogbox. Damit ist die Aktivität *"Auftragserteilung"* komplett abgearbeitet. Als Auftragsart wurde *"Beschaffung"* ausgewählt und daher wird der Auftrag in den Nach-

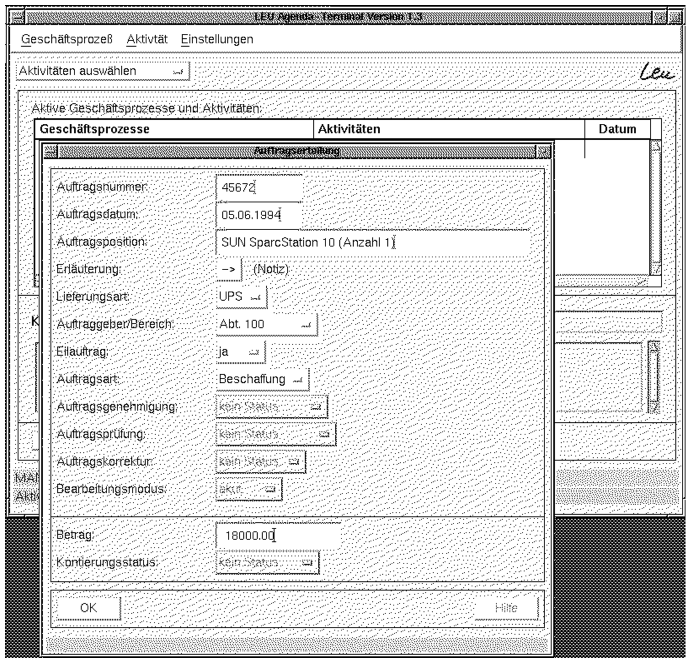


Abbildung 8.8: Bearbeitung des Dialogs *"Auftragserteilung"* durch Frau Hofmann

bereichskanal *"erstellte Beschaffung"* geschrieben (siehe Abb. 8.1 auf Seite 112). Der Agenden-Controller wird vom Geschäftsprozeß-Interpreter über die Beendigung der Aktivität mit der Nachricht *KSB_VI_ACB_AGENCY_-DONE* benachrichtigt. Nach Empfang dieser Nachricht löscht der Agenden-Controller die Aktivität *"Auftragserteilung"* über die Nachricht *KSB_ACB_-TI_DELETE_AGENCY* aus der Datenhaltung des Agenda-Terminals von Frau Hofmann.

Nun startet Herr Nowak sein Agenda-Terminal und loggt sich mit seiner Rolle *Einkäufer* in die LEU-Ausführung ein. Er wird mit dem Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* verbunden (siehe Anmeldung Frau Hofmann). Der Agenden-Controller überprüft, ob Herr Nowak zur Mitarbeit im Geschäftsprozeß berechtigt ist, und stellt fest, daß er einige Aktivitäten ausführen darf. Das Ergebnis dieser Berechtigungsüberprüfung meldet er an das Agenda-Terminal von Herrn Nowak zurück (siehe Anmeldung Frau Hofmann). Daraufhin wird der Geschäftsprozeß in die Datenhaltung des Agenda-Terminals von Herrn Nowak eingetragen und an der Oberfläche angezeigt. Da derzeit keine manuellen Aktivitäten im Geschäftsprozeß zur Bearbeitung anstehen, wird dem Agenda-Terminal von Herrn Nowak keine Aktivität übermittelt.

Während dessen hat der Geschäftsprozeß-Interpreter einen neuen Aktivierbarkeittest durchgeführt und festgestellt, daß die manuellen Aktivitäten *"Auftragsprüfung (normal)"* und *"Auftragssonderprüfung"* aktivierbar sind. Er übermittelt diese beiden Aktivitäten über Nachrichten vom Typ *KSB_VI_ACB_AGENCY* an den Agenden-Controller. Der Agenden-Controller überprüft, ob einer der derzeit an der Ausführung beteiligten Benutzer zur Bearbeitung dieser Aktivitäten berechtigt ist. Die Überprüfung ergibt, daß sowohl Frau Neumann als auch Herr Nowak die Ausführung dieser Aktivitäten übernehmen können. Der Agenden-Controller übermittelt die Aktivitäten mit Nachrichten vom Typ *KSB_ACB_TI_AGENCY* an die Agenda-Terminals dieser beiden Personen. Die Agenda-Terminals empfangen diese Meldung, tragen die Aktivitäten in die interne Datenhaltung ein und aktualisieren ihre Oberfläche. Abbildung 8.9 gibt den derzeitigen Stand der Agenden von allen an der Ausführung beteiligten Personen wider.

Herr Nowak wählt nun die Aktivität *"Auftragsprüfung (normal)"* in seinem Agenden-Terminal zur Bearbeitung aus. Hierbei werden ihm die von Frau Hofmann erfaßten Auftragsdaten angezeigt (siehe Abb. 8.10).
Die Aktivität wird über die Nachricht *KSB_TI_ACB_START_AGENCY* an den Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* übermittelt. Der Agenden-Controller stellt fest, daß die Aktivität auch an das Agenda-Terminal von Frau Neumann geschickt worden ist und nun aus der Agenda von Frau Neumann gelöscht werden muß. Daher sendet er die zu löschende Aktivität mit der Nachricht *KSB_ACB_TI_DELETE_AGENCY* an das Agenda-Terminal von Frau Neumann. Nach dem Empfang dieser Nachricht löscht ihr Agenda-Terminal die Aktivität aus seiner Datenhaltung und aktualisiert die Oberfläche.
Hierauf leitet der Agenden-Controller die Aktivität an den Geschäftsprozeß-

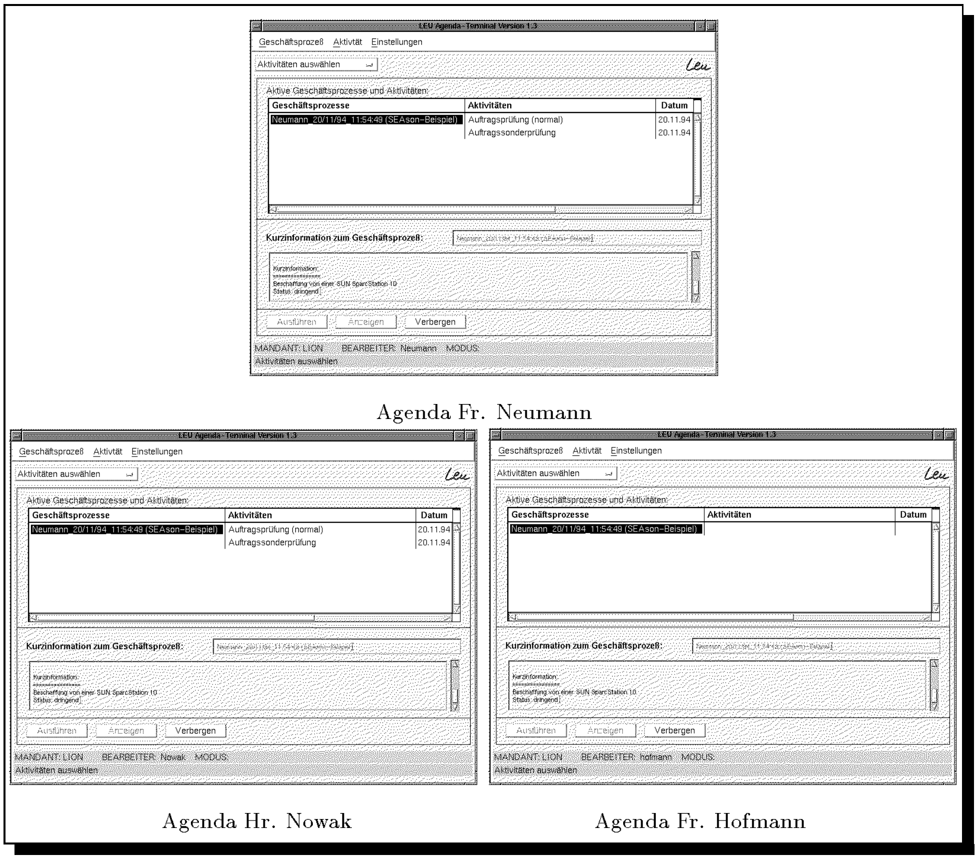


Abbildung 8.9: Stand der Agenden (2)

Interpreter weiter. Dieser startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Desweiteren führt der Geschäftsprozeß-Interpreter erneut eine Aktivierbarkeitsprüfung durch und stellt fest, daß die Aktivität *"Auftragssonderprüfung"* durch die Konfliktbereinigung (siehe $\xrightarrow{2.2}$Konflikte in FUNSOFT-Netzen) nicht mehr aktivierbar ist. Somit übermittelt er diese Aktivität mit der Nachricht *KSB_VI_ACB_DELETE_-AGENCY* an den Agenden-Controller. Dieser leitet diese Information über die Nachricht *KSB_ACB_TI_DELETE_AGENCY* an die Agenda-Terminals von Herr Nowak und Frau Neumann weiter und löscht die Aktivität aus seiner Datenhaltung. Die Agenda-Terminals von Herr Nowak und Frau Neumann löschen nach dem Empfang dieser Nachricht die Aktivität ebenfalls aus ihrer Datenhaltung und aktualisierem ihre Oberfläche.

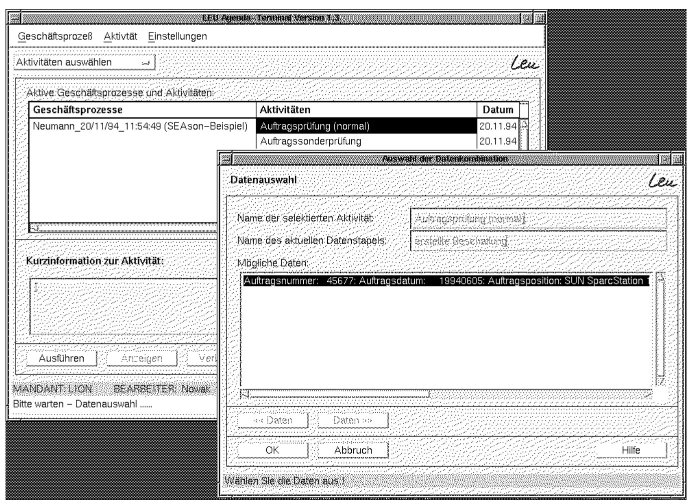


Abbildung 8.10: Auswahl der Aktivität *Auftragsprüfung (normal)* durch Herrn Nowak

Der Dialog-Interpreter hat während dieser Zeit den an die Aktivität *"Auftragsprüfung (normal)"* angebundenen Dialog gestartet und ihn auf dem Bildschirm von Herrn Nowak zur Anzeige gebracht (Abb. 8.11).

Herr Nowak bekommt den von Frau Hofmann erfaßten Auftrag auf sei-

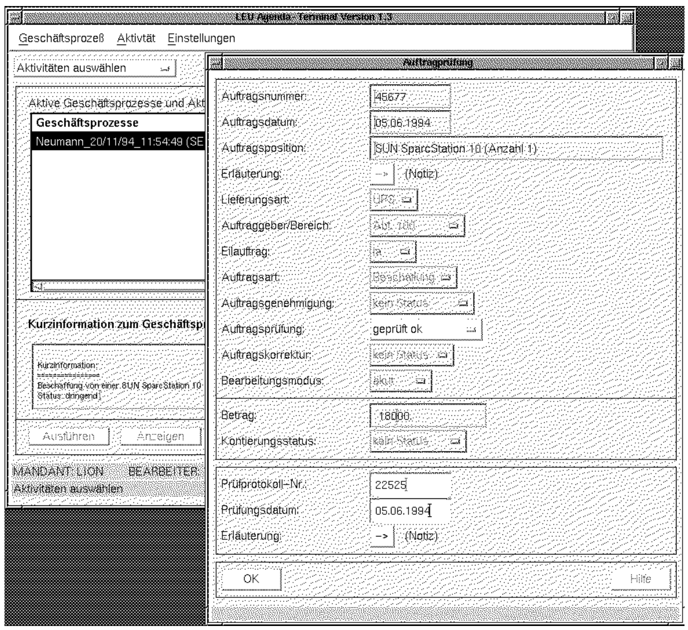


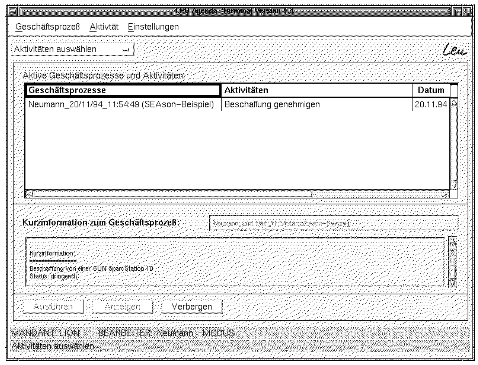
Abbildung 8.11: Bearbeitung des Dialogs *"Auftragsprüfung (normal)"* durch Herrn Nowak

nem Bildschirm angezeigt. Für diesen Auftrag hat er nun einen Prüfungsstatus im *Feld Auftragsprüfung* zu vergeben. Hierbei besteht für ihn die Möglichkeit zwischen *"geprüft ok"* und *"geprüft abgelehnt"* zu wählen. Herr Nowak gibt dem Auftrag den Status *"geprüft ok"*. Des weiteren hat er die Prüfung zu protokollieren. Hierzu gibt Herr Nowak eine *Prüfungsprotokoll-Nummer* und das *Prüfungsdatum* ein und hinterlegt im Feld *Erläuterung* eine *"Prüfungsnotiz"*.

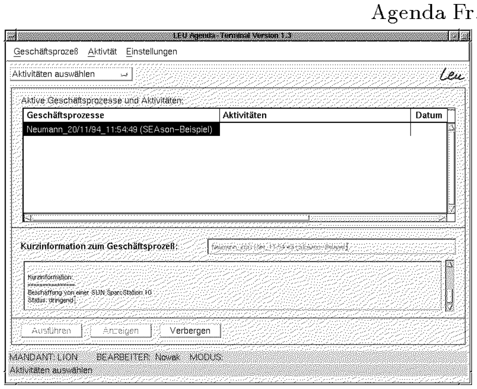
Nach Betätigung des OK-Buttons im Dialog *Auftragsprüfung* (Abb. 8.11) ist die Bearbeitung der Aktivität beendet und der Nachbereich der Aktivität wird gefüllt. Der Auftrag wird in den Nachbereichskanal *"geprüfter Auftrag"* und das Prüfungsprotokoll in den Nachbereichskanal *"erstelltes Prüfungsprotokoll"* geschrieben (siehe Abb. 8.1 auf Seite 112). Der Agenden-Controller wird vom Geschäftsprozeß-Interpreter mit der Nachricht *KSB_VI_ACB_AGENCY_DONE* über die Beendigung der Aktivität *"Auftragsprüfung (normal)"* informiert und leitet diese Information mit der Nachricht *KSB_ACB_TI_DELETE_AGENCY* an das Agenda-Terminal von Herrn Nowak weiter. Herr Nowaks Agenda-Terminal löscht nach dem Empfang dieser Nachricht die Aktivität *"Auftragsprüfung (normal)"* aus seiner Datenhaltung und aktualisiert die Oberfläche, in der die in Bearbeitung befindlichen Aktivitäten angezeigt werden. Während dessen hat der Geschäftsprozeß-Interpreter eine neue Aktivierbarkeitsprüfung durchgeführt und festgestellt, daß die automatische Instanz *"Protokollarchivierung"* und die manuelle Instanz *"Beschaffung genehmigen"* aktiviert werden können. Für die Abarbeitung der automatischen Instanz startet er einen C-Interpreter, der das erstellte Prüfungsprotokoll archiviert. Nach Beendigung dieser Aktivität wird ein Token vom Typ *"Nachricht"* in den Nachbereichskanal *"Ende Archivierung"* geschrieben.

Die aktivierbare manuelle Aktivität *"Beschaffung genehmigen"* wird vom Geschäftsprozeß-Interpreter an den Agenden-Controller übermittelt (Vorgehensweise siehe oben). Der Agenden-Controller überprüft, ob eine derzeit an der Ausführung des Geschäftsprozesses beteiligte Person zur Bearbeitung der Aktivität *"Beschaffung genehmigen"* berechtigt ist und stellt fest, daß Frau Neumann diese Aktivität ausführen darf. Somit übermittelt der Agenden-Controller die Aktivität an das Agenda-Terminal von Frau Neumann.
Abbildung 8.12 gibt den derzeitigen Stand der Agenden von allen an der Ausführung beteiligten Personen wider.

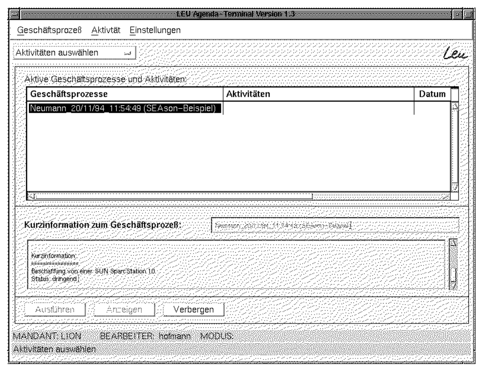
Frau Neumann startet die Aktivität *"Beschaffung genehmigen"* in ihrem Agenda-Terminal. Beim Starten bekommt sie die zum Auftrag erfaßten Daten angezeigt (Abb. 8.13).
Hierauf wird die Aktivität mit der Nachricht *KSB_TI_ACB_START_-*



Agenda Fr. Neumann



Agenda Hr. Nowak



Agenda Fr. Hofmann

Abbildung 8.12: Stand der Agenden (3)

AGENCY an den Agenden-Controller zum Starten übermittelt. Der Agenden-Controller überprüft, ob diese Aktivität an mehrere Benutzer zur Auswahl übermittelt worden ist und somit nun aus den anderen Agenden gelöscht werden muß. Da Frau Neumann die Aktivität *"Beschaffung genehmigen"* jedoch als einzige in ihrer Agenda stehen hatte, kann der Agenden-Controller die Aktivität direkt an den Geschäftsprozeß-Interpreter über die Nachricht *KSB_ACB_VI_START_AGENCY* weiterreichen. Dieser startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Der Dialog-Interpreter lädt den zur Aktivität *"Beschaffung genehmigen"* gehörenden Dialog und stellt ihn auf dem Rechner von Frau Neumann dar (siehe Abb. 8.14). Frau Neumann bekommt den von Frau Hofmann erfaßten und von Herrn Nowak geprüften Auftrag auf ihrem Bildschirm angezeigt.

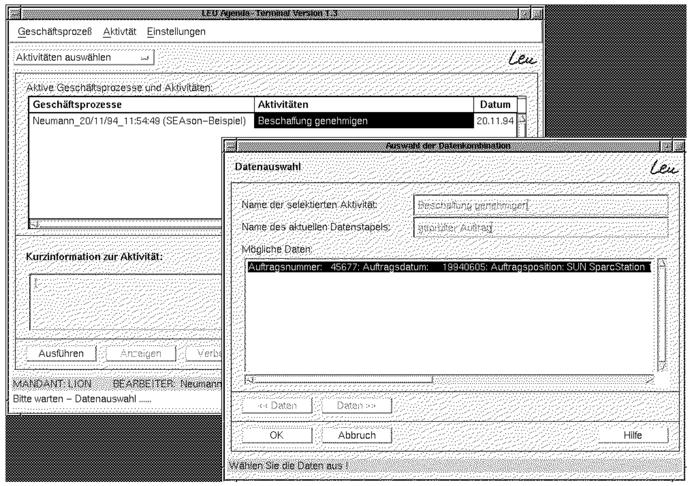


Abbildung 8.13: Auswahl der Aktivität *"Beschaffung genehmigen"* durch Frau Neumann

Für diesen Auftrag hat sie nun einen "Genehmigungsstatus" im *Feld Auftragsgenehmigung* zu vergeben. Hierbei kann Frau Neumann den Status *"genehmigt* und *"nicht genehmigt* vergeben. Sie genehmigt die Beschaffung und erstellt ein Prüfungsprotokoll, wobei sie eine *Prüfungsprotokoll-Nummer*, das *Prüfungsdatum* und im *Feld Erläuterung* eine "Prüfungsnotiz" hinterlegt. Nach Betätigung des OK-Buttons im Dialog *"Beschaffung genehmigen"* (Abb. 8.14) ist die Bearbeitung der Aktivität beendet und der Nachbereich der Aktivität wird gefüllt. Der genehmigte Auftrag wird in den Nachbereichskanal *"genehmigter Auftrag"* und das Genehmigungsprotokoll in den Nachbereichskanal *"erstelltes Prüfungsprotokoll"* geschrieben (siehe Abb. 8.1 auf Seite 112). Das Agenda-Terminal von Frau Neumann wird über die Beendigung der Aktivität *"Beschaffung genehmigen"* informiert (Vorgehensweise siehe oben).

Der Geschäftsprozeß-Interpreter führt eine neue Aktivierbarkeitsprüfung durch und stellt fest, daß die automatischen Instanzen *"Protokollarchivierung"* und *"Tokenfüller"* aktiviert werden können. Für die Abarbeitung der automatischen Instanz *"Protokollarchivierung"* startet der Geschäftsprozeß-Interpreter einen C-Interpreter, der das erstellte Prüfungsprotokoll archi-

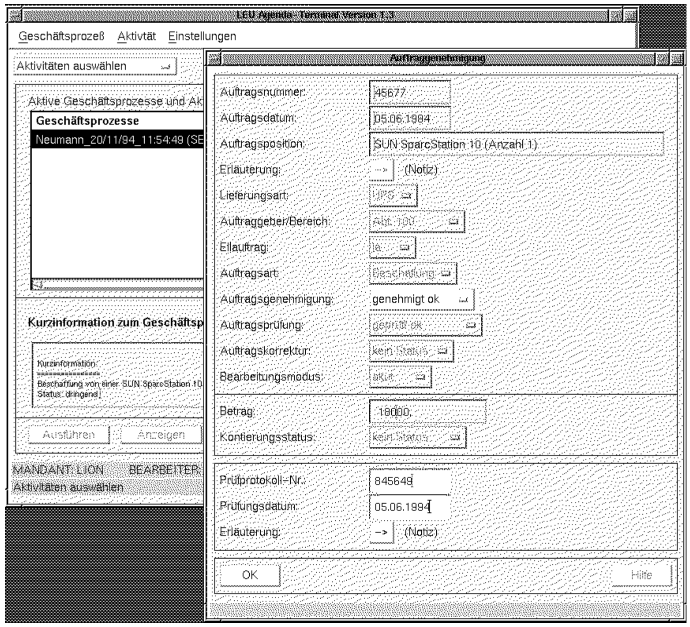


Abbildung 8.14: Bearbeitung des Dialogs *"Beschaffung genehmigen"* durch Frau Neumann

viert. Nach Beendigung dieser Aktivität wird ein Token vom Typ *"Nachricht"* in den Nachbereichskanal *"Ende Archivierung"* geschrieben. Die automatische Instanz *"Tokenfüller"* liest den genehmigten Auftrag aus ihrem Vorbereichskanal und schreibt ihn in die Nachbereichskanäle *"Auftrag ohne Lieferant"* und *"zu archivierender Auftrag"* (siehe Abb. 8.1 auf Seite 112). Der Kanal *"zu archivierender Auftrag"* liegt im Vorbereich der *verfeinerten Instanz "Auftragsarchivierung"*. Zu dieser Instanz gehört das in Abbildung 8.2 auf Seite 113 dargestellte Subnetz. Der Kanal *"zu archivierender Auftrag"* liegt dort im Vorbereich der automatischen Instanz *"selektiere Prüfungsprotokolle"*. Diese Instanz ist jedoch erst aktivierbar, wenn auch der $\xrightarrow{2.2}$*Systemkanal "erstellte Zahlungsbestätigung"* gefüllt worden ist.

Der Kanal *"Auftrag ohne Lieferant"* liegt im Vorbereich der *verfeinerten Instanz "Lieferant benachrichtigen"*. Zu dieser Instanz gehört das in Abbildung 8.3 auf Seite 113 dargestellte Subnetz. Der Kanal *"Auftrag ohne Lieferant"* liegt dort im Vorbereich der manuellen Instanz *"Lieferant auswählen"*.

Eine vom Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Auftragserteilung"* durchgeführte Aktivierbarkeitsprüfung ergibt, daß derzeit die Instanz *"Lieferant auswählen"* aktivierbar ist. Diese Instanz wird dem Agenden-Controller mit der Nachricht *KSB_VI_ACB_AGENCY* übermittelt. Der Agenden-Controller überprüft, ob eine derzeit an der Ausführung des Geschäftsprozesses beteiligte Person zur Bearbeitung dieser Aktivität berechtigt ist. Die Überprüfung ergibt, daß Herr Nowak diese Aktivität bearbeiten darf. Der Agenden-Controller übermittelt die Aktivität somit an das Agenda-Terminal von Herrn Nowak.

Nun startet Herr Riehl sein Agenda-Terminal und loggt sich mit seiner Rolle *Buchhalter* in die LEU-Ausführung ein. Er wird mit dem Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* verbunden (siehe Anmeldung Frau Hofmann). Der Agenden-Controller überprüft, ob Herr Riehl zur Mitarbeit im Geschäftsprozeß berechtigt ist, und stellt fest, daß er die Aktivität *"Vorkontierung"* ausführen darf. Das Ergebnis dieser Berechtigungsüberprüfung meldet er an das Agenda-Terminal von Herrn Riehl zurück. Daraufhin wird der Geschäftsprozeß in die Datenhaltung des Agenda-Terminals von Herrn Riehl eingetragen und an der Oberfläche angezeigt. Da derzeit keine manuellen Aktivitäten, zu deren Bearbeitung Herr Riehl berechtigt ist, im Geschäftsprozeß zur Bearbeitung anstehen, wird dem Agenda-Terminal von Herrn Riehl keine Aktivität übermittelt.

Abbildung 8.15 gibt den derzeitigen Stand der Agenden von allen an der Ausführung beteiligten Personen wider.

Herr Nowak startet nun die Aktivität *"Lieferant auswählen"*. Beim Starten bekommt er die zum Auftrag erfaßten Daten angezeigt (Abb. 8.16). Die Aktivität wird von Herrn Nowaks Agenda-Terminal mit der Nachricht *KSB_TI_ACB_START_AGENCY* an den Agenden-Controller zum Starten übermittelt. Der Agenden-Controller überprüft, ob diese Aktivität an mehrere Benutzer zur Auswahl übermittelt worden ist und somit nun aus den anderen Agenden gelöscht werden muß. Da Herr Nowak die Aktivität *"Beschaffung genehmigen"* jedoch als einziger in seiner Agenda stehen hatte, kann der

Abbildung 8.15: Stand der Agenden (4)

Agenden-Controller die Aktivität direkt an den Geschäftsprozeß-Interpreter über die Nachricht *KSB_ACB_VI_START_AGENCY* weiterreichen. Dieser startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Der Dialog-Interpreter lädt den zur Aktivität *"Lieferant auswählen"* gehörenden Dialog und stellt ihn auf dem Rechner von Herrn Nowak dar (siehe Abb. 8.17).

Im zur Aktivität *"Lieferant auswählen"* gehörenden Dialog wählt Herr Nowak *"Herrn Rudolph Hansen - Branche Hardware"* mit der Lieferantennummer *65349* als Lieferant für die zu beschaffende *SUN SparcStation 10* aus. Nach Beendigung des Dialogs wird der Auftrag in den Kanal *"zu*

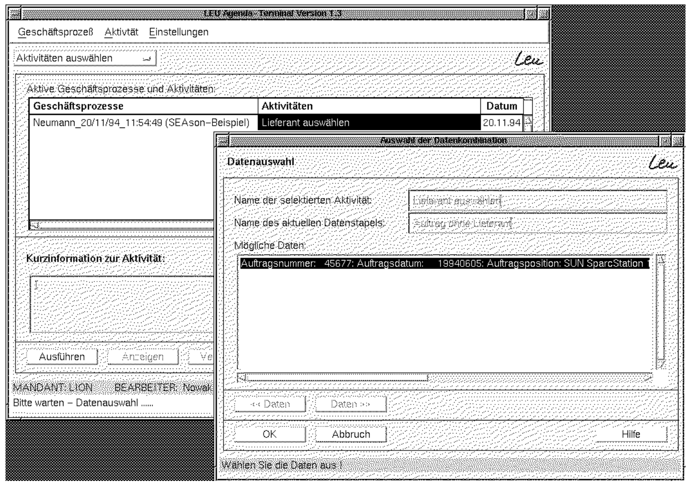


Abbildung 8.16: Auswahl der Aktivität *"Lieferant auswählen"* durch Herrn Nowak

versendender Auftrag" geschrieben. Die Beendigung der Aktivität *"Lieferant auswählen"* wird dem Agenden-Controller über die Nachricht *KSB_VI_ACB_AGENCY_DONE* mitgeteilt. Dieser leitet diese Mitteilung über die Nachricht *KSB_ACB_TI_DELETE_AGENCY* an das Agenda-Terminal von Herrn Nowak weiter. Nach dem Erhalt dieser Nachricht löscht das Agenda-Terminal die Aktivität aus seiner internen Datenhaltung.

Die vom Geschäftsprozeß-Interpreter ausgeführte Aktivierbarkeitsprüfung ergibt, daß nun die automatische Aktivität *"Versendung"* ausgeführt werden kann (siehe Abb. 8.3 auf Seite 113). Zur Bearbeitung dieser Instanz wird ein neuer C-Interpreter gestartet. Die an die automatische Instanz angebundene Funktion erzeugt einen Ausdruck des Auftrags. Nachdem der Ausdruck erstellt worden ist, beendet sich der C-Interpreter und der Auftrag wird in den Kanal *"versandter Auftrag"* geschrieben.

Die erneut durch den Geschäftsprozeß-Interpreter durchgeführte Aktivierbarkeitsprüfung ergibt, daß nun die manuelle Instanz *"Vorkontierung"* aktivierbar ist. Diese Aktivität wird nun an den Agenden-Controller übermittelt (Vorgehensweise siehe oben). Der Agenden-Controller ermittelt die

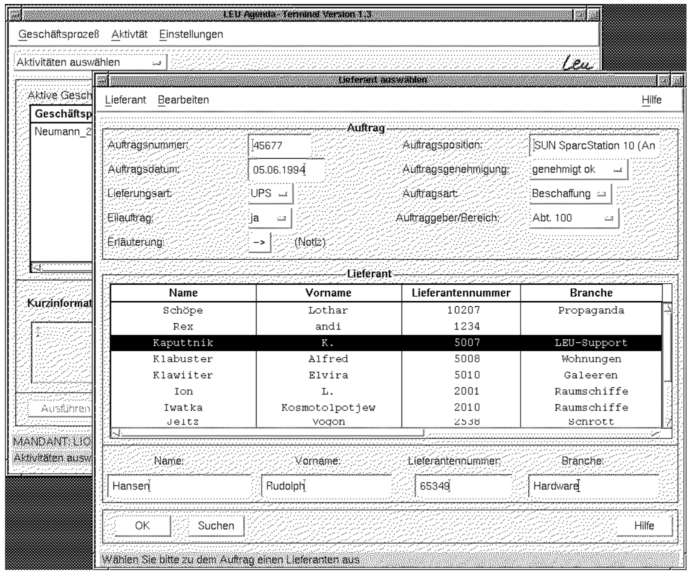
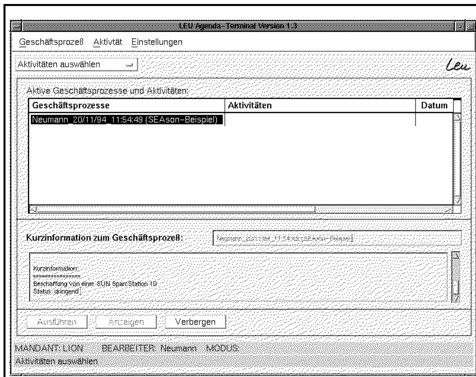


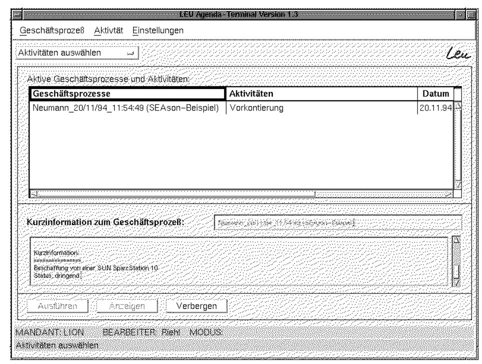
Abbildung 8.17: Bearbeitung des Dialogs *"Lieferant auswählen"* durch Herrn Nowak

zur Bearbeitung dieser Aktivität berechtigten derzeit eingeloggten Personen und stellt fest, daß die Aktivität *"Vorkontierung"* von Herrn Riehl bearbeitet werden darf. Er übermittelt die Aktivität an das Agenda-Terminal von Herrn Riehl (Vorgehensweise siehe oben).
Abbildung 8.18 gibt den derzeitigen Stand der Agenden aller an der Ausführung beteiligten Personen wider.

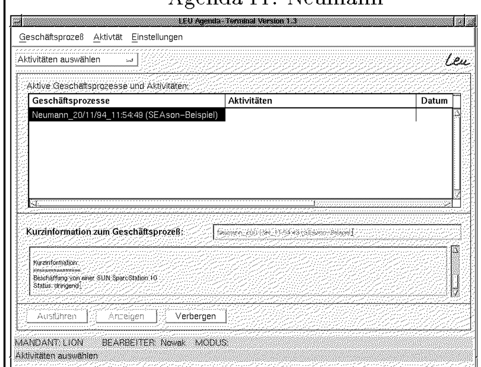
Herr Riehl startet nun die Aktivität *"Vorkontierung"*. Beim Starten bekommt er die zum Auftrag erfaßten Daten angezeigt (Abb. 8.19). Hierauf wird die Aktivität vom Agenda-Terminal von Herrn Riehl über den Agenden-Controller an den Geschäftsprozeß-Interpreter weitergeleitet (Vorgehensweise siehe oben). Der Geschäftsprozeß-Interpreter startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Der Dialog-Interpreter lädt den zur Aktivität *"Vorkontierung"* gehörenden Dialog und stellt ihn auf dem Rechner von Herrn Riehl dar (siehe Abb. 8.20).



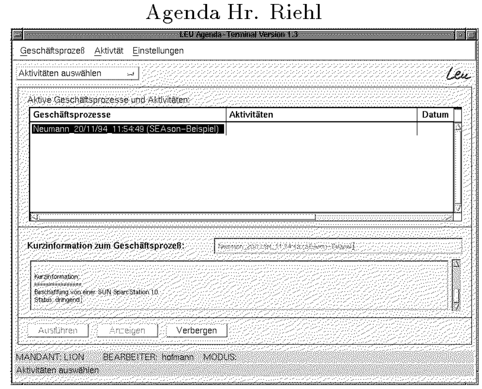
Agenda Fr. Neumann



Agenda Hr. Riehl



Agenda Hr. Nowak



Agenda Fr. Hofmann

Abbildung 8.18: Stand der Agenden (5)

Im zur Aktivität *"Vorkontierung"* gehörenden Dialog wird Herr Riehl der versandte Auftrag angezeigt. Anhand dieser Auftragsdaten nimmt Herr Riehl nun eine Vorkontierung des Auftrags vor. Nachdem Herr Riehl den Dialog über den OK-Button beendet hat, wird der vorkontierte Auftrag in den $\xrightarrow{2.2}$Systemkanal *"vorkontierter Auftrag"* geschrieben.

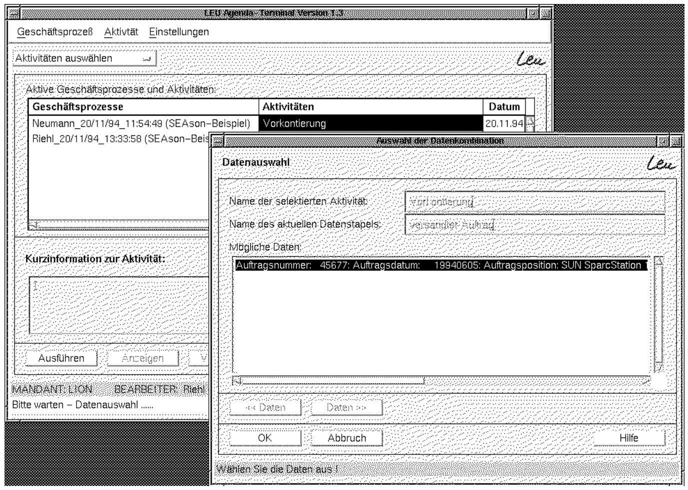


Abbildung 8.19: Auswahl der Aktivität *"Vorkontierung"* durch Herrn Riehl

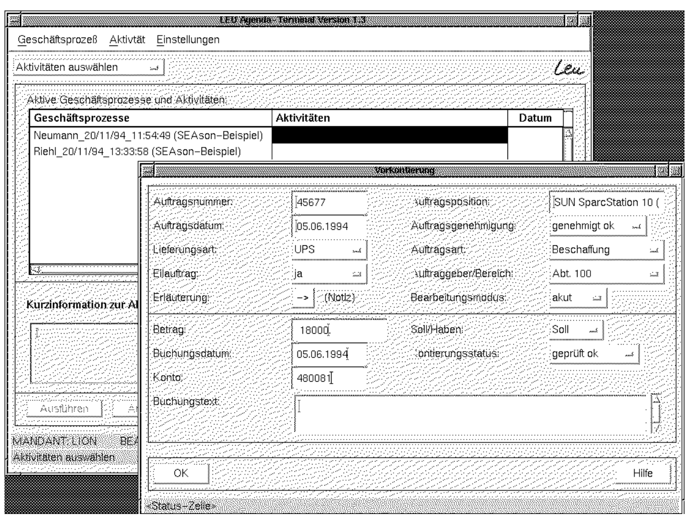


Abbildung 8.20: Bearbeitung des Dialogs *"Vorkontierung"* durch Herrn Riehl

Des weiteren wird ein Objekt vom Typ *Nachricht* erzeugt und in den Kanal *"Ende Auftragsbearbeitung"* geschrieben (siehe Abb.8.3 auf Seite 113). Die Beendigung der Aktivität *"Vorkontierung"* wird dem Agenda-Terminal von Herrn Riehl mitgeteilt (Vorgehensweise siehe oben), worauf die Aktivität aus der Datenhaltung des Agenda-Terminals gelöscht wird.

Herr Riehl startet nun zum Geschäftsprozeßmodell *"Rechnungsbearbeitung"* einen mehrbenutzerfähigen Geschäftsprozeß (siehe Abb. 8.21).

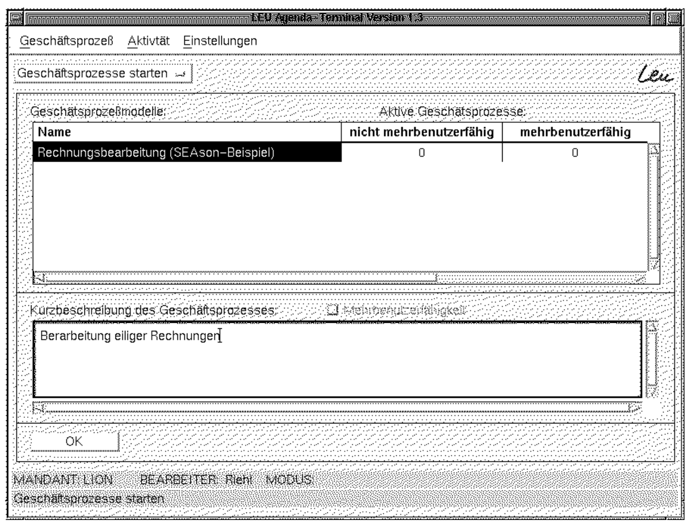


Abbildung 8.21: Herr Riehl startet zum Geschäftsprozeßmodell *"Rechnungsbearbeitung"* einen neuen Geschäftsprozeß

Hierbei gibt er als Kurzinformation an, daß mit diesem Geschäftsprozeß eilige Rechnungen bearbeitet werden sollen.
Der LEU-Controller wird über den neu zu startenden Geschäftsprozeß informiert, startet einen Agenden-Controller und teilt diesem über die IPC-Nachricht *KSB_LC_ACB_START_INC* alle benötigten Informationen mit. Der Agenden-Controller startet daraufhin den Geschäftsprozeß-Interpreter und übermittelt ihm mit der IPC-Nachricht *KSB_ACB_VI_START_NEW_PROCESS* die Informationen, die er benötigt, um den Geschäftsprozeß zu starten. Nachdem der Geschäftsprozeß-Interpreter den

neuen Geschäftsprozeß erfolgreich gestartet hat, sendet er die Nachricht *KSB_VI_ACB_PROCESS_STARTED* an den Agenden-Controller und beginnt danach mit der Interpretation des Ablaufmodells. Nachdem der Agenden-Controller vom Geschäftsprozeß-Interpreter die Nachricht *KSB_-VI_ACB_PROCESS_STARTED* empfangen hat, unterrichtet er mit der Nachricht *KSB_ACB_TI_NEW_INC* alle Agenda-Terminals über die Existenz des neuen Geschäftsprozesses. Die Agenda-Terminals melden sich daraufhin unter Angabe der Benutzerrollen beim Agenden-Controller des neuen Geschäftsprozesses an (Vorgehensweise siehe oben). Der Agenden-Controller überprüft für jeden Mitarbeiter, der sich bei ihm anmeldet, ob er zur Mitarbeit am Geschäftsprozeß berechtigt ist, und meldet das Ergebnis dieser Berechtigungsüberprüfung an das jeweilige Agenda-Terminal zurück (Vorgehensweise siehe oben). Da nur Herr Riehl zur Mitarbeit im von ihm gestarteten Geschäftsprozeß berechtigt ist, wird der Geschäftsprozeß nur in die Datenhaltung von Herrn Riehls Agenda-Terminal eingetragen und die Oberfläche des Agenda-Terminals aktualisiert. Der zur Anzeige gebrachte Geschäftsprozeßname ist hierbei *"Riehl_20/11/94_13:33:58(SEAson-Beispiel)"*.

Der Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Rechnungsbearbeitung"* stellt bei seiner Aktivierbarkeitsprüfung fest, daß die automatische Instanz *"Archivierung vorkontierter Auftrag"* aktivierbar ist, da im $\xrightarrow{2.2}$Systemkanal *"vorkontierter Auftrag"* der von Herrn Riehl zuletzt bearbeitete Auftrag liegt (siehe Abb. 8.4 auf Seite 114). Die Information, daß im $\xrightarrow{2.2}$Systemkanal eim Auftrag liegt, wurde beim Starten des Geschäftsprozesses aus der Datenbank ausgelesen (Der Geschäftsprozeß *"Auftragserteilung"* hat den Auftrag dort hinterlegt, als er in den Systemkanal hineingeschrieben wurde). Der Geschäftsprozeß-Interpreter startet für die Abarbeitung dieser automatischen Instanz einen C-Interpreter und teilt ihm die zur Bearbeitung der Instanz benötigten Daten über eine IPC-Nachricht mit. Nachdem der vorkontierte Auftrag bei der Abarbeitung der, an diese Instanz angebundene, Funktion archiviert worden ist, beendet sich der C-Interpreter. Die Ausführung der Instanz ist beendet, und der vorkontierte Auftrag wird in den Kanal *"vork. Auftrag"* geschrieben (siehe Abb. 8.4 auf Seite 114).

Der Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Rechnungsbearbeitung"* führt erneut eine Aktivierbarkeitsprüfung durch und stellt fest, daß die manuelle Aktivität *"Rechnungsbearbeitung"* ausführbar ist. Er übermittelt diese Aktivität an den Agenden-Controller des Geschäftsprozesses *"Rechnungsbearbeitung"* (Vorgehensweise siehe oben). Der Agenden-Controller überprüft, ob Herr Riehl, der einzige zur Mitarbeit an diesem Geschäftsprozeß berechtigte Mitarbeiter, zur Bearbeitung dieser Aktivität berechtigt ist. Da Herr Riehl alle Aktivitäten innerhalb des Geschäftsprozesses bearbeiten darf, übermittelt der Agenden-Controller die manuelle Aktivität *"Rechnungsbearbeitung"* an das Agenda-Terminal von Herrn Riehl.

Abbildung 8.22: Stand der Agenden (6)

Abbildung 8.22 gibt den derzeitigen Stand der Agenden aller an der Ausführung beteiligten Personen wider.

Herr Riehl startet nun die Aktivität *"Rechnungsbearbeitung"*. Beim Starten bekommt er die zum Auftrag erfaßten Daten angezeigt (Abb. 8.23).

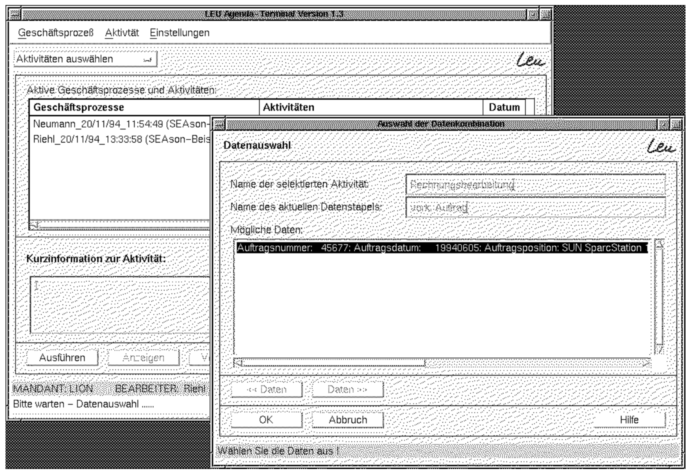


Abbildung 8.23: Auswahl der Aktivität *"Rechnungsbearbeitung"* durch Herrn Riehl

Hierauf wird die Aktivität vom Agenda-Terminal von Herrn Riehl über den Agenden-Controller an den Geschäftsprozeß-Interpreter weitergeleitet (Vorgehensweise siehe oben). Der Geschäftsprozeß-Interpreter startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Zur Aktivität *"Rechnungsbearbeitung"* gehört ein Dialogflow (bestehend aus zwei Fenstern). Der Dialog-Interpreter lädt diesen Dialogflow und stellt ihn auf dem Rechner von Herrn Riehl dar (siehe Abb. 8.24).
Der Dialogflow besteht aus den Dialogfenstern *Rechnungsprüfung* (Startfenster) und *Zahlungsanweisung*. Im Dialogfenster *Rechnungsprüfung* vergleicht Herr Riehl die Positionen des Auftrags mit denen der zugehörigen Rechnung, die über den Pushbutton *Suchen* zur Anzeige gebracht und selektiert werden muß (Die Rechnung für den Auftrag ist schon eingegangen und mit einem anderen Geschäftsprozeß im System erfaßt worden).
Hierauf öffnet sich das zweite Fenster des Dialogflows (siehe Abb. 8.25). Im Dialogfenster *Zahlungsanweisung* erfolgt die Auszahlung an den Lieferanten. Herr Riehl veranlaßt die ausgemachte Zahlung von *18.000,- DM*

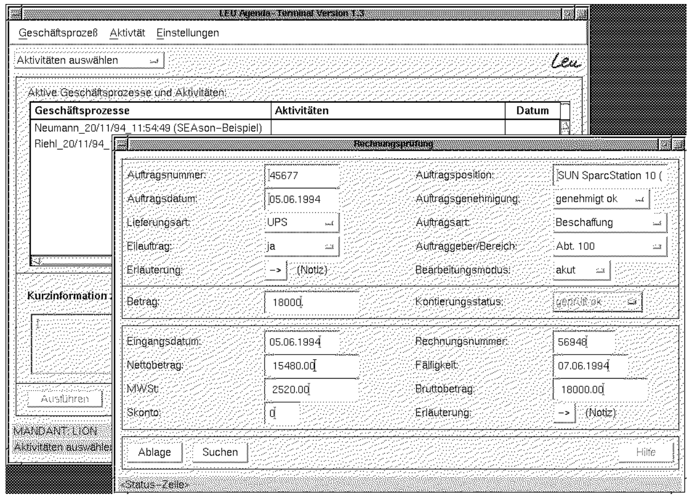


Abbildung 8.24: Bearbeitung des Dialogs *"Rechnungsbearbeitung (1)"* durch Herrn Riehl

an den Lieferanten. Nach Betätigung des *OK*-Buttons ist der Dialogflow komplett bearbeitet und der Nachbereich der Instanz *"Rechnungsbearbeitung"* wird gefüllt. Da die Rechnung zu Beginn dieser Aktivität schon im System erfaßt gewesen ist, muß der Auftrag nicht erneut vorgelegt werden. Darum wird ein Objekt vom Typ *"Zahlungsbestaetigung"* erzeugt und in den $\xrightarrow{2.2}$Systemkanal *"erstellte Zahlungsbestätig."* geschrieben (siehe Abb. 8.4 auf Seite 114). Das Füllen des $\xrightarrow{2.2}$Systemkanals wird vom Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Rechnungsbearbeitung"* über die Nachricht *KSB_VI_ACB_SYSTEM_CHANNEL* an den zugehörigen Agenden-Controller übermittelt. Der Agenden-Controller leitet diese Information über die Nachricht *KSB_ACB_LC_SYSTEM_CHANNEL* an den LEU-Controller weiter. Der LEU-Controller leitet diese Information mit der Nachricht *KSB_LC_ACB_SYSTEM_CHANNEL* an die Agenden-Controller aller anderen Geschäftsprozesse weiter. Auf diese Weise erhält der Agenden-Controller des Geschäftsprozesses *"Auftragserteilung"* Kenntnis über das Füllen des Systemkanals *"erstellte Zahlungsbestätig."*. Er reicht diese Information an den Geschäftsprozeß-Interpreter weiter. Nachdem auf diese Weise der Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Auftragserteilung"* über das Füllen des Systemkanals *"erstellte Zahlungs-*

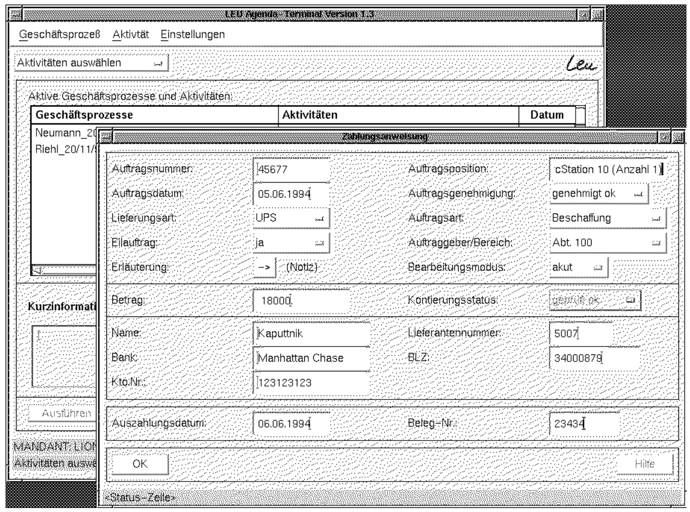


Abbildung 8.25: Bearbeitung des Dialogs *"Rechnungsbearbeitung (2)"* durch Herrn Riehl

bestätig." informiert worden ist, führt der Geschäftsprozeß-Interpreter eine Aktivierbarkeitsprüfung durch und stellt fest, daß die automatische Aktivität *"selektiere Prüfungsprotokolle"* aktivierbar ist (siehe Abb. 8.2 auf Seite 113). Er startet für diese Aktivität einen neuen C-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Durch diese automatische Instanz werden alle zu diesem Auftrag bisher erstellten Protokolle selektiert. Nach der Selektion ist diese Aktivität beendet, der C-Interpreter beendet sich und der Auftrag wird in den Kanal *"erledigter Auftrag"* geschrieben. Der Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Auftragserteilung"* führt daraufhin erneut eine Aktivierbarkeitsprüfung durch und stellt fest, daß nun die manuelle Instanz *"Erstellung Auftragsprotokoll"* aktivierbar ist (siehe Abb. 8.2 auf Seite 113). Die Instanz wird an den zugehörigen Agenden-Controller übermittelt (Vorgehensweise siehe oben). Dieser überprüft, welche der derzeit am Geschäftsprozeß teilnehmenden Personen zur Bearbeitung dieser Aktivität berechtigt sind, und stellt fest, daß nur Herr Nowak diese Aktivität bearbeiten darf. Der Agenden-Controller leitet die Aktivität somit an das Agenda-Terminal von Herr Nowak weiter.

Abbildung 8.26: Stand der Agenden (7)

Abbildung 8.26 gibt den derzeitigen Stand der Agenden aller an der Ausführung beteiligten Personen wider.

Herr Nowak startet nun die Aktivität *"Erstellung Auftragsprotokoll"*. Beim Starten bekommt er die zum Auftrag erfaßten Daten angezeigt (Abb. 8.27). Hierauf wird die Aktivität vom Agenda-Terminal von Herrn Nowak über den Agenden-Controller an den Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Auftragserteilung"* weitergeleitet (Vorgehensweise siehe oben). Der Geschäftsprozeß-Interpreter startet für die Bearbeitung der Aktivität einen neuen Dialog-Interpreter und übermittelt ihm alle für die Ausführung der Aktivität benötigten Daten. Der Dialog-Interpreter lädt den zu dieser Aktivität gehörenden Dialog und stellt ihn auf dem Rechner von Herrn Nowak

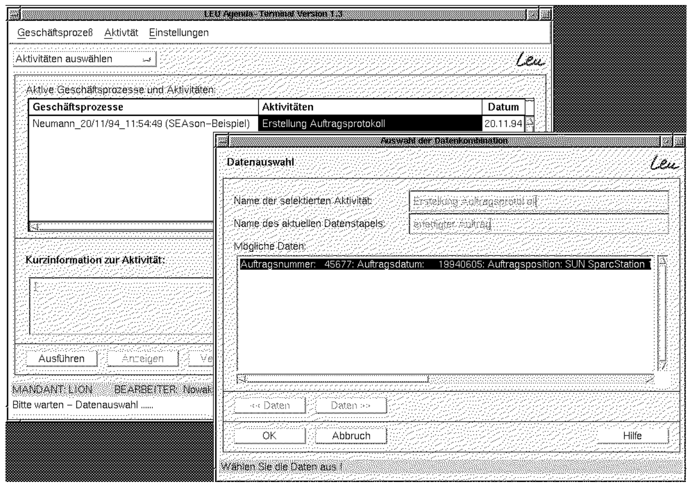


Abbildung 8.27: Auswahl der Aktivität *"Erstellung Auftragsprotokoll"* durch Herrn Nowak

dar (siehe Abb. 8.28).

Herr Nowak erstellt mit diesem Dialog das Schluß-Protokoll des Auftrags. Nachdem Herr Nowak den Dialog über Betätigung des *OK*-Buttons beendet hat, wird das erstellte Auftragsprotokoll in den Kanal *"erstelltes Auftragsprotokoll"* geschrieben (siehe Abb. 8.2 auf Seite 113).

Eine vom Geschäftsprozeß-Interpreter des Geschäftsprozesses *"Auftragserteilung"* durchgeführte Aktivierbarkeitsprüfung ergibt, daß nun die automatische Aktivität *"AP-Archivierung"* ausgeführt werden kann. Für diese Aktivität startet der Geschäftsprozeß-Interpreter einen neuen C-Interpreter. Dieser führt die zu dieser Aktivität gehörende Funktion aus. Innerhalb dieser Funktion wird dann der gesamte abgearbeitete Auftrag archiviert. Nach Beendigung der Archivierung beendet sich der C-Interpreter, ein Objekt vom Typ *Nachricht* wird erzeugt und in den Kanal *"Ende AP-Archivierung"* geschrieben.

Danach ist die Auftragsbearbeitung beendet und keine Aktivität mehr ak-

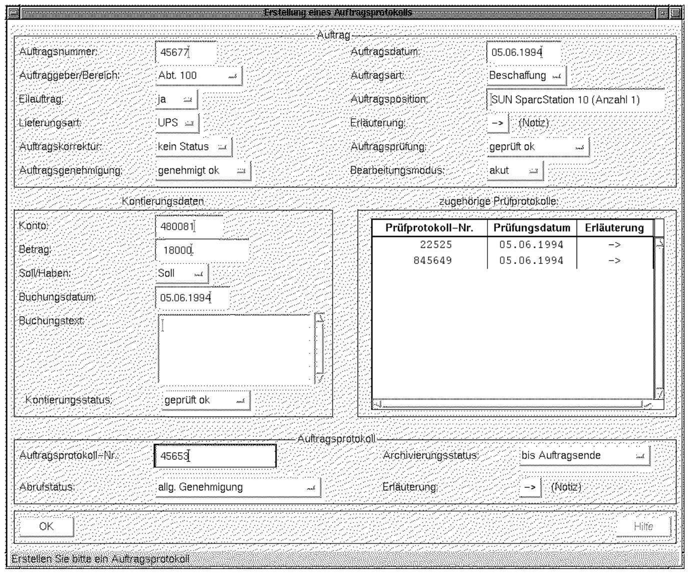


Abbildung 8.28: Bearbeitung des Dialogs *"Erstellung Auftragsprotokoll"* durch Herrn Nowak

tivierbar.

Literaturverzeichnis

[Bar90] R. Barker. *CASE*Method Entity Relationship Modelling.* Addison-Wesley, Wokingham, England, 1990.

[BFGG92] S. Bandinelli, A. Fugetta, C. Ghezzi, und S. Grigolli. *Process Enactment in SPADE.* Aus J.-C. Derniame, Hrsg., *Software Process Technology - Proceedings of the* 2^{nd} *European Software Process Modeling Workshop*, Seiten 67–83, Trondheim, Norway, September 1992. Springer. Appeared as Lecture Notes in Computer Science 635.

[Chr92] G. Chroust. *Modelle der Software-Entwicklung.* Oldenbourg, München, Wien, 1992.

[CMD91] C.Ghezzi, M.Jazayeri, und D.Mandrioli. *Fundamentals of Software Engineering.* Prentice-Hall, Englewood Cliffs, N.J., 1991.

[Dav93] T. Davenport. *Process Innovation - Reengineering Work through Information Technology.* Harvard Business School Press, Boston, US, 1993.

[DG90] W. Deiters und V. Gruhn. *Managing Software Processes in MELMAC.* Aus *Proceedings of the Fourth ACM SIGSOFT Symposium on Software Development Environments*, Seiten 193–205, Irvine, California, USA, December 1990. ACM Software Engineering Notes, 15(6), December 1990.

[DG91] W. Deiters und V. Gruhn. *Software Process Model Analysis Based on FUNSOFT Nets. Mathematical Modeling and Simulation*, (8), May 1991.

[DK94] W. Deiters und R. Karl. *Workflow Management – Groupware Computing.* dsk Beratungs-GmbH für Bürokommunikation, Pfaffenhofen, 1994.

[EG91] W. Emmerich und V. Gruhn. FUNSOFT Nets: A Petri–Net based Software Process Modeling Language. Aus *Proc. of the 6th International Workshop on Software Specification and Design*, Como, Italy, 1991.

[EJP+91] W. Emmerich, G. Junkermann, B. Peuschel, W. Schäfer, und S. Wolf. *MERLIN: Knowledge-Based Process Modeling.* Aus A. Fugetta, R. Conradi, und V. Ambriola, Hrsg., *Proceedings of the European Software Process Modeling Workshop*, Milan, Italy, May 1991. AICA Press.

[Gen87] H.J. Genrich. *Predicate/Transition Nets.* Aus W. Brauer, W. Reisig, und G. Rozenberg, Hrsg., *Petri Nets: Applications and Relationships to Other Models of Concurrency*, Berlin, FRG, 1987. Springer. Lecture Notes in Computer Science 254.

[GJLO89] P. Griffiths, P. Jamart, A. Legait, und D.E. Oldfield. *The ALF Approach to Process Modelling.* Aus *ESPRIT '89 Conference Proceedings*, Brussels, Belgium, November 1989. Kluwer Academic Publishers for the Commission of the European Communities.

[God83] H.P. Godbersen. *Funktionsnetze - Eine Modellierungskonzeption zur Entwurfs- und Entscheidungsunterstützung.* Ladewig Verlag, Birkach, 1983.

[Gru91a] V. Gruhn. *The Software Process Management Environment MELMAC.* Aus A. Fugetta, R. Conradi, und V. Ambriola, Hrsg., *Proceedings of the European Software Process Modeling Workshop*, Milan, Italy, May 1991. AICA Press.

[Gru91b] V. Gruhn. *Validation and Verification of Software Process Models.* PhD thesis, University Dortmund, June 1991. Technical Report No. 394/91.

[Gru93a] V. Gruhn. *Entwicklung von Informationssystemen in der LION-Entwicklungsumgebung.* Aus G. Scheschonk und W. Reisig, Hrsg., *Petri-Netze im Einsatz für Entwurf und Entwicklung von Informationssystemen*, Seiten 31–45, Berlin, FRG, September 1993. Springer.

[Gru93b] V. Gruhn. *Software Process Simulation in MELMAC. Systems Analysis - Modelling - Simulation*, 11:121–141, 1993.

[HC93] M. Hammer und J. Champy. *Reengineering the Corporation.* Harper Business, New York, US, 1993.

[HP94] Hewlett Packard GmbH (Veranstalter). Seminarunterlagen zum Seminar *Dokumentenmanagement- und Workflow-Konzepte als wichtige Instrumente für ein Business-Process Re-Engineering*, März 1994. Seminarort: Bad Homburg.

[HRV84] O. Herzog, W. Reisig, und R. Valk. *Petri-Netze: Ein Abriss ihrer Grundlagen und Anwendungen. Informatik Spektrum*, 1984.

[ISA92] ISA GmbH. *Dialog Manager Übersicht. Dokumentation. Release A.02.00*, 1992.

[Jen87] K. Jensen. *Coloured Petri nets. A way to describe and analyse real world system - without drowning in unnecessary details.* Aus *Proc. of the 5th Int. Conf. on Systems Engineering*, Dayton, 1987.

[KPS93] A. Karrer, M.H. Penedo, und C. Shu. *A Survey of Software Engineering Environment Architectural Approaches.* Technical report, TRW, November 1993.

[Leh87] M. Lehman. *Process Models, Process Programs, Programming Support - Invited Response To A Keynote Address By Lee Osterweil.* Aus *Proceedings of the 9th International Conference on Software Engineering*, Monterey, California, 1987.

[Leh89] M. Lehman. *Uncertainty in Computer Application and its Control Through the Engineering of Software.* *Journal of Software Maintenance*, 1(1), September 1989.

[Lon93] J. Lonchamp. *A Structured Conceptual and Terminological Framework for Software Process Engineering.* Aus *Proceedings of the 2^{nd} International Conference on the Software Process - Continuous Software Process Improvement*, Berlin, Germany, February 1993.

[PS92] B. Peuschel und W. Schäfer. *Concepts and Implementation of a Rule-based Process Engine.* Aus *Proceedings of the 14th International Conference on Software Engineering*, Melbourne, Australia, May 1992.

[Sch94] M. Schneider. Diplomarbeit: *Werkzeuge zur Optimierung erweiterter Entity–Relationship Modelle und deren Abbildung in ein relationales Datenbankschema.* Universität Dortmund, August 1994.

[Shn91] B. Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Addison Wesley, New York, 1991.

[Som92] I. Sommerville. *Software Engineering.* Addison-Wesley, fourth Ausgabe, 1992.

[SR92] B. Singh und G.L. Rein. *Role Interaction Nets (RINs): A Process Description Formalism.* Technical Report CT-083-92, MCC, Austin, Texas, US, July 1992.

[Ste92] W.Richard Stevens. *Programmieren von UNIX-Netzen – Grundlagen, Programmierung, Anwendung.* Coedition der Verlage *Carl Hanser* und *Prentice-Hall International*, 1992.

[War89] B. Warboys. *The IPSE 2.5 Project: Process Modelling as the Basis for a Support Environment.* Aus N. Madhavji, W. Schäfer, und H. Weber, Hrsg., *Proceedings of the First International Conference on System Development Environments and Factories*, London, 1989. Pitman Publishing.

[Zuc91] J.-D. Zucker. *The ALF System*. Aus F. Long, Hrsg., *Software Engineering Environments*. Ellis Horwood, September 1991.