# Workflow-Systeme im WWW

**Johann Eder, Herbert Groiss, Walter Liebhart**

Institut für Informatik
Universität Klagenfurt
Klagenfurt, AUSTRIA
*e-mail: {eder, herb, walter}@ifi.uni-klu.ac.at*
*URL: http://www.ifi.uni-klu.ac.at*

**Zusammenfassung**

Die neuen Telekommunikationsmedien und -dienste, insbesondere das Internet mit dem World-Wide Web, verändern die Kostenstruktur für Telekommunikation erheblich. Damit wird ein wesentlicher Parameter für die Gestaltung von inner- und überbetrieblichen Geschäftsprozessen sosehr verändert, daß eine Neuorganisation vielfach angezeigt ist. Zur Unterstützung der Definition und Durchführung von Geschäftsprozessen haben sich Workflow-Managementsysteme bereits vielfach bewährt. Um mit den geänderten Rahmenbedingungen mithalten zu können, müssen moderne Workflow-Managementsysteme aber auch die Möglichkeiten, die die neue Telekommunikationsinfrastruktur anbietet, adäquat nutzen. Hier werden die grundsätzlichen Realisierungs- und Einsatzmöglichkeiten für Workflow-Systeme im World-Wide-Web aufgezeigt und anhand eines Prototyps präsentiert.

# 1 Einleitung

Die beobachtbare Internationaliserung der Geschäftstätigkeit sowie die Beschleunigung und Verdichtung inner- und überbetrieblicher Geschäftsprozesse stellen Unternehmen vor die Herausforderung, neue Systeme der Informations- und Kommunikationslogistik rasch für die Realisierung ihrer Informationssysteme zu nutzen, um im Wettbewerb bestehen zu können bzw. sich Wettbewerbsvorteile zu schaffen. Die neuen Schlagworte wie „lean management“ und „business process reeingineering“ zielen auf Organisationstrukturen, die prozeßhaft entlang der Wertschöpfungskette entworfen werden. Für die Umsetzung, Koordination und Verwaltung von Geschäftsprozessen wurden Workflow-Management-Systeme entwickelt und vor allem im innerbetrieblichen Bereich auch bereits erfolgreich eingesetzt. Die Unterstützung der Geschäftsprozesse durch Workflow-Systeme bringt neben besserer Dokumentation und Prozeßspezifikation insbesondere eine deutlich geringere Durchlaufzeit von Geschäftsprozessen. Dies wird vor allem dadurch erzielt, daß die Transportzeiten von Dokumenten drastisch verringert werden. Durch die gleichzeitige Verfügbarkeit eines elektronischen Dokumentes an mehreren Orten gleichzeitig, sind neue nebenläufige Prozeßstrukturen möglich, die wiederum die Liegezeiten und damit auch die Durchlaufzeiten verringern.

Der sich abzeichnende „Information Superhighway", insbesondere das Internet, verändert die Zeit- und Kostenstruktur für Informationsvertrieb erheblich. Die Verfügbarkeit von Information, der Austausch von beliebigen elektronischen Dokumenten wird erheblich preiswerter und der Zugang zu Information sowie zu entfernten informationstechnischen Einrichtungen wird durch weitverbreitete Standardprotokolle wie das World-Wide-Web (WWW) entscheidend vereinfacht und verbillgt. Für Unternehmen erfordert diese Veränderung wesentlicher Parameter ein Überdenken und vielfach eine Neuorganisation der inner- und überbetrieblichen Geschäftsprozesse (business process reengineering). Insbesondere stellt sich die Forderung nach einer Verknüpfung von Workflow-Managementsystemen und der neuen Kommunikationsinfrastruktur.

Die Ausweitung der Unterstützung von Geschäftsprozessen und die weltweite Kommunikation verbunden mit der dabei entstehenden Hetereogenität verstärkt auch die Forderung nach intelligenten Mechanismen zur Behandlung von Fehlern und Ausnahmen. Dafür werden ausgehend von den von der Datenbanktechnologie her bekannten Transaktionskonzepten für die spezifischen Bedürfnisse von Workflow-Managementsystemen neuartige Workflow-Transaktionen entwickelt, die vor allem darauf abzielen, Ausnahmesituationen möglichst automatisch zu beheben (Recovery) und - vor allem im Fehlerfall - automatisch alternative Prozesse anzustoßen.

In diesem Beitrag wollen wir die grundlegenden Konzepte für die Realisierung von Workflows im World-Wide-Web vorstellen und die Einsatzmöglichkeiten für die Realisierung internationaler Informationssysteme diskutieren. Zur Evaluierung dieser Konzepte wurde ein Workflow-System prototypisch realisiert.

Die hier vorgestellten Lösungsansätze zu diesem Problem gingen aus einem Projekt hervor, das vom Institut für Informatik der Universität Klagenfurt gemeinsam mit der Fa. CSE Systems, einem der führenden Hersteller von Workflow-Systemen, durchgeführt wird.

# 2 Eine Referenzarchitektur für Workflow-Managementsysteme

Die Workflow Management Coalition, eine non-profit Organisation, der u.a. die wichtigsten Hersteller von Workflow-Managementsystemen angehören, hat sich die Standardisierung im Bereich Workflow-Management zum Ziel gemacht. Insbesondere wurde bereits eine Referenzarchitektur für Workflow-Systeme erarbeitet, die die wichtigsten Komponenten solcher Systeme beschreibt und Schnittstellen zwischen diesen Komponenten festlegt. Abb. 1 zeigt diese Referenzarchitektur [Coa94]. Für die Integration von World-Wide-Web und Workflow- Systemen sind natürlich vor allem die Schnittstellen zwischen den Komponenten einer genaueren Betrachtung zu unterziehen.

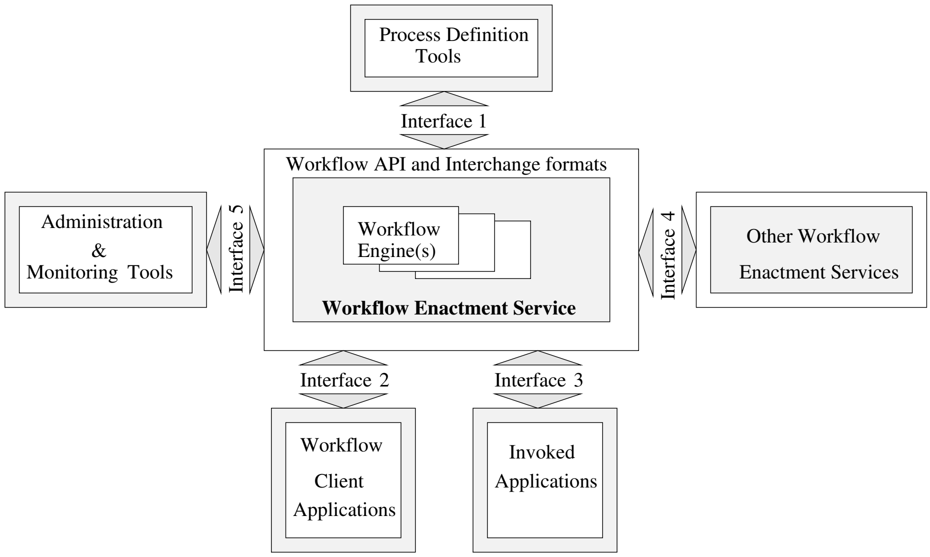


Abbildung 1: **Referenzarchitektur**

Die fünf Schnittstellen zu einem Workflow-System nach der Referenzarchitektur sind wie folgt:

1. Definition von Geschäftsprozessen: Über diese Schnittstelle wird die Struktur der Prozesse an die Workflow Maschine übermittelt, die dann die Durchführung von Exemplaren dieser Prozesse steuert.

2. Kommunikation des Workflow-Systems mit Clients: Das Workflow-System kommuniziert mit den Benutzern über die Workflow- Clients, die insbesondere die anstehenden Aufgaben für die Benutzer anzeigen, sowie die Information über die Durchführung von Aufgaben und erzeugte Dokumente bzw. Daten an das Workflow-System weiterleiten.

3. Kommunikation mit anderen Applikationen: Über diese Schnittstelle kommuniziert ein Workflow-System mit anderen Anwendungsprogrammen und tauscht Daten und Steuerungsinformation aus.

4. Kommunikation zwischen Workflow-Systemen: Komplexe Prozesse, insbesondere überbetriebliche erfordern die Kommunikation zwischen mehreren Workflow-Systemen, die jeweils die Steuerung der Durchführung von Teilen eines Prozesses wahrnehmen.

5. Monitoring und Dokumentation von Geschäftsprozessen: Über diese Schnittstelle wird ein Workflow-System administriert. Außerdem können über diese Schnitstelle Daten über die durchgeführten Prozesse analysiert werden.

Insbesondere die Schnittstellen zwei, vier und fünf müssen auch dahingehend realisiert werden, daß die Kommunikation über internationale Datennetze und verbreitete Kommunikationsprotokolle möglichst offen erfolgen kann.

# 3 Interaktion mit dem World-Wide Web

Die weite Verbreitung von Internet und die neuen Möglichkeiten, die das World Wide Web (WWW) bietet, machen dieses zu einem idealen Medium zum Datenaustausch und zur Client- Server Kommunikation. Das WWW beruht auf drei Grundkonzepten: die einheitliche Adressierung von Informationen im Internet mit dem Uniform Resource Locator (URL), der Darstellung von Information mit der Sprache HTML und der Übertragung über das HTTP Protokoll.

Das *HTML* (HyperText Markup Language) Format [BLC95] bietet die Möglichkeit, unterschiedliche Medientypen, wie Texte, Graphiken, Ton und Video, in einem Dokument zu vereinen. Mittels sogenannter Hyperlinks kann mit Mausklick von einem Dokument zu einem anderen verzweigt werden - komplex stukturierte Informationen können so übersichtlich präsentiert werden. Wichtig für die Interaktion mit dem WWW ist die Möglichkeit der *fill-out forms*, also ausfüllbare Formulare. Sie erlauben es dem Benutzer, Daten an den Server zu schicken. Eine neuere HTML-Erweiterung erlaubt auch das Laden von Files auf den Server.

*HTTP* (HyperText Transfer Protocol [BLFF95]) ist ein sehr einfaches Übertragungsprotokoll von Dokumenten über das Internet. Der Client (Browser) fordert ein Dokument bei einem Server an, indem eine TCP/IP Verbindung zu diesem Server aufgebaut wird und der String „GET *Dokument-URL*“ übertragen wird. Der Server antwortet mit einigen Zeilen Statusinformation und dem Inhalt des Dokuments und schließt die Verbindung. Die Dokument-URL kann auch ein Programmaufruf am Server sein, zurückgeschickt wird dann die Ausgabe des Programms. Außer der Funktion GET gibt es noch POST zum Verschicken von Formularen an den Server.

Im folgenden werden kurz die unterschiedlichen Möglichkeiten der Interaktion mit einem Workflow-System über das WWW beschrieben:

**Initialisierung von Prozessen:** Der Anstoß eines Geschäftsprozesses erfolgt in der Regel von außerhalb des Unternehmens oder der Organisation. Gibt man dem Kunden die Möglichkeit den Prozeß auch selbst mittels einer geeigneten Schnittstelle zu starten, kann viel Zeit und somit Geld gespart weren. Über WWW ist eine einfache Realisierung möglich: der Kunde füllt ein HTML-Formular aus, das automatisch ins Workflow-System weitergereicht wird. Bei Kopplung mit anderen Applikationen, z.B. Online-Katalogen, Lagerverwaltungssystemen u.ä. können dem Benutzer die nötigen Informationen bereitgestellt werden und eine vollständige und korrekte Dateneingabe abgeprüft werden. Sind die Informationen einmal im Workflow-System, können sie dem richtigen Bearbeiter zugeordnet werden.

**Kooperation (Schnittstelle 4):** Laufende Prozesse benötigen meist weitere Interaktion zwischen den Partnern: z.B. bei Rückfragen an den Kunden oder wenn der Kunde zusätzliche Information schickt. Dies kann durch Austausch von HTML Dokumenten zwischen den Systemen erfolgen. Jedes teilnehmende Workflow-System muß nur definieren, welche Prozesse beim Empfangen welcher Formulartypen angestoßen oder fortgesetzt werden. Bei Verwendung von standardisierten Dokumenten, wie den UN/EDIFACT Standard [Rap95], wird ein hoher Grad an Interoperabilität erreicht. Das Versenden von Dokumenten an Partner, die über kein Workflow-System verfügen, erfolgt über email. Bei Verwendung eines Mailprogramms, das HTML interpretieren kann, wie z.B. dem Netscape Browser [Net95], kann der Benutzer das in der email enthaltene Formular direkt lesen, ausfüllen und wieder zurückschicken. Bei verteilter Bearbeitung von Prozessen kann der gleiche Mechanismus - das Austauschen von HTML-Formularen - verwendet werden. Die Daten eines Prozesses werden zur Weiterverarbeitung an ein anderes Workflow-System weitergeschickt.

**Monitoring (Schnittstelle 5):** Jeder, der einen Prozeß initialisert oder daran teilgenommen hat, sollte die Möglichkeit haben, sich über den weiteren Fortgang des Prozesses zu informieren. Darüberhinaus muß der Systemadministrator den Zustand aller laufenden Prozesse, der Dokumente, etc. abfragen können. Dies leistet im allgemeinen eine Monitoring-Komponente. Die Verwendung von WWW ermöglicht erstens einen weltweiten Zugriff auf die Daten, zweitens eröffnen die Hypertext-Darstellungsmöglichkeiten von HTML ganz neue Möglichkeiten bei der Präsentation der Daten: Zum Beispiel können die Benutzer durch Prozeßbeschreibungen mit Links zu den beteiligten Personen oder Tasks browsen. Somit können die Vorgänge im Workflow-System auch für Kunden transparent gemacht werden. Durch Einbindung von Graphiken können übersichtliche Darstellungen für Arbeitslisten o.ä. gefunden werden.

**Benutzerschnittstelle (Schnittstelle 2):** Die völlige Integration des Workflow-Systems mit dem WWW erreicht man, indem WWW Browser auch als Standard-Benutzerschnittstelle verwendet werden. Damit erreicht man mehrere Ziele:

- Unterstützung unterschiedlicher Plattformen: Da WWW Browser bereits auf praktisch allen Hard- und Softwareplattformen verfügbar sind, ist größtmögliche Unabhängigkeit gewährleistet. Außerdem fallen Kosten für Installation, Wartung und Administration von verschiedenen Client-Implementierungen weg.

- Kurze Einarbeitungszeit: Die meisten potentiellen Benutzer des Workflow-Systems werden den Umgang mit einem Web Browser bereits beherrschen. Somit sinkt die Schwellenangst sowie die Einarbeitungszeit in das Workflow-System.

- Mobiles Arbeiten: Die Unterstützung unterschiedlicher Plattformen, das Wegfallen der Installation von Software auf der Clientseite, sowie die lose Kopplung zwischen Server und Client erleichtert auch das verteilte Bearbeiten von Aufgaben.

In Abschnitt 5 wird am Workflow-System Panta Rhei gezeigt, wie diese Interaktionsmöglichkeiten konkret realisiert werden können.

# 4 Workflow Transaktionen

Geschäftsprozesse müssen als eine Einheit gesehen werden, die aus verschiedenen Teilaktivitäten bestehen, wobei jede einzelne Teilaktivität zum Erfolg eines Gesamtprozesses beiträgt. Dies wirft natürlich sehr schnell die Frage auf, was denn passiert, wenn eine Teilaktivität scheitert oder aufgrund von Netzproblemen überhaupt nicht gestartet werden kann. Bei einfachen und wenig komplexen Prozessen stellt dies kein wirkliches Problem dar, denn die (wenigen) Ausnahme- und Fehlerfälle werden entweder bereits vom Workflow Designer zur Modellierungszeit entsprechend modelliert oder direkt zur Laufzeit vom Benutzer behandelt. Dies ändert sich jedoch sehr schnell, wenn es sich um große, komplexe Prozesse (z.B. im Telekommunikationsbereich, im Gesundheitswesen, im Finanzbereich etc.) handelt, deren Teilaktivitäten durch Applikationen realisiert werden, die auf heterogenen, autonomen und / oder verteilten Systemen ausgeführt werden. Zusätzlich muß davon ausgegangen werden, daß im Rahmen dieser automatischen Aktivitäten - im Gegensatz zu den meisten manuellen Aktivitäten - intensive Datenverarbeitung (Manipulation von Datenbeständen in unterschiedlichsten Datenbanken) stattfindet. Unter solchen Voraussetzungen sollte das Workflow-System möglichst automatisch entscheiden, ob das Scheitern einer Teilaktivität zum Mißerfolg des gesamten Prozesses führt, ob die Aktivität noch einmal wiederholt werden muß, ob vielleicht einige andere Aktivitäten aus Korrektheitsgründen zurückgesetzt werden müssen oder ob ganz regulär mit der Abarbeitung der verbleibenden Aktivitäten fortgefahren werden kann. Leider besitzen heute am Markt befindliche Workflow-Systeme kaum die Fähigkeit solche und ähnliche Fehler- bzw. Ausnahmesituationen zuverlässig, effizient (wenn möglich automatisch) und korrekt zu behandeln [GHS95].

Als Basistechnologie zur Lösung der zuvor diskutierten Problembereiche scheinen die Transaktionskonzepte aus dem Datenbankbereich ein adäquater Ansatz zu sein. Diese sind jedoch aus verschiedensten Gründen nur bedingt wiederverwendbar, was die Entwicklung von sogenannten Workflow-Transaktionen erfordert. Die Idee der Workflow-Transaktionen besteht nun darin, einige der vielversprechenden Eigenschaften daten-orientierter (erweiterter) Transaktionsmodelle [GR93, Elm92] in prozeß-orientierte Workflow-Systeme zu integrieren, um damit die zuverlässige Abarbeitung von fehlertoleranten Geschäftsprozessen im Mehrbenutzerbetrieb zu erreichen.

## 4.1 Fehlerquellen und Fehlertypen

Erfahrungen aus der Praxis belegen, daß die Behandlung von Fehler- und Ausnahmesituationen sehr hohe Kosten verursachen. Um nun Lösungsanätze für diese Problematik erarbeiten zu können, werden zuvor mögliche Fehlerquellen identifiziert und relevante Fehlertypen einer workflow-basierten Prozeßbearbeitung aufgezeigt. Mögliche Fehlerquellen sind: (1) Fehler im Workflow-System (der Workflow-Engine), (2) Aktivitätsfehler sowie (3) Kommunikationsfehler).

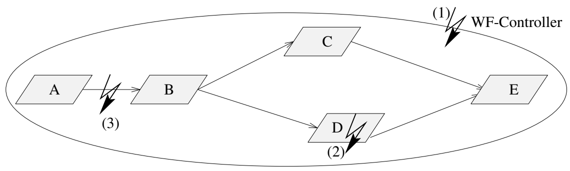


Abbildung 2: **Fehlerquellen**

*Workflow-System (Engine-) Fehler* (z.B. ein Systemabsturz) bewirken ein abnormales Ende sämtlicher Prozesse die gerade aktiv waren. Ziel eines adäquaten Fehlerbehandlungsmechanismus muß es nun sein, nach einem erfolgreichen Wiederanlauf des Systems alle unterbrochenen Prozesse in einen konsistenten Zustand zu bringen, um danach automatisch mit der Prozeßweiterbearbeitung fortfahren zu können. Zusätzlich gilt, daß dabei möglichst wenig bereits getätigte Arbeit verloren gehen soll. Die Wiederherstellung eines konsistenten Workflow-Systems nach einem Workflow-Engine Fehler ist wesentlich einfacher, wenn das System selbst auf einer Datenbank basiert (wie z.B. Panta Rhei [EGN94]) und somit die von der Datenbank vorgesehenen Fehlerbehandlungsmechanismen direkt genutzt werden können. Workflow-System (Engine-) Fehler sind derzeit noch Gegenstand intensiver Forschung (siehe z.B. [KAGM96, AKA$^{+}$95]). *Aktivitätsfehler* umfassen Fehler innerhalb einer Aktivität. Die Behandlung dieser Fehlerklasse wird im Anschluß noch detaillierter diskutiert. *Kommunikationsfehler* sind Fehler, die die Verbindung zwischen Workflow-Scheduler einerseits und den Aktivitäten andererseits betreffen. Die Behandlung solcher Fehler ist vor allem für Workflow-Systeme relevant, die über das WWW agieren. Im Rahmen dieser Ausarbeitung werden in erster Linie Aktivitätsfehler und Kommunikationsfehler behandelt, wobei folgende Fehlertypen relevant sind [EL95]:

- *Systemfehler* umfassen primär informations-technologisch bedingte Fehler (wie z.B. Ausfall einer Systemkomponente, Probleme beim Aufbau einer Verbindung, etc.) sowie Programmfehler in den einzelnen Anwendungen (z.B. Division durch Null). Solche Fehler führen entweder direkt (bzw. indirekt, falls eine Aktivität überhaupt nicht gestartet werden kann) zu einem abnormalen Ende einer Aktivität.

- *Semantische (logische) Fehler* umfassen Ausnahmen innerhalb eines Geschäftsprozesses, die nicht auf Versagen der eingesetzten Informationstechnologie zurückgeführt werden können (z.B. die Aktivität „Hotel buchen“ schlägt fehl, weil das Hotel bereits ausgebucht ist). In einem solchen Fall terminiert die Aktivität zwar negativ aber nicht abnormal.

## 4.2 Workflow Recovery

Workflow Recovery [EL96] bedeutet, daß das Workflow-System mit bestimmten, im laufenden Betrieb möglichen Fehlersituationen „automatisch“, d.h. wenn möglich ohne Hilfe von außen, fertig wird, so daß die Abarbeitung der Prozesse nicht nachhaltig beeinflußt oder sogar unterbrochen wird. Folgende Recoverykonzepte können dabei zur Behandlung von Fehlern und Ausnahmen in Workflow-Systemen zur Verfügung gestellt werden:

1. *Recoverykonzepte nach einem Systemfehler:* Ein Systemfehler bewirkt ein abnormales Ende einer oder mehrerer zum Zeitpunkt des Systemfehlers gerade aktiven Aktivitäten. Die Recoverykomponente des Workflow-Systems muß nun dafür sorgen, daß einerseits die möglicherweise inkonsistenten Effekte der unvollständig ausgeführten Aktivitäten beseitigt werden (crash recovery) und daß andererseits eine Prozeßweiterbearbeitung (forward execution) ermöglicht wird. Aktivitäten werden in Workflow-System vernünftigerweise als „black boxes“ behandelt und man möchte damit meinen, daß die Aktivitäten im Fehlerfalle selbst dafür verantwortlich sind, mögliche Inkonsistenzen zu bereinigen (sie sollten fehler-atomar sein). Dies ist jedoch eine zu idealistische Annahme und es müssen daher bereits zur Modellierungszeit entsprechende Investitionen getätigt werden, um eine fehlertolerante Abarbeitung zu erreichen. In [EL95] wird ein Modell zur Modellierung fehlertoleranter Workflows vorgestellt. Auf Basis eines solchen Modells können sodann zur Laufzeit Aktivitätsfehler - je nach Aktivitätstyp - folgendermaßen abgefangen werden:

    - Bei automatischen Aktivitäten:
        - Automatischer Neustart der abgebrochenen Aktivität: Dies ist möglich, wenn die Aktivität selbständig ein automatisches Rollback durchführt oder wenn ein solches Rollback nicht erforderlich ist und die Aktivität auf Grund ihrer Struktur neu gestartet werden kann.

  - Automatischer Start einer anderen Aktivität: Es kann u.U. notwendig sein, eine Ersatzaktivität zu starten, die z.B. inkonsistente Effekte beseitigt und die gewünschte Aktivität nochmals ausführt oder aufgrund von Verbindungsproblemen überhaupt auf einem anderen Knoten im Netz ausgeführt wird.
  - Manuelle Intervention: Ist eine Aktivität nicht fehler-atomar und exisitiert keine Ersatzaktivität, so ist u.U. ein manueller Eingriff erforderlich (das Workflow-System informiert den Prozeßverantwortlichen).
- Bei manuellen Aktivitäten: Der Benutzer sollte auf den möglichst aktuellsten (zuletzt „gesicherten") Zustand aufsetzen und darauf aufbauend seine Arbeit fortsetzen können. Wie dies erreicht wird, liegt in aller Regel außerhalb des Einflußbereichs des Workflow-Systems.

Um die Erfolgsaussichten im Falle eines Systemfehlers zu verbesseren, kann das Workflow- System eine abgebrochene Aktivität (bzw. deren Ersatzaktivität) mehrmals neu starten. Sollte dies jedoch zu keinem Erfolg führen, so wird in einem nächsten Schritt die Recovery-Prozedur für semantische Fehler aktiviert um so eine automatische Prozeßweiterbearbeitung zu ermöglichen.

2. *Recoverykonzepte nach einem semantischen Fehler:* Ein semantischer (logischer) Fehler tritt auf, wenn eine Aktivität negativ terminiert. Recoverymaßnahmen sind nicht nach jedem semantischen Fehler notwendig sondern nur wenn:

   - Eine *vitale* (essentielle) Aktivität negativ terminiert (z.B. die vitale Aktivität „Abbuchung" terminiert negativ, weil das Konto bereits überzogen ist). Ob eine Aktivität vital ist oder nicht, muß vom Workflow Designer im Rahmen der Prozeßmodellierung spezifiziert werden.
   - Eine autorisierte Instanz (Akteur) aus bestimmten Gründen einen aktiven Prozeß oder Teile davon rückgängig machen möchte (z.B. ein noch in Arbeit befindlicher Reiseantrag wird vom Abteilungsleiter wieder zurückgeholt) bzw. abbricht (user-abort).

   Recovery bedeutet nun, daß u.U. eine oder mehrere zuvor abgearbeitete Aktivitäten rückgängig gemacht (kompensiert) werden müssen bis wieder ein konsistenter Zustand erreicht ist, von dem aus auf einem alternativen Pfad die Prozeßweiterbearbeitung erfolgen kann. Diese Kombination aus „backward recovery" und „forward recovery" entspricht dem Konzept des „Backtrackings". Das in [EL95] vorgestellte Modell bietet speziell dafür entsprechende Modellierungskonstrukte („choices") an, welche genau das zuvor gewünschte Verhalten ermöglichen.

# 5 Panta Rhei

Im diesem Abschnitt wird das Workflow-system Panta Rhei[1] beschrieben, welches vollständig in das Web integriert ist - jede Benutzerinteraktion wird über einen WWW Browser durchgeführt.

## 5.1 Darstellung von Workflows

Die wichtigsten Konzepte bei der Definition von Workflows sind:

*Prozeß:* beschreibt die Struktur eines Geschäftsprozesses
*Task:* beschreibt eine elementare Aufgabe, die von einem Akteur durchgeführt wird.
*Benutzer:* führt einen Task aus.
*Rolle:* definiert eine Menge von Benutzern, die eine gemiensame Fähigkeit oder Eigenschaft haben (z.B. Buchhalter oder Französichkenntnisse).

Wie ein Prozeß abläuft, wird in der Scriptsprache WDL definiert. Die Ausführung einzelner Tasks ist durch die Angabe des Benutzers, gefolgt von Tasknamen und - in Klammern - den bearbeiteten Formularen, definiert. Die Zeile `secr make_refusal(appl)` bedeutet, daß jemand, der zur Rolle `secr` gehört, den Task `make_refusal` durchführt und das Dokument `appl` bearbeitet.Kontrollstrukturen wie `if`, `while` und `do until` finden dabei Anwendung. Ebenso sind Konstrukte für die parallele Ausführung von Teilen des Prozesses vorhanden.

In einem Header werden einige Eigenschaften des Prozesses definiert, wie Administrator, Kurzbeschreibung, maximale Laufzeit und Subject. Bei der Abarbeitung eines solchen Prozesses, startet das Workflow-System jeweils einen Task, d.h. informiert den zugeordneten Benutzer, daß eine Aufgabe anliegt. Nachdem der Benutzer den Task beendet hat, startet das System den oder die Nachfolgeaufgaben.

## 5.2 Interaktion zwischen Workflow-Systemen

Die Interaktion zwischen zwei Systemen basiert auf dem Austausch von Formularen [GE95]. Das syntaktische Element zur Spezifikation dieser Interaktion in WDL ist:

remote *taskname(informs*, outforms*)*

Tritt ein solcher Task bei der Ausführung eines Prozesses auf, werden die *outforms* an die Empfänger verschickt. Falls keine *informs* angegeben werden, wird der Prozeß fortgesetzt. Sind *informs* angegeben, wird der Prozeß auch fortgesetzt, aber beim ersten nachfolgenden Task, der dieses Formular benötigt, angehalten, bis es eintrifft. Die Empfänger

[1] Heraklit: „Alles fließt“

der Formulare sind in diesen selbst vermerkt. Diese Art der Interaktion ermöglicht eine sehr einfache Implementierung des Interfaces. Es sind nur Schnittstellen zum Versenden und Empfangen von Formularen notwendig. Beim Empfangen eines Formulars wird ein Prozeß gestartet oder fortgesetzt. Die Schnittstelle 4 eines Workflow-Systems ist somit dadurch beschrieben, welche HTML Formulare es empfangen kann, welche Prozesse daraufhin angestoßen werden und ob bzw. welche Formulare als Anwort zurückgeschickt werden.

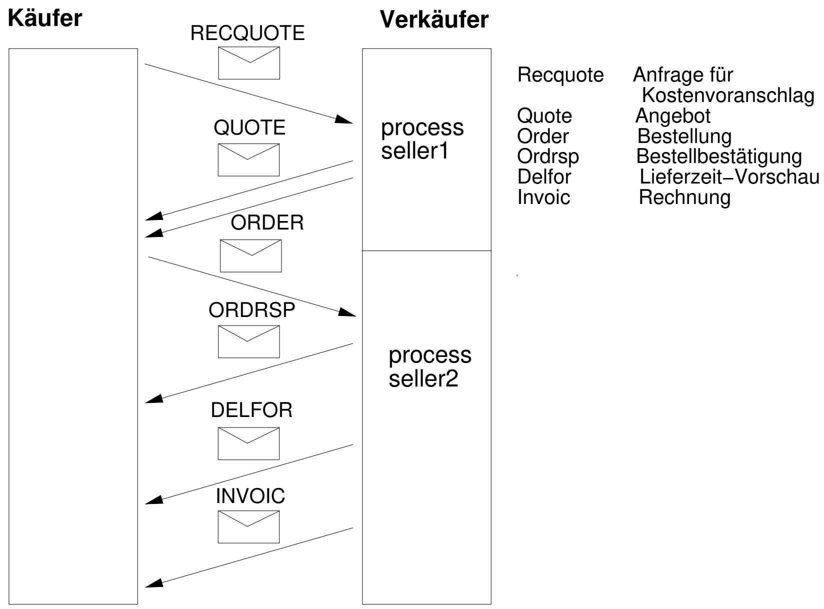


Abbildung 3: **Interaktionen in einem Verkaufsprozeß**

Abb. 3 zeigt die Interaktion in einem Verkaufsprozeß. Im Workflow-System des Verkäufers sind die zwei Prozesse `seller1` und `seller2` definiert. Der erste Prozeß behandelt die Schritte der Angebotslegung und Versendung, der zweite Prozeß definiert den Ablauf vom Eingang der Bestellung bis zum Erstellen der Rechnung.

In einem möglichen Szenario beginnt der Kunde die Interaktion entweder mit der Anfrage um einen Kostenvoranschlag (Start von Prozeß `seller1`) oder gleich mit einer Bestellung (`seller2`). Die Prozesse werden jeweils beim Empfangen des Formulars gestartet. Das Verschicken der Formulare kann von anderen WWW-Applikationen, wie zum Beispiel einem Online-Katalog, aus erfolgen.

## 5.3 Systemarchitektur

Abb. 5 zeigt den Aufbau des Panta Rhei Workflow-Systems mit seinen drei Hauptkomponenten:

```
process seller1(r) -- receives request for quote
r RECQUOTE;
q QUOTES;
o ORDERS;
begin
  seller make_quote(r,q);
  remote send_quote(q,o);
end;

process seller2(o) -- receives order
o ORDERS;
d DELFOR;
i INVOIC;
begin
  clerk   start_del(o,d);
  remote  send_del_mess(d);
  system  delivery(d)  -- waits until delivery is done;
  clerk   make_invoice(o,i);
  remote  send_invoice(i);
end;
```

Abbildung 4: **Definition eines Verkaufsprozesses in WDL**

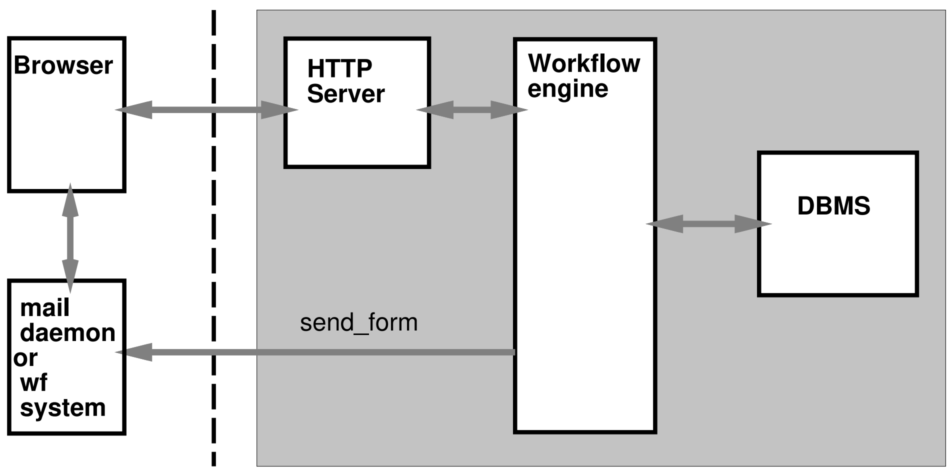


Abbildung 5: **Architektur des Panta Rhei Systems**

Der *HTTP-Server* ist die Schnittstelle des Workflow-Systems zum Netz. Anfragen von einem Client werden in Prozeduraufrufe übersetzt und an des Workflow-Managementsystem weitergeleitet.

Die *Workflow-engine* selbst ist eine Sammlung von Prozeduren, die die eigentliche

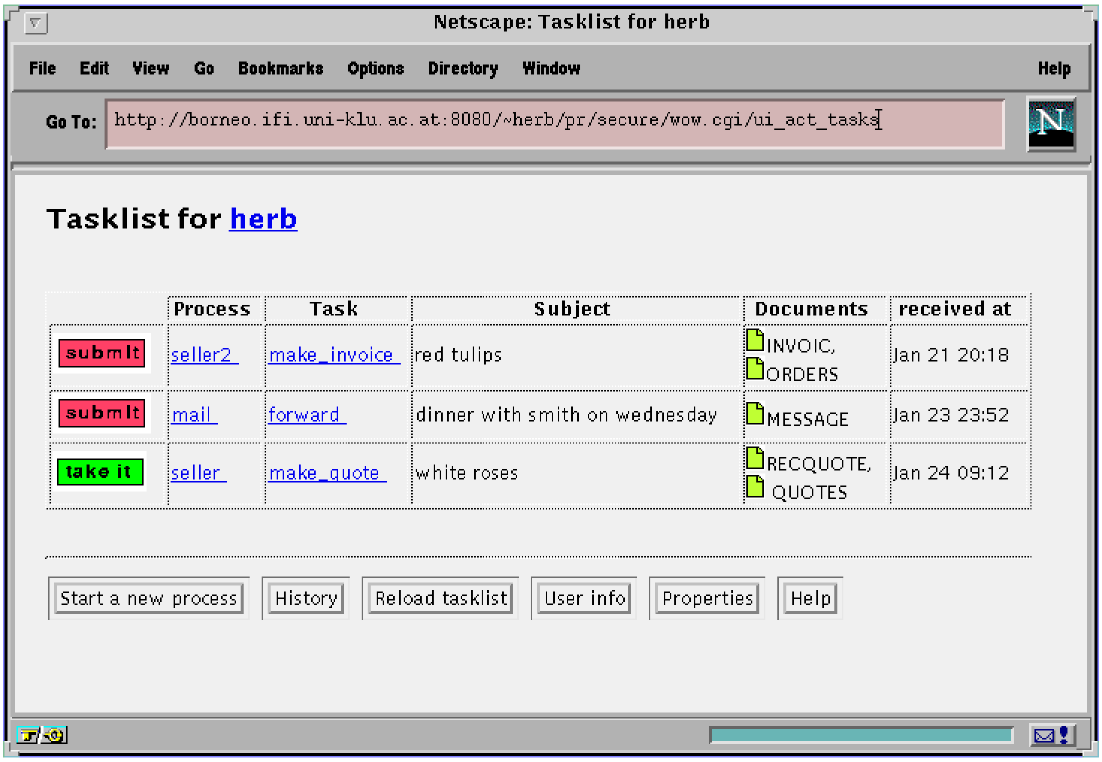


Abbildung 6: **Taskliste im Netscape browser**

Funktionalität des Workflow-Systems, wie Verändern von Formularen, Beenden eines Tasks, u.ä. bereitstellt.

Das *Datenbankmanagementsystem* speichert alle für die Prozeßausführung relevanten Informationen. Dies sind die Prozeßdefinitionen, die Stati der laufenden Prozesse, sowie alle Beutzerdaten, die in Formularen abgelegt werden.

Als Benutzerclient kann ein beliebiger Web Browser, wie Netscape [Net95] oder Microsoft Explorer Verwendung finden. Alle oben beschriebenen Interaktionsmöglichkeiten können damit durchgeführt werden. Abb. 6 zeigt einen Bildschirm der Schnittstelle eines „normalen“ registrierten Benutzers. Dieser erhält nach Anmeldung an das System eine sogenannte Tasklist (Arbeitskorb), die jeden von diesem Benutzer durchzuführenden Task in einer Zeile darstellt. Ist der Task dem Benutzer bereits fix zugeordnet, enthält die Zeile ein Submit-Feld zum Abschicken des Tasks nach Fertigstellung der Bearbeitung. Wenn der Task einer Rolle zugeordnet wurde, muß einer der in der Rolle enthaltenen Benutzer den Task übernehmen (mit „take it“). Weiters sind bei jedem Task Hyperlinks zu Prozeßbeschreibung, Taskbeschreibung und zu den zu bearbeitenden Dokumenten vorhanden.

Das Verschicken von Informationen an externe Personen erfolgt unterschiedlich, je nachdem ob die Person ein Workflow-System, das Formulare empfangen kann, installiert hat oder nicht. Wenn die URL eines Workflow-Systems als Empfänger angegeben wird, werden die Daten (Formulare) an diese URL geschickt. Andernfalls werden die Daten per email versendet. In diesem Fall werden nicht nur die rohen Daten sondern auch der HTML Code zum Aufbau des Formulars im Browser verschickt. Bei Verwendung eines Mail-Programms, das HTML Dokumente interpretieren kann, kann der Kunde somit in die email verpackte Formulare sofort ausfüllen und mit Mausklick verschicken. Abb. 7 zeigt ein Bestellformular, das auf eine Anfrage hin verschickt wurde (Schnittstelle zwischen den Prozessen *seller1* und *seller2*). Einige Positionen sind bereits vorausgefüllt, andere - wie zum Beispiel die Anzahl der gewünschten Produkte - sind noch vom Benutzer auszufüllen. Eine ebenfalls in das Formular integrierte JavaScript Routine berechnet den Gesamtpreis dynamisch. Ist das Formular ausgefüllt, erfolgt das Abschicken durch simplen Mausklick auf den *submit* Button. Die Empfängeradresse ist im HTML Dokument versteckt.

In dieser Weise kann die komplette Korrespondenz zwischen den Partnern in einem Geschäftsprozeß über das Workflow-System laufen, was den Vorteil hat, daß sämtliche Dokumente automatisch archiviert werden und über das Netz verfügbar sind. Das beschriebene System wurde mit dem Datenbanksystem ORACLE, dem WOW Orcale-Web Gateway [ORA95] und dem NCSA HTTP-Server [NCS95] implementiert.

# Literatur


[AKA+95] G. Alonso, M. Kamath, D. Agrawal, A. El Abbadi, R. Günthör, and C. Mohan. Failure Handling in Large Scale Workflow Management Systems. Technical report, IBM Almaden Research Center, 1995.

[BLC95] T. Berners-Lee and D. Connolly. Hypertext Markup Language 2.0. http://www.w3.org/hypertext/WWW/MarkUp/html-spec/html-spec_toc.html, June 1995. Internet Draft.

[BLFF95] T. Berners-Lee, R. Fielding, and H. Frystyk. Hypertext Transfer Protocol - HTTP/1.0. http://www.w3.org/WWW/Protocols, August 1995. Internet Draft.

[Coa94] Workflow Management Coalition. Glossary - A Workflow Management Coalition Specification. Brussels, Belgium, 1994.

[EGN94] J. Eder, H. Groiss, and H. Nekvasil. A Workflow System Based on Active Databases. In G. Chroust and A. Benczur, editors, *CON 94: Workflow Management: Challenges, Paradigms and Products*, pages 249–265. Oldenbourg, Linz, Austria, 1994.

[EL95] J. Eder and W. Liebhart. The Workflow Activity Model WAMO. In *Proc. of the 3rd Int. Conference on Cooperative Information Systems*, Vienna, Austria, May 1995.

[EL96] J. Eder and W. Liebhart. Workflow Recovery. To appear in: 1st IFCIS Int. Conference on Cooperative Information Systems, IEEE Computer Society Press, June 1996.

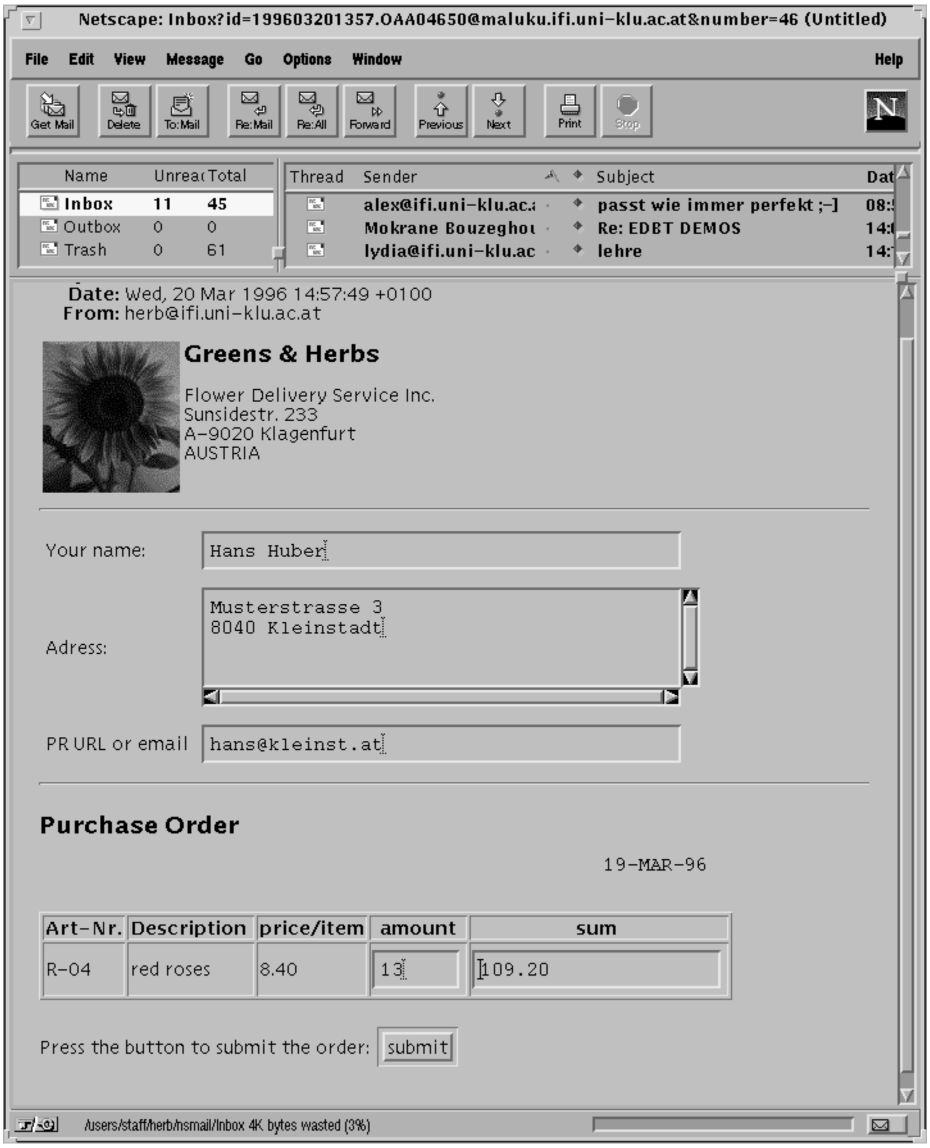


Abbildung 7: **Bestellformular als Teil einer e-mail**

[Elm92] A.K. Elmagarmid. *Database Transaction Models for Advanced Applications*. Morgan Kaufmann Publishers, 1992.

[GE95] Herbert Groiss and Johann Eder. Interoperability with World Wide Workflows. In *1st World Conference on Integrated Design & Process Technology*, Austin, TX, December 1995.

[GHS95] D. Georgakopoulos, M. Hornick, and A. Shet. An Overview of Workflow Management: From Process Modeling to Workflow Automation. In A. Elmagarmid, editor, *Distributed and Parallel Databases*, volume 3, pages 119–153. Kluwer Academic Publishers, Boston, 1995.

[GR93] J. Gray and A. Reuter. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, 1993.

[KAGM96] M. Kamath, G. Alonso, R. Guenthoer, and C. Mohan. Providing High Availability in Very Large Workflow Management Systems. In P. Apers, M. BouzegHoub, and G. Gardarin, editors, *5th Int. Conference on Extending Databse Technology (EDBT96)*, Avignon, France, March 1996. Springer.

[NCS95] NCSA. NCSA httpd. http://hoohoo.ncsa.uiuc.edu/, 1995.

[Net95] Netscape. Netscape Browser 2.0. http://home.netscape.com, 1995.

[ORA95] ORACLE. The Oracle World Wide Web Interface Kit. http://dozer.us.oracle.com:8080/, 1995.

[Rap95] UN/EDIFACT Rapporteurs. *United Nations Directories for Electronic Data Interchange for Administration, Commerce and Transport*. United Nations, 1995. http://www.unicc.org/unece/trade/untid/Welcome.html.

## Die Autoren

O.Univ.Prof. Dipl.-Ing. Dr. Johann Eder studierte Informatik an der Universität Linz wo er 1982 zum Diplom-Ingenieur für Informatik graduiert und 1985 zum Doktor der technischen Wissenschaften promoviert wurde. 1989 habilitierte er sich für das Fach Angewandte Informatik an der Universität Klagenfurt. 1990 wurde er zum Außerordentlichen Universitätsprofessor der Universität Wien ernannt. 1991 wurde er als Ordentlicher Universitätsprofessor für Informations- und Kommunikationssysteme an die Universität Klagenfurt berufen. Als Gastprofessor lehrte er außerdem an den Universitäten Hamburg und Wien (mehrmals). Die Schwerpunkte der wissenschaftlichen Tätigkeit Prof. Eders sind Weiterentwicklung der Datenbanktechnolgie (objektorientierte, aktive, deduktive Datenbanken), Datenbankentwurf und -tuning, Informations- und Kommunikationssysteme, insbesondere Workflow-Systeme. Im Bereich der Workflow-Systeme beschäftigt er sich vor allem mit der Modellierung von Workflows, der Realisierung von Workflow-Management-Systemen mit Hilfe von (aktiven) Datenbanken sowie mit Workflow-Transaktionen und offenen Workflows.

Dipl.-Ing. Dr. Herbert Groiss studierte Informatik an der TU-Wien, war dort von 1987 bis 1989 Vertragsassistent. Seither ist Universitätsassistent er am Institut für Informatik der Universität Klagenfurt. Er beschäftigt sich mit regelbasierten Sprachen, Workflow-Systemen und dem WWW, bei letzterem insbesondere mit der Integration mit Datenbank- und Workflow-Systemen.

Mag. Walter Liebhart studierte Angewandte Informatik an der Universität Klagenfurt. Seit 1994 ist er Vertrags- bzw. Universitätsassistent am Institut für Informatik der Universität Klagenfurt. Im Rahmen seiner Dissertation beschäftigt er sich mit erweiterten Transaktionskonzepten zur Fehler- und Ausnahmebehandlung in Workflow-Systemen.