



In: Zimmermann, Harald H.; Schramm, Volker (Hg.): Knowledge Management und Kommunikationssysteme, Workflow Management, Multimedia, Knowledge Transfer. Proceedings des 6. Internationalen Symposiums für Informationswissenschaft (ISI 1998), Prag, 3. – 7. November 1998. Konstanz: UVK Verlagsgesellschaft mbH, 1998. S. 32 – 46

# „KHSFlow“

## Ein System zur hypertextbasierten Workflowmodellierung und –steuerung

**Mohsen Shafaei**

DATEV eG  
Sigmundstraße 172, 90329 Nürnberg  
email: [Mohsen.Shafaei@datev.de](mailto:Mohsen.Shafaei@datev.de)

---

### Inhalt

1. Einleitung
2. Das Konstanzer Hypertext System (KHS)
3. Das hypertextbasierte Workflow-System KHSFlow
  - 3.1 Einige Grundlagen
  - 3.2 Warum KHSFlow?
  - 3.3 Das KHSFlow-Workflowmodell
  - 3.4 Das Workflow-Beispiel „Bildungsurlaub“
  - 3.5 Der KHSFlow-Designer
  - 3.6 Die Architektur
4. Ausblick

### Zusammenfassung

Workflow-Management-Systeme werden als Mittel zur Modellierung und Automatisierung betrieblicher Vorgänge (Workflows) eingesetzt. Dieser Beitrag zeigt, wie die Workflowmodellierung und -ausführung durch den Hypertext-Ansatz unterstützt werden kann. Im folgenden wird beispielhaft das hypertextbasierte Workflow-Management-System KHSFlow vorgestellt, welches auf das offene Hypertextsystem KHS (Konstanzer Hypertext System) aufsetzt. Es wird gezeigt, daß die Eigenschaften offener Hypertextsysteme dem Gegenstand des Workflow-Managements zugute kommen und sich damit einige der in diesem Bereich existierenden Probleme beseitigen lassen.



Dieses Dokument wird unter folgender [creative commons](http://creativecommons.org/licenses/by-nc-nd/2.0/de/) Lizenz veröffentlicht:  
<http://creativecommons.org/licenses/by-nc-nd/2.0/de/>

## 1. Einleitung

Workflows sind formal beschriebene Geschäftsprozesse, die von einem Workflow-Management-System zum Ablauf gebracht werden können. Ein Workflow-Management-System unterhält sowohl eine Modellierungskomponente, die der Spezifikation der Workflows dient, als auch eine Steuerungskomponente, die eine Ausführungsumgebung für Workflows definiert. Doch die meisten heute auf dem Markt vorhandenen Workflow-Management-Systemen beruhen auf herstellerspezifischen Modellierungs- und Steuerungskomponenten und zeigen immer noch Schwachstellen, vor allem bezüglich der Erweiterbarkeit und Wiederverwendbarkeit des Workflowmodells sowie Integration externer Informationsressourcen und -quellen. Es existiert keine einfache, einheitliche Workflowdefinitionssprache, die von jedem problemlos verstanden und eingesetzt werden kann. In der Workflowausführungsphase wird der Anwender oft zwischen verschiedenen Umgebungen bzw. Anwendungen geführt, die ihn bei der Workflowausführung unterstützen sollten. Dabei besteht aber die Gefahr, daß der Anwender schnell den Überblick verliert und sich überfordert fühlt.

Immer mehr Organisationen und Firmen erstreben eine unternehmensweit einheitliche Bürokommunikationsstrategie und versprechen sich durch den unternehmensweiten Einsatz von neuen attraktiven Technologien Zeit- und Kosteneinsparungen. Das Intranet ist ein Beispiel für solche Technologien. Als ein hypertextbasiertes unternehmensweites Netz integriert das Intranet die häufig heterogenen DV-Landschaften der Unternehmen unter der einheitlichen und benutzerfreundlichen Oberfläche eines WWW-Browsers. Der Einsatz von Workflow-Management-Systemen in verteilten heterogenen Umgebungen kann auch durch die Verwendung des WWW in Verbindung mit Komponenten, wie ActiveX und JavaBeans etc. unterstützt werden. Deswegen bemühen sich heute noch viele Hersteller der Workflow-Systemen um die Integration ihres Workflow-Systems mit dem WWW. Der Workflow-Pionier Action Technologies bietet beispielsweise mit Metro eine Webadaption an. Auch wenn nach Maßstäben der Hypertextforschung WWW es gar nicht verdient, als Hypertext-System bezeichnet zu werden, geht der Trend dahin Hypertext-Systeme mit dem WWW zu integrieren, um erstens die Inhalte der Hypertext-Systeme im WWW präsentieren zu können und zweitens intellektuelle, datenbankbasierte Hypertext-Systeme als WWW-Autoren-Systeme zu verwenden. Die Verbindung des offenen Hypertext-Systems KHS (Konstanzer Hypertext Systems) mit dem WWW ist ein Beispiel hierfür (vgl. Aßfalg 96). Hypertext-Systeme sind Informationssysteme, die seit Anfang der neunziger Jahre eine neue Forschungsrichtung innerhalb der Informatik etabliert haben. Sie finden ihre Anwendung vorwiegend in den Bereichen: Online-Dokumentation, Wörterbücher und Nachschlagewerke, Messen und Werbung, Journalismus, Fremdsprachen, Forschung, Wissensrepräsentation, Software Engineering, Workflow-Management (Prozeßanalyse), etc. Auch diese Informationssysteme können durch den Einsatz eines Workflow-Systems unterstützt werden. Nicht nur der Autor des Hypertextes könnte während seine Modellierungsarbeit (Aufbau des Hypertextes) unterstützt werden, sondern auch den Benutzern des Hypertextes kann dieser Einsatz zugute kommen. Daß die Verknüpfung der zwei Forschungsrichtungen Workflow-Management und Hypertext vorteilhaft ist, ist offensichtlich. Wie sollen aber diese

miteinander verknüpft werden? Kauft man sich ein Workflow-System, welches dann in eigenes Hypertextsystem integriert werden soll, oder integriert man besser ein Hypertextsystem in eigenes Workflow-System. Eine andere Vorgehensweise präsentiert der vorliegende Beitrag. Es wird für ein offenes datenbankbasiertes Hypertextsystem (das KHS), welches mit WWW verbunden ist und daneben auch auf andere Mächtigkeiten aufweist, eine Workflowkomponente konzipiert und implementiert. Es wird gezeigt, daß die Eigenschaften eines offenen Hypertextsystems bzw. die in so einem System bereits implementierten Funktionalitäten auch dem Workflow-Ansatz zugute kommen können und sich einige der in diesem Bereich existierenden Probleme beseitigen lassen.

In den folgenden Abschnitten werden zuerst die grundlegenden Eigenschaften des zugrundeliegenden offenen Hypertextsystems KHS (Konstanzer Hypertext System) beschrieben. Danach wird das KHSFlow (hypertextbasierte Workflowkomponente des KHS) vorgestellt.

## 2. Das Konstanzer Hypertext System (KHS)

Ein Kriterium, das zur Kategorisierung der Hypertextsysteme führt, ist die Offenheit bzw. Geschlossenheit dieser Systeme gegenüber externen Informationsquellen und -ressourcen. Offene Hypertextsysteme ermöglichen die Einbindung externer Informationsquellen in ihre Hypertextbasis und bieten Funktionalitäten zur Integration externer Informationsressourcen an, während geschlossene Hypertextsysteme diese Funktionalitäten nicht vorsehen. KHS ist ein *offenes datenbankbasiertes Hypertextsystem*, an dem mehrere Benutzer gleichzeitig arbeiten können. KHS wurde am Lehrstuhl für Informationswissenschaft der Universität Konstanz unter der Leitung von Herrn Prof. Dr. Rainer Kuhlen im Rahmen des DFG-Projekts WITH entwickelt. Die Implementierung des KHS erfolgte auf Unix-Workstations unter Einsatz der Smalltalk/VisualWorks Entwicklungsumgebung und des objektorientierten Datenbanksystems GemStone. Dank der Plattformunabhängigkeit der Smalltalk/VisualWorks ist KHS aber auch unter Macintosh, Windows95 und WindowsNT 4.0 lauffähig. KHS basiert auf die zuerst im [Kuhlen95, S. 432] publizierten Eigenschaften für offene Hypertextsysteme und beruht auf der Typisierung von allen Hypertextobjekten. Hypertextobjekte sind Knoten und Links sowie Hypertexte selbst als Sammlung von Knoten und Links [Ham93b]. Knoten stellen inhaltstragende Elemente des Hypertextes dar, welche durch deren Typisierung verschiedenartige Informationen wie Fließtext, Abbildung und Text, Formulare, Tabellen, Audio Informationen, Graphen und zahlreiche anders formatierte Inhalte haben können. Jeder Knotentyp kann nur Informationen von bestimmter, vordefinierter Form und Struktur aufnehmen. Knoten können aber auch Anweisungen enthalten, die sofort ausgeführt werden, wenn der Knoten selektiert wird. Strukturknoten (non-terminale Einheiten) präsentieren im KHS weitere Knotentypen, die weitere terminale bzw. non-terminale Einheiten enthalten können und außerdem Gegenstand von Verknüpfungen sein können. KHS stellt eine Reihe von Interaktions- und Präsentationswerkzeugen zur Verfügung. Als wichtigstes Werkzeug ist der KHS-Standardbrowser zu erwähnen, der dem Benutzer ständig zur Verfügung steht und ihn bei der Definition von Knoten und

Verknüpfungen, bei der Präsentation und Modifikation der Knoteninhalte, bei der Navigation in den Verknüpfungsnetzen oder Strukturhierarchien, bei der Aktivierung weiterer Werkzeuge zum Retrieval sowie bei der Aktivierung umfassender (graphische) Übersichten etc., unterstützt. Das mit KHS abgedeckte Aufgabenspektrum geht weit über Standardaufgaben, wie man sie bei konventionellen Büroarbeitsplätzen findet, hinaus. Terminplanung, Adreßverwaltung, Literaturverwaltung, Korrespondenz (Email) mit Archivierung, Online-Recherche und Verwaltung der Ergebnisse, Erstellung von WWW-Hypertexten, Softwareerstellung, Diskussionforen und elektronische Zeitschriften sind während einer KHS-Sitzung keine Fremdwörter. Die Integration des KHS in den Internet-Mehrwertdienst WWW -über die direkte Anbindung der KHS-Datenbank und die dynamische Generierung der HTML-Dokumente zur Anforderungszeit- ergänzt die Offenheit des Systems (nach außen) und verleiht ihm einen extra Glanz. (Ham96) stellt unter anderem eine umfassende wissenschaftliche Ausarbeitung des KHS zur Verfügung.

### 3. Das hypertextbasierte Workflow-System KHSFlow

#### 3.1 Einige Grundlagen

Jedes Workflow-Management-System muß die Mächtigkeit zur Erstellung, Modifikation und Interpretation von Workflowmodellen bieten. Dazu ist neben der Definition und Realisierung einer Sprache zur Formulierung dieser Modelle auch erforderlich, daß die Wartbarkeit und das Umfeld der Workflow-Management-Systeme berücksichtigt wird. Damit können die Modelle auch ausgetauscht und extern präsentiert werden. Viele Probleme bei der Um-setzung eines Anwendungssystems mit einem Workflow-Management-System sind auf ein unzulängliches Workflowmodell zurückzuführen. Jablonski stellt im (VoB96) eine Reihe fundamentaler Anforderungen an ein Workflowmodell vor, welche neben dem modularen Aufbau eines Workflow-Systems die wichtigsten Merkmale solcher Systeme darstellen. Er unterscheidet anwendungsbezogene Anforderungen (*Erweiterbarkeit, dynamische Anpaßbarkeit, Wiederverwendbarkeit und Offenheit*) von den modellbezogenen Anforderungen (*Graphische, formale und ausführbare Repräsentationsform*). Er sieht die Modularität als wichtigste Eigenschaft eines Workflowmodells an und führt unter Berücksichtigung der Unterscheidung zwischen sachlichen (funktionaler, operationaler, verhaltensbezogener, Informationsbezogener, organisatorischer und kausaler Aspekt) und technischen Aspekten (historischer und transaktionaler Aspekt) ein allgemeines aspektorientiertes Workflowmodell ein.

Auch die Architektur eines Workflow-Management-Systems muß einen modularen Aufbau aufweisen. Die WfMC (Workflow Management Coalition) stellt ein Standard-Referenzmodell vor, welches aus einem kontrollteil und fünf Schnittstellen (Prozeßdefinitions-komponente, Prozeßadministrations- und Überwachungskomponente, Applikations-komponente, Benutzerschnittstelle und Schnittstelle für Kommunikation mit anderen Workflowsystemen) besteht und die Trennung der Komponenten für die Analyse und formale Beschreibung von Prozeßmodellen und dem Workflow-Kontrollteil (Engine) erzwingt.

### 3.2 Warum KHSFlow?

Betrachtet man das WfMC-Referenzmodell, die anwendungsbezogenen und modellbezogenen Anforderungen an ein Workflowmodell sowie die im (Kuhlen95) vorgestellten Eigenschaften der offenen Hypertextsysteme (OHS), so stellt man fest, daß die offene Hypertextsysteme, und somit auch das KHS, Eigenschaften aufweisen, die den Anforderungen an Workflowmodelle und Workflow-Management-Systeme gerecht werden. Diese Gemeinsamkeiten haben dazu beigetragen, daß man sich dafür entschieden hat, für das KHS eine eigene Workflowkomponente zu implementieren und kein auf dem Markt existierendes Workflow-System einzusetzen. Die folgende Tabelle faßt die wichtigsten Gemeinsamkeiten zusammen:

|   | Anforderungen an Workflowmodelle und -systeme                                  | Eigenschaften der OHS  |
|---|--|--|
| 1 | Erweiterbarkeit des Workflowmodells (neue Modellelemente)                      | OHS sind offen bezüglich des Umfangs (neue Knotentypen)  |
| 2 | Wiederverwendbarkeit der Workflowspezifikationen                               | Typisierung der Knoten und Verknüpfungen   |
| 3 | Offen bezüglich der Integration vorhandener und neuer Anwendungssysteme        | Offen für heterogene Informationsressourcen  |
| 4 | Bereitstellung der externen Daten  | Einbindung externer Informationsressourcen   |
| 5 | Schutz der Workflowtypen / Workflowinstanzen vor unbefügtem Zugriff            | Datenbankorientierte / Filesystemorientierte Schutzmechanismen   |
| 6 | Graphische Präsentation der Workflows  | Diverse Präsentationwerkzeuge  |
| 7 | Durchführung anderer Tätigkeiten neben dem Workflowgeschäft (andere Werkzeuge) | Komfortable Oberfläche, die dem Nutzer erlaubt, vom Hypertextsystem aus z.B. auch andere Mehrwertdienste des internationalen Informationsmarktes zu nutzen (E-Mail, Internet, Online-Datenbanken, Gopher, ...) |

Das Projekt KHSFlow hat die prototypische Realisierung und Evaluierung eines hypertextbasierten Workflow-Systems am Beispiel personalbezogener Daten der DATEV eG zum Ziel und wird am Lehrstuhl für Informationswissenschaft der Universität Konstanz unter der Leitung von Herrn Prof. Dr. Rainer Kuhlen entwickelt. Die Implementierungsarbeiten erfolgen auf WindowsNT unter dem Einsatz der Entwicklungsumgebung VisualWorks3.0 (VisualWave30) und des relationalen Datenbanksystems Oracle. KHSFlow baut auf die bereits im KHS implementierten Funktionalitäten (Klassen und Methoden) und stellt zusätzliche Klassen und Funktionen zur Verfügung, die den Anwender in allen Phasen des Workflow-Managements (Informationserhebung, Analyse, Optimierung, Modellierung und Ausführung) unterstützen sollen.

### 3.3 Das KHSFlow-Workflowmodell

Ein Workflow kann anhand von Aspekten modelliert werden, die zusammen die verschiedenen Facetten eines Workflows beschreiben (BuJa94). Jablonski und Bußler stellen in (JaBu96) ein solches aspektorientiertes, modularaufgebautes Workflowmodell (MOBILE) vor, das im Rahmen der Workflowforschung am Lehrstuhl für Datenbanken der Universität Erlangen-Nürnberg entstand. Das Workflowmodell des KHSFlow ist ähnlich wie das MOBILE-Workflowmodell aufgebaut und berücksichtigt zur Zeit folgende Aspekte:

Der *funktionale Aspekt* definiert den Schritt bzw. die Schritte eines Workflows („Was“). Der *verhaltensbezogene Aspekt* beschreibt das „Wann“ der Ausführung. Hier wird die Abarbeitungsreihenfolge der Workflows unter Verwendung diverser Kontrollflußkonstrukten festgelegt. Der *organisatorische Aspekt* gibt Auskunft über die ausführenden Personen („Wer“, das kann auch ein Programm sein). Der *informationsbezogene Aspekt* spezifiziert die Daten, die bei der Ausführung der Workflows benötigt werden („Womit“, Datenfluß). Der *operationale Aspekt* beschreibt Ressourcen und Module, welche bei der Workflowausführung benötigt bzw. gestartet werden („Wie“). Um die natürlich-sprachliche Beschreibung der Workflows zu ermöglichen, sind bestimmte Mo-dellelemente vorgesehen. Außerdem können weitere Beschreibungselemente jederzeit in das Modell integriert werden. Die Modellierung nach Aspekten erlaubt die Betrachtung von Workflows von einem bestimmten Gesichtspunkt aus und ermöglicht die Zerteilung des Modellierungsraumes in kleine, überschaubare Teile.

Im KHSFlow sind Workflows Instanzen der Klasse Workflowtyp. Eine Workflowinstanz ist entweder ein Elementar-Workflow oder ein Komposit-Workflow, welcher aus mindestens einem oder unter Umständen  $n$  Elementar-Workflows (Subworkflows) bestehen kann. Außerdem beinhaltet eine Workflowinstanz weitere Elemente, welche die oben erwähnten aspekt-bezogenen Informationen enthalten. Workflows werden im KHSFlow durch typisierte Strukturknoten dargestellt, die wiederum weitere Strukturknoten oder (terminale) Knoten enthalten können. Die Struktur dieser Knoten ist ausgehend von dem aktuellen Workflowmodell vorgegeben und kann jederzeit erweitert bzw. angepaßt werden. Um das Workflowmodell zu veranschaulichen, wird im folgenden zuerst ein Beispiel-Workflow modelliert.

### 3.4 Das Workflow-Beispiel „Bildungsurlaub“

Ein Bildungsurlaub (Kurz: Bur) muß vom Mitarbeiter schriftlich beantragt werden (BurBeantragen). Nach der Genehmigung des Antrages (BurGenehmigen) durch den jeweiligen Vorgesetzte wird der Antrag entweder abgelehnt (BurAblehnen) oder an die Personalabteilung zur Prüfung (BurPrüfen) weiter-geleitet. Dort wird zuerst geprüft, ob das Unternehmensgesamtkontingent für Bildungsurlaube bereits ausgeschöpft ist (GesKontPrüfen) oder ob der Antragsteller sein persönliches Bildungsurlaubskontingent (maximal sechs Tage innerhalb von zwei Jahren) erreicht hat (PersKontPrüfen). Abhängig von dieser Prüfung bekommt der Antragsteller automatisch entweder eine Absage (BurAbsage) oder eine Zusage (BurZusage). Im Falle einer Zusage werden die Antragsdaten an das Gleitzeitsystem weitergeleitet (S. Abb. 1)

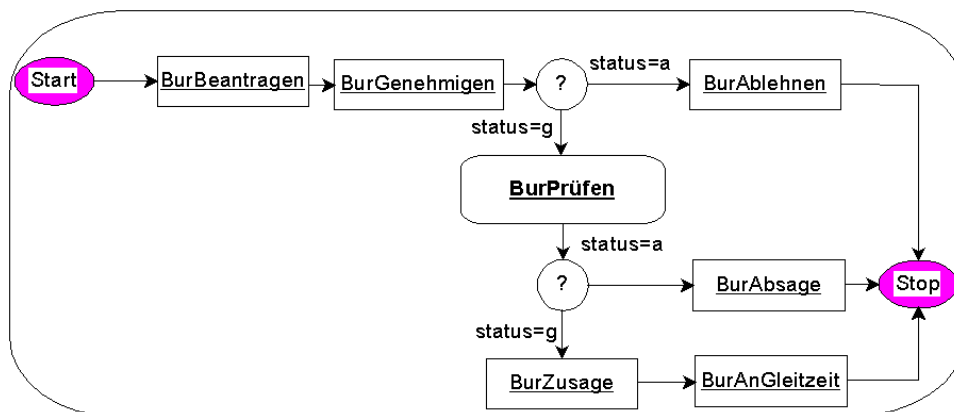


Abb. 1 : Workflowtyp „Bildungsurlaub“

Der funktionale Aspekt ist in der Abbildung durch Rechtecke (elementare Workflows) bzw. Rechtecke mit gerundeten Ecken (kompositere Workflows) beschrieben. Der Workflowtyp „Bildungsurlaub“ ist also ein kompositere Workflow, der aus sechs elementaren und einem kompositere Workflow (BurPrüfen) besteht. Der Subworkflow „BurPrüfen“ ist wiederum ein kompositere Workflow, der aus zwei elementaren Workflows „GesKontPrüfen“ und „PersKontPrüfen“ besteht (graphisch nicht dargestellt). Der verhaltensbezogener Aspekt ist durch Kontrollkonstrukte und Pfeile zwischen den einzelnen Workflows dargestellt. Die von Workflows benötigten Daten, die ausführende Personen der Workflows sowie die Programme (Funktionen bzw. Module), die mit den elementaren Workflows verbunden sind, sind hier nicht dargestellt, müssen jedoch während der Modellierungsarbeit spezifiziert werden. Um das Beispiel-Workflow mit KHSFlow zu modellieren, geht man wie folgt vor:

Man erstellt bzw. öffnet ein Hypertextdokument, welches mehrere typisierte Hypertextobjekte enthalten kann (auch Workflowtypen sind Hypertextobjekte) und

fügt in das Hypertextdokument einen Knoten vom Typ Composit-Workflow ein. Das System fragt nach dem Knotennamen (in unserm Fall Bildungsurlaub) und fügt den Knoten ein, welcher automatisch weitere (vorher definierte) aspektspezifische Sub-Knoten (Description, Variablen, Constraints, ControlFlow, DataFlow und Rolle) beinhaltet. In den Knoten **Description** kann man die verbale Beschreibung des Workflows eingeben. Der kompositere Knoten **Variablen** kann weitere Knotentypen (Input,Output,Inout) enthalten und spezifiziert Workflow-Variablen (z.B. Input initiator: String oder Input antrag: Antrag). Hier können die bekannten Datentypen (String, Integer, Float,...) sowie vorher definierte Datentypen bzw. Datenstrukturen verwendet werden. Der kompositere Knoten **Constraints** kann ebenso weitere spezielle Knoten ( EnterConstraint, RunConstraint sowie ExitConstraint) enthalten, die beispielsweise angeben, wann ein Workflow nicht ausgeführt werden darf (Beispiel: RunConstraint (not\_after\_20pm)). In den Knotentyp **Rolle** kann man beispielsweise die wichtigsten Daten (Name, Vorname, Abteilung, Email-Adresse Zuständigkeit,...) einer Person abspeichern, die den Workflow ausführt. Im Knoten **ControlFlow** wird festgelegt, in welcher Reihenfolge (bei welchen Zuständen) die Workflows ausgeführt werden müssen. Schließlich

kann der Datenfluß zwischen den Workflows in den kompositen Knoten **DataFlow** spezifiziert werden.

Als nächster Schritt muß der Knoten Bildungsurlaub selektiert werden (man befindet sich in diesem Knoten), um weitere Workflow-Knoten aus unserem Beispiel einfügen zu können (beliebige Erfassungsreihenfolge). Ein elementarer Workflow-Knoten beinhaltet ebenso weitere Knotentypen (Description, Variablen, Constraints, Application und Rolle). Der Knotentyp **Applikation** beinhaltet die Operation (Funktion, Methode, Modul, Programm), die aktiviert wird, wenn ein elementarer Workflow ausgeführt wird (also das „Wie“). Somit ist die Struktur des kompositen Workflows „Bildungsurlaub“ gestaltet. Die Abbildung 2 zeigt den KHS-Standardbrowser nach der Selektion des Knotens Bildungsurlaub.

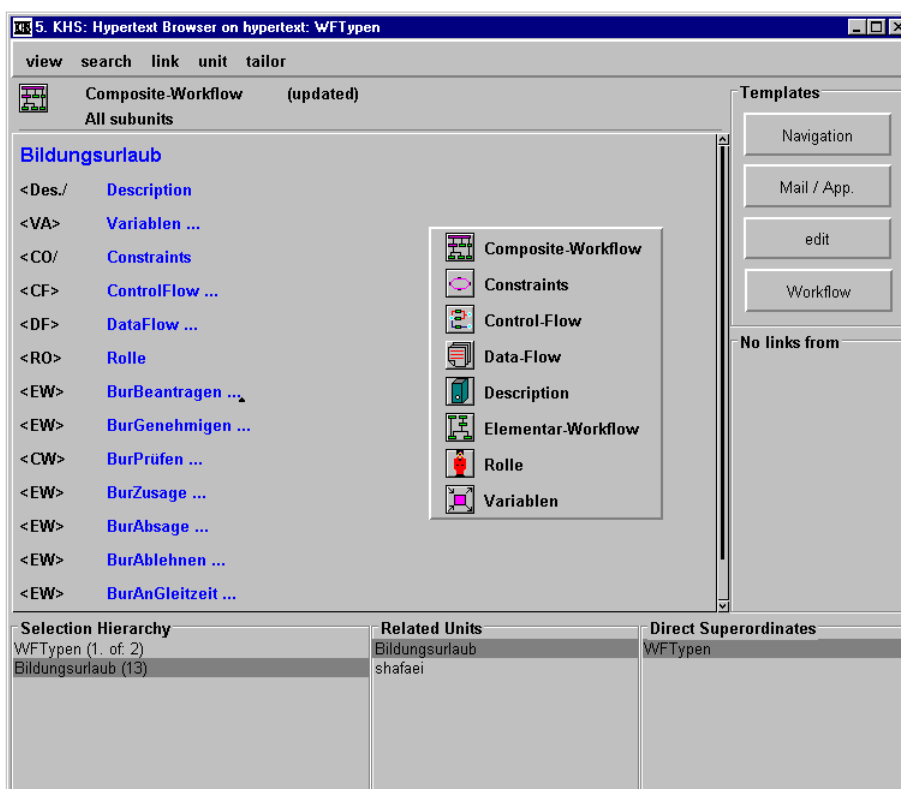


Abb. 2: Der KHS-Standardbrowser. Selektiert ist der komposite Workflow-Knoten „Bildungsurlaub“. Das mittlere Fenster zeigt alle Knotentypen, die in den aktuell selektierten Knoten einfügbar sind.

### 3.5 Der KHSFlow-Designer

KHSFlow stellt zusätzlich zum KHS-Standardbrowser einen speziellen Workflowmodellierungsbrowser (den KHSFlow-Designer) zur Verfügung, der dem Benutzer die Möglichkeit bietet, Workflows via Drag&Drag zu spezifizieren. Die folgenden Abbildungen zeigen den Designer bei der Darstellung des kompositen Workflow-Beispiels „Bildungsurlaub“ und des elementaren Workflows „BurBeantragen“. Abhängig vom Typ des selektieren Knotens ändert sich der Inhalt des Toolbars (Typisierungskonzept). Die zwei Listen auf der linken Seite der Abbildungen zeigen die aktuell selektierte Hierarchieebene (oben links) und die Nachbarknoten des aktuellen Knotens (unten links) an.



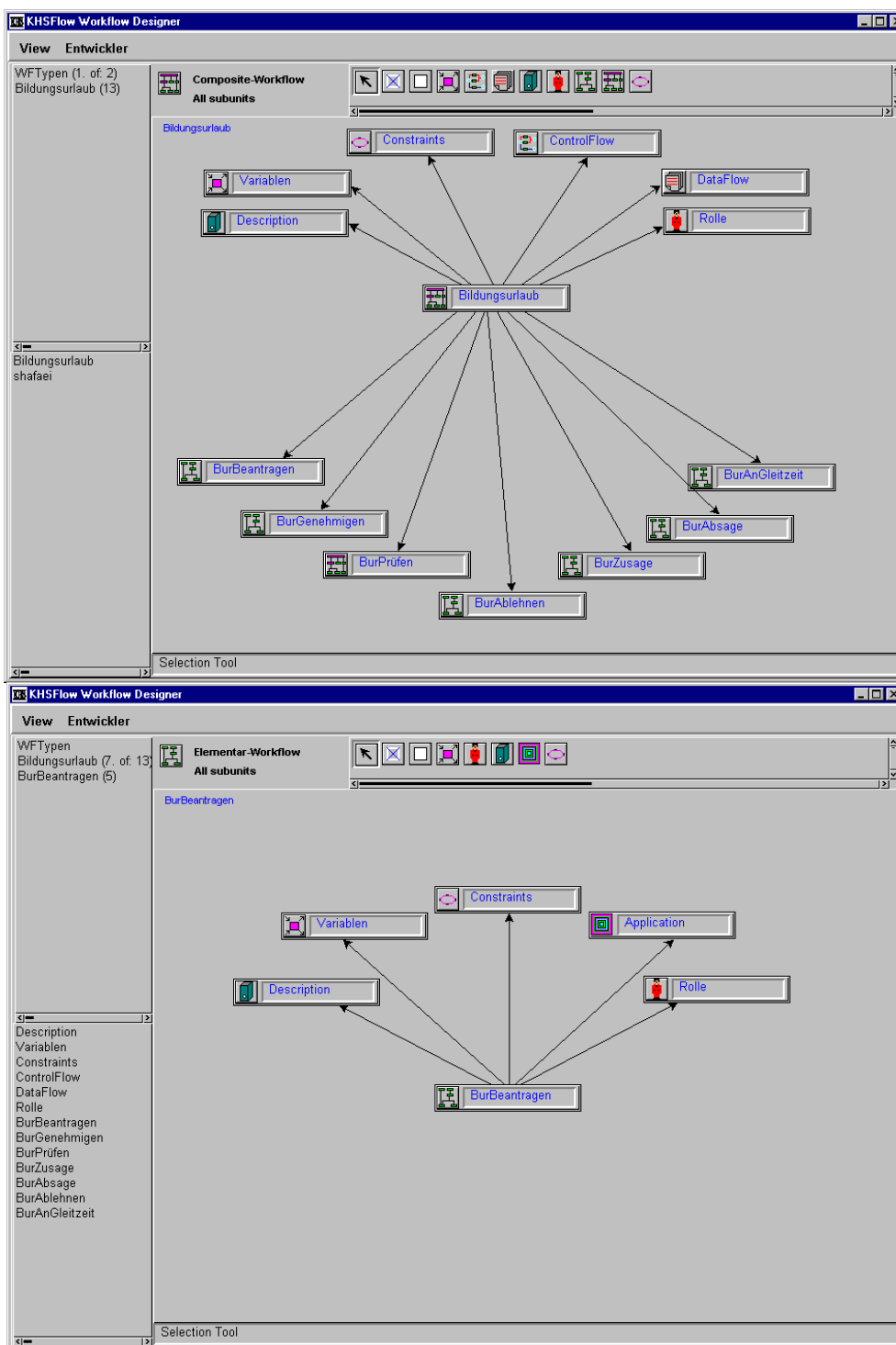


Abb. 3: Der KHSFlow-Designer. Dargestellt wird hier der funktionale Aspekt

Der Mausklick auf ein Listenelement verursacht die sofortige Navigation zu diesem Knoten und update des Toolbars. Ist der selektierte Knoten ein Strukturknoten, der Sub-Knoten enthält, so entsteht automatisch ein solches Netz, wie es im mittleren Fenster der Abbildungen (Abb. 3) zu sehen ist (der Doppelklick auf ein Rechteck (CompositeUnitFigur/FormUnitFigur) bewirkt das gleiche), sonst wird der Inhalt des elementaren Knotens dargestellt .

Rechtecke präsentieren Knoten (Ikone betonen den Knotentyp) und Pfeile bringen die referentiellen Knoten-Verknüpfungen zur Darstellung. Der Knotentyp ControlFlow enthält die strukturellen Verknüpfungen, die die Abarbeitungsreihenfolge von Knoten (Workflows) bestimmen. Wie in der Abbildung 4 ersichtlich (s. Toolbar), werden hier bestimmte Kontrollfluß-konstrukte (Sequence, Parallel, Bedingte Ausführung, Limitierung, Verzögerung, And, Or, Start, Stop und Auswahl ) sowie spezielle Mechanismen zur Verfügung gestellt, die dem Modellierer erlauben, Kontrollflüsse in gewohnter Form, also via Drag&Drop und per Mausklick, spezifizieren zu können. Die Abbildung zeigt die Kontrollflüsse des Workflow-Beispiels „Bildungsurlaubs“. Um die Parallität zur Darstellung zu bringen, wurde das Beispiel leicht modifiziert (Parallele Ausführung der Workflows „BurZusage“ und „BurAnGleitzeit“).

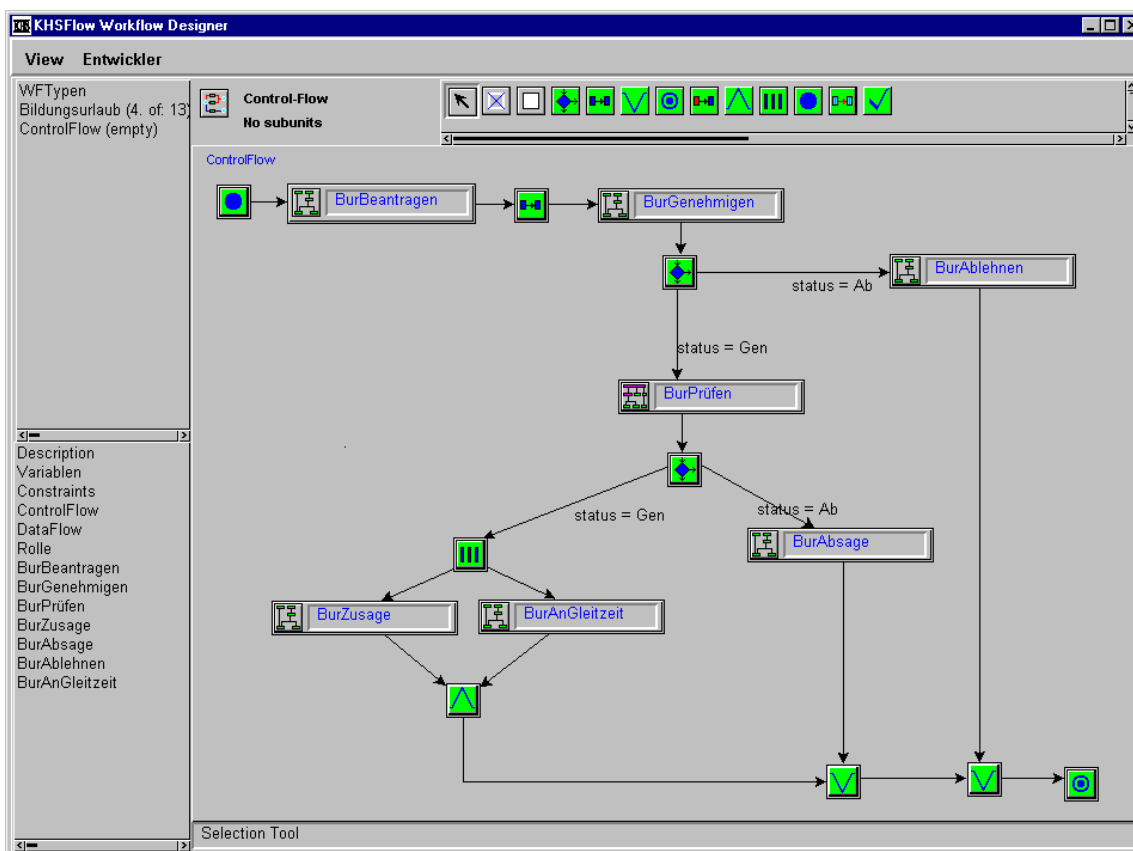


Abb. 4: Der KHSFlow-Designer. Selektiert ist der Knotentyp „ControlFlow“, der die aktuelle Abarbeitungsreihenfolge des kompositen Workflows „Bildungsurlaub“ graphisch darstellt.

Das grob vorgestellte KHSFlow-Workflowmodell kann mit geringem Aufwand auf weitere Elemente erweitert werden. Ein wesentlicher Vorteil des Modells besteht darin, daß der Modellierer keine besonderen Kenntnisse bezüglich einer Scriptsprache haben müßte, um Workflowtypen definieren zu können. Die definierten Workflowtypen können samt ihrer Struktur und Subworkflows innerhalb eines anderen Workflowtyps wiederverwendet werden. Zu beachten ist noch die Tatsache, daß der hier eingeführte Strukturknoten „ControlFlow“ (ControlConstructions) nichts anderes darstellt als spezielle Hypertext-

Verknüpfungen, die nicht nur genau zwei Hypertextobjekte miteinander verknüpfen, sondern auch die Verbindung von mehreren Ursprungs- und Zielobjekten miteinander erlauben.

Nachdem das KHSFlow-Workflowmodell sowie der KHSFlow-Designer grob eingeführt sind, soll in den folgenden Abschnitten die Architektur des KHSFlow kurz dargestellt werden. Danach werden die wichtigsten Komponenten des KHSFlow aufgelistet.

### 3.6 Die Architektur

Eine grobe Betrachtung des Systems läßt erkennen, daß das System aus zwei Teilen besteht: Server-Teil und Client-Teil. Der Server ist eine leistungsfähige Maschine, auf welcher die KHS-Anwendung und somit auch das KHSFlow ständig aktiv sind. VisualWorks 3.0 und VisualWave3.0 bilden die Basis für das KHS und KHSFlow. Der Server ist fähig, sowohl mit der objektorientierten Datenbanksystem GemStone als auch mit der relationalen Datenbanksystem Oracle zu kommunizieren. Auch Zugriffe auf Hostdatenbanken (wie z.B. DB2-Datenbanken) sind vom Server aus (über ODBC-Schnittstellen) möglich. Be-dingt durch den Einsatz vom VisualWave kann der Anwender sowohl über das Intranet (unternehmensintern) als auch über das Internet (unternehmensüber-greifend) auf die im KHS bzw. KHSFlow implementierten Anwendungen zu-greifen. Der Anwender muß hierfür natürlich ein Zugriffsrecht haben (d.h. dem Server bekannt sein). Wir werden in den folgenden Abschnitten sehen, wie das funktionieren kann. Abbildung 5 stellt die Umgebung des KHSFlow ganz grob dar. KHSFlow ist vorerst als ein in diese Umgebung integriertes Workflow-System zu betrachten. KHSFlow ist modular aufgebaut, basiert auf das WfMC-Referenzmodell und besteht aus sieben Komponenten, die im folgenden grob erläutert werden.

*KHSFlow-Designer.* Die Workflowtypen (Modelle) können via KHS-Standard-browser definiert werden und mit Hilfe des Table-Of-Content-Browser des KHS in grafischer Form betrachtet werden. Der KHSFlow-Designer verknüpft die Funktionalitäten der beiden Browser. Während der Table-Of-Content-Browser nur die graphische Präsentation der definierten Modelle ermöglicht, bietet der KHSFlow-Designer eine graphische Oberfläche zur Definition, Manipulation und Präsentation der Modelle.

*KHSFlow-Admin.* Diese Komponente unterstützt den Administrator des Systems bei Administration und Überwachung der Workflows und arbeitet eng mit dem Kontrollteil, der u. a. auch über die Stati der Workflows Bescheid weiß, zusammen.

*KHSFlow-OrgManager.* Die Historie und somit auch die aktuelle Organisationsstruktur des Unternehmens (Aufbauorganisatoin) ist hier abgebildet. Der KHSFlow-OrgManager liefert auf Anfragen des Kontrollteils bzw. des KHSFlow-Designers Informationen über den Bearbeiter des Workflows.

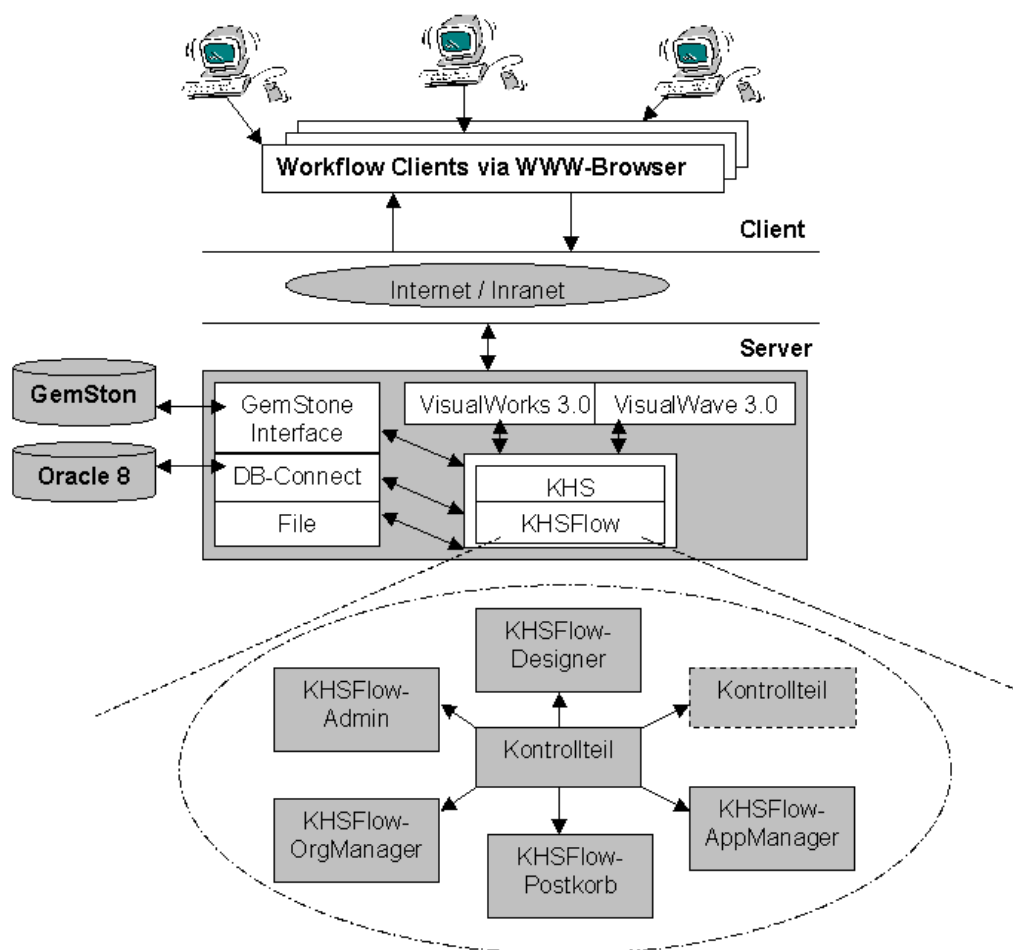


Abb. 5: Die Architektur und wichtigste Komponenten des KHSFlow

**KHSFlow-AppManager.** Spezielle Applikationsaufrufe, Java-Applets, ActiveX-Komponenten und Smalltalk-Controller, die Methoden zum Aufruf bestimmter Applikationen enthalten oder selbst Applikationen öffnen und verwalten, sind dem ApplikationManager bekannt und werden beim Bedarf vom ihm gestartet bzw. referenziert.

**KHSFlow-Postkorb.** Der KHSFlow-Postkorb ist die Schnittstelle des Endanwenders (Client) zum KHSFlow. Der Postkorb ist ein speziell für das KHSFlow entwickelter Dialog, der bedingt durch den Einsatz von VisualWave auch webfähig ist. Der Anwender kann den KHSFlow-Postkorb vom WWW-Browser aus aufrufen (Der Postkorb kann auch bei dem Client direkt installiert werden). Nach der Identifikation und Authentifikation des Benutzers wird dann seine Arbeits-liste aufgebaut. Der Anwender kann Workflows, die auf seiner Liste existieren, bearbeiten, weiterleiten, löschen, etc. Aus dem aktuellen Workflowtyp-Katalog, der ebenso hier zur Verfügung steht, kann der Anwender Workflows instantiieren (s. Abb. 6).

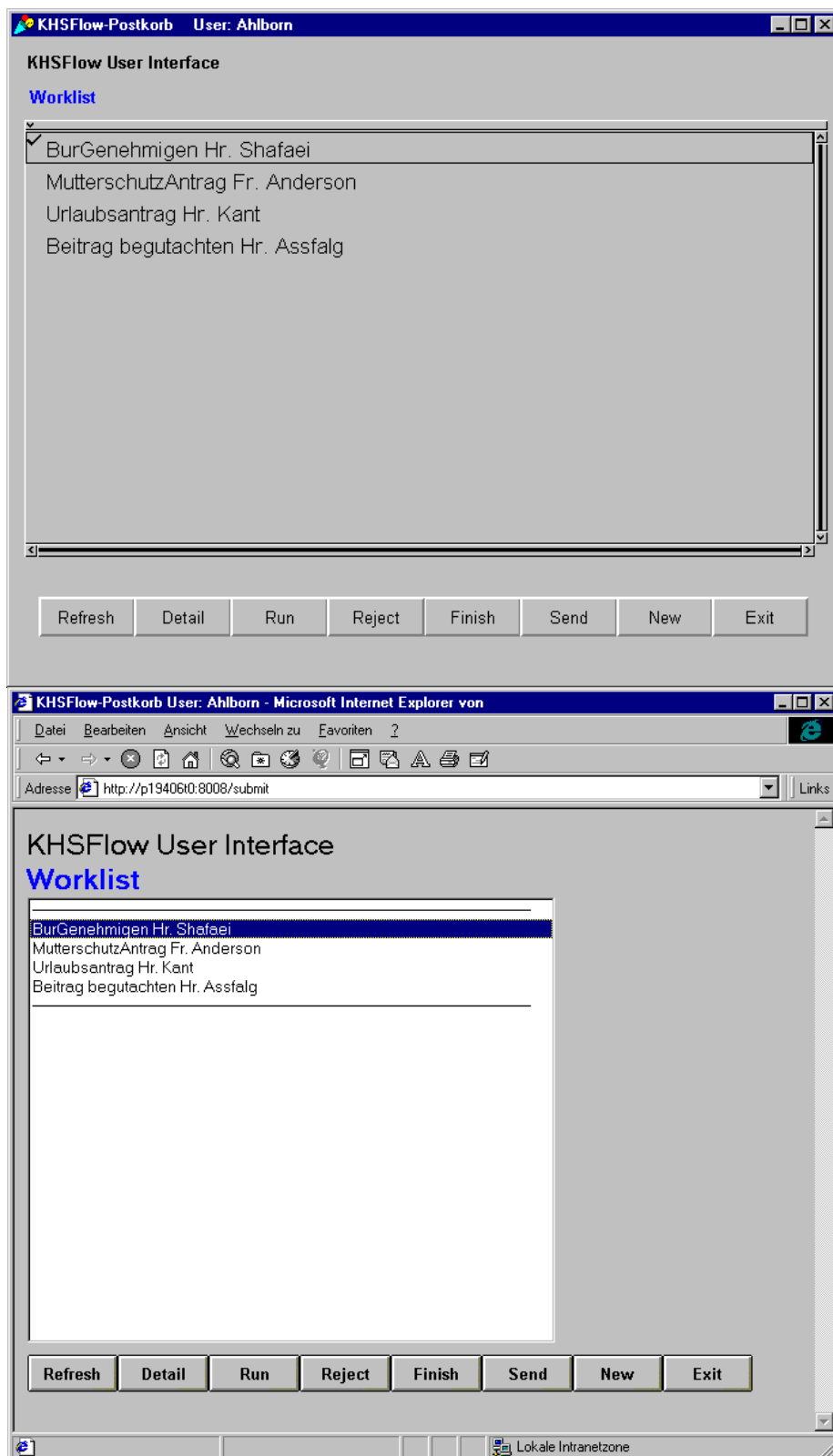


Abb. 6: Der KHSFlow-Postkorb

*Der Kontrollteil.* Der Kontrollteil ist das Herz des KHSFlow. Eine der wichtigsten Aufgaben des Kontrollteils ist die Instantiierung des abstrakt definierten Workflowtypen. Auf Grundlage des aktuellen Zustands eines Vorgangs sowie des vorgegebenen abstrakten Modells entscheidet der Kontrollteil, welcher Workflow

als nächster zu bearbeiten wäre. Der Kontrollteil besteht aus mehreren Modulen (z.B. ObjectMaker, WorkTable, StatusManager,...). Der Kontrollteil kommuniziert ständig mit anderen Komponenten, die für ihn bestimmte Aufgaben erledigen.

#### 4. Ausblick

Wir arbeiten momentan an der Implementierung des oben vorgestellten Ansatzes. Zur Zeit können Workflows mit Hilfe des KHSFlow-Designers problemlos modelliert werden. Die aktuell verfügbaren Kontrollflußkonstrukte ermöglichen einfache Modellierung komplexer Abhängigkeiten. Neue Kontrollflußkonstrukte können problemlos eingeführt werden. Obwohl einige Komponenten des Systems noch nicht (fertig) implementiert sind, können zur Zeit doch einfache Workflows ausgeführt werden. Der Client kann von WWW-Browser aus mit dem Server kommunizieren und Workflows bearbeiten oder instantiiieren (starten). Die Zusammenarbeit zwischen KHSFlow-Komponenten läuft befriedigend.

Parallel zur Weiterentwicklung des Systems, modellieren wir die Geschäftsprozesse (Workflows) der Personalabteilung der DATEV mit Hilfe des KHSFlow. Dadurch wird die Funktionalität des Systems anhand praxisrelevanter Beispiele getestet und bei Bedarf erweitert.

#### Literatur

- [Aßfal96] R. Aßfal: *Integration eines offenen Hypertext-Systems in den Internet-Mehrwertdienst World Wid Web*. Ein Ansatz unter Verwendung eines objektorientierten Datenbanksystems, Hartung-Gorre Verlag, Konstanz 1996.
- [BuJa94] Bußler C., Jablonski S.: An Approach to Integrate Workflow Modeling and Organization Modeling in an Enterprise. In: Proc. 3<sup>rd</sup> IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprise (1994), S. 81-95.
- [Conklin87] J. Conklin: *Hypertext: An introduction and survey*, IEEE Computer 20 (1987) 9. S. 215-220.
- [HaRi93] Hammwöhner R., Rittberger M. : *KHS - Ein offenes Hypertext-System*, Bericht 28-93 (WITH-3/93) Mai 1993, Informationswissenschaft, Universität Konstanz.
- [Ham93b] Hammwöhner R. , Kuhlen R. : *Semantik Control of Open Hypertext Systems by Typed Objects*, Bericht 34-93 (WITH-6/93) August 1993, Informationswissenschaft, Universität Konstanz.
- [Ham96] Hammwöhner R. : *KHS - Ein offenes Hypertext System*, Habilitation 1996, Informationswissenschaft, Universität Konstanz.

- [Jab95] Jablonski S. : *Workflow-Management-Systeme, Modellierung und Architektur*. Thomson's aktuelle Tutorien, TAT 9, International Thomson Publishing, Bonn 1995.
- [JaBu96] Jablonski S., Bußler C. : *Workflow Management Modeling Concepts, Architecture and Implementation* International Thomson Computer Press, London, Bonn,... 1996.
- [Kuhlen91] Kuhlen R.: *Hypertext - Ein nicht-lineares Medium zwischen Buch und Wissensbank*, Edition SEL-Stiftung, Springer-Verlag, Berlin 1991.
- [Kuhlen95] Kuhlen R.: *Informationsmark, Chancen und Risiken der Kommerzialisierung von Wissen*, Schriften zur Informationswissenschaft Band 15, Univ.- Verlag, Konstanz 1995.
- [Schulz97] Schulze W.: *Workflow-Management Entwicklung von Anwendungen und Systeme*, dpunkt.verlag, Heidelberg 1997
- [Shafaei96] Shafaei M.: *Workflow & Hypertext*, Diplomarbeit 1996, Informationswissenschaft, Universität Konstanz.
- [ÖstV96] Österle H., Vogler P.: *Praxis des Workflow-Managements, Grundlagen, Vorgehen, Beispiele*, Vieweg, Wiesbaden 1996.
- [VoB96] Vossen G., Becker J.: *Geschäftsprozeßmodellierung und Workflow-Management: Modelle, Methoden, Werkzeuge*, 1. Auflage, Thomson Publishing GmbH, Bonn 1996.