

Hinz, Holger; Kimms, Alf

Working Paper — Digitized Version

Btx PC-Verbund im Hochschulunterricht: Ein entscheidungsorientiertes cash Management-System

Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 291

Provided in Cooperation with:

Christian-Albrechts-University of Kiel, Institute of Business Administration

Suggested Citation: Hinz, Holger; Kimms, Alf (1992) : Btx PC-Verbund im Hochschulunterricht: Ein entscheidungsorientiertes cash Management-System, Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 291, Universität Kiel, Institut für Betriebswirtschaftslehre, Kiel

This Version is available at:

<https://hdl.handle.net/10419/161416>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

Nr. 291

Btx/PC-Verbund im Hochschulunterricht:
Ein entscheidungsorientiertes
Cash Management-System

Holger Hinz*
Alf Kimms

April 1992

* Dr. Holger Hinz, wiss. Assistent am
Lehrstuhl für Finanzwirtschaft
Direktor: Prof. Dr. Reinhart Schmidt
Institut für Betriebswirtschaftslehre
cand. rer. nat. Alf Kimms

1. Einleitung¹

Der Einsatz von Entscheidungsunterstützungssystemen ist ein besonderes Anliegen im betriebswirtschaftlichen Hochschulunterricht. Die Schwierigkeit besteht dabei darin, eine Integration von betriebswirtschaftlichen Methoden und Inhalten mit Informatik- und Medienkenntnissen herbeizuführen, um neben einem hohen theoretischen Ausbildungsniveau auch eine für die Praxis akzeptable Ausbildung gewährleisten zu können.

Cash Management-Systeme (CMS) sollen der Unterstützung von Entscheidungen im Finanzmanagement dienen. Untersucht man die derzeit von externen Anbietern (Banken, Berater, Softwarehäuser usw.) auf dem Markt angebotenen Systeme im Hinblick auf ihre Fähigkeit, *Decision Support* zu liefern, so wird rasch deutlich, daß diese aufgrund mangelnder Einbindung solcher Systeme in die unternehmensinterne EDV schnell an ihre Grenzen stoßen.² Vielfach werden noch reine Informationssysteme oder solche mit der zusätzlichen Möglichkeit, Transaktionen durchzuführen, angeboten. Allein die neuen *Treasury Workstations* (sog. Systeme der dritten Generation) lassen erstmals eine Hinwendung zu mehr Entscheidungsunterstützung erkennen, indem sie etwa What-If-Analysen oder bestimmte Simulationen zulassen.³

In dieser Arbeit wird daher die prototypische Implementation eines einfachen Cash Management-Systems dokumentiert, die auf einer Kopplung unternehmensinterner Daten mit externen Kontoinformationen basiert. Die Vorhaltung lokaler

¹ Teile dieser Arbeit entstanden im Rahmen einer Fortgeschrittenenübung zum Thema "Elektronische Finanzinformations- und -planungssysteme".

² Vgl. z.B. die Übersichten bei Eistert, T.: *Electronic Banking und Cash Management Systeme*, Studie zur Wirtschaftsinformatik, Nr. 1, Stuttgart 1991, S. 61; Müller, M.: *Vergleichende Analyse existierender Cash Management-Systeme aus Anwendersicht*, Diplomarbeit, Lehrstuhl für Finanzwirtschaft, Kiel 1990, S. 113ff.

³ Vgl. o.V.: *Supplement "Treasury Workstations" to Corporate Finance*, Oktober 1990, S. 25-26.

Informationen wird hier auf einem Personalcomputer realisiert, welcher zur Beschaffung der externen Kontoinformationen bzw. zur Realisation bestimmter Transaktionen auf eine Anbindung an das Bildschirmtextsystem (Btx) der TELEKOM zurückgreift.⁴

Im folgenden werden wir zunächst exemplarisch ein Problem der Verwaltung mehrerer Bankkonten definieren. Sodann spezifizieren wir die Lösung dieses Problems, indem wir den softwaremäßigen Aufbau des Prototyps näher beschreiben und dessen Wirkungsweise anhand des Beispielproblems aufzeigen. Abschließend erfolgt eine kritische Würdigung der Implementierung und ein Ausblick.

2. Das Entscheidungsproblem

Eine der Basisaufgaben im Treasury Management liegt in der Steuerung der Kontokorrentkonten. Im Hinblick auf das Rentabilitätsziel haben sich in der Praxis verschiedene Formen des Konten-Clearing durchgesetzt, die die Glattstellung einzelner Konten durch Überweisungstransaktionen beinhalten.

Eine praktikable Vorgehensweise wäre etwa, ein sogenanntes Clearing-Konto zu definieren, das den Gesamtkontensaldo aufnimmt, nachdem ein kostenminimaler Geldtransfer zwischen den beteiligten Konten durchgeführt wurde. Möglicherweise muß bei diesem Transfer berücksichtigt werden, daß bestimmte geschäftspolitische Regelungen oder gesetzliche Bestimmungen nicht verletzt werden. Solche geschäftspolitischen Restriktionen können z.B. Mindest- oder Höchstkontenbestände in Abhängigkeit von Bankenpräferenzen sein. Planungsrelevante gesetzliche Bestimmungen wären z.B. Kapitaltransferbeschränkungen, falls die Konten in verschiedenen Ländern angesiedelt sind. Diese Liste von Restriktionen ließe sich noch erweitern.

⁴ Vgl. auch Kuhlmann, E.: Vom Btx zum Bildinformationssystem, in: L.J. Heinrich; G. Pomberger; R. Schauer (Hrsg.): Die Informationswirtschaft im Unternehmen, Linz, 1991, S. 99-119.

Beispiel: Verwaltung von Bankkonten

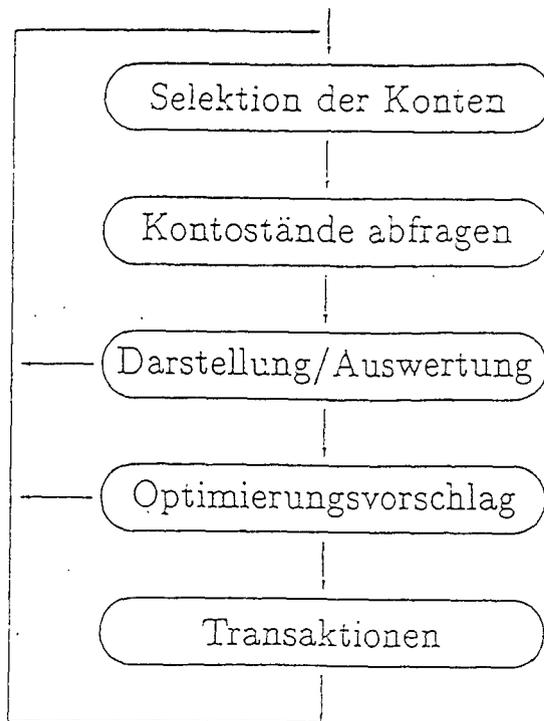


Abbildung 1: Steuerung von Kontokorrentkonten

Damit ergibt sich ein Problemaufbau, wie er sich in Abbildung 1 zusammengefaßt dargestellt findet. Das Treasury Management muß dabei zunächst eine Selektion der steuerungsrelevanten Konten vornehmen, deren Kontostände zyklisch abzufragen sind. Nach Darstellung bzw. Reportgenerierung erfolgt eine Auswertung der Datensituation im Hinblick auf die vorformulierte Problemstruktur derart, daß ein Optimierungsvorschlag zur Problemlösung generiert wird. Abschließend bedarf es dann noch der Realisation der ermittelten optimalen Transaktionen.

3. Aufbau und Ablauf des Programmsystems

3.1 Aufbau des Programmsystems

Zur Realisation eines Systems, welches die oben beschriebenen Informations-, Entscheidungs- und Transaktionsfunktionen übernehmen kann, wurde hier eine PC-Host-Kopplung vorgenommen, die es gestattet, die Vorteile des Bildschirm-

textsystems der Telekom mit denen einer lokalen Recheneinheit zu verbinden.

Die softwaremäßige Steuerung des Systems liegt dabei auf dem lokalen Rechner und beinhaltet zunächst die Verwaltung der Kontenstammdaten (vgl. Abbildung 2) inklusive der mit diesen Kontendaten verknüpften Btx-Zugriffsdateien, die für den maschinellen Zugriff auf die im Bildschirmtextsystem vorgehaltenen Kontobestände notwendig und vom Benutzer zu pflegen sind.

Weiterhin wird zur Beschickung des Btx-Systems ein Softwarerecoder eingesetzt, der in der Lage ist, Befehlsfolgen einer Pascal-artigen Abfragesprache auszuführen. Auf diese Weise kann die aufwendige Schnittstellenprogrammierung zum Btx-System - etwa in Form der API-Schnittstelle⁵ - vermieden werden. Die zur Steuerung des Btx-Dialogs notwendigen Befehlsdateien werden zur Laufzeit generiert und dem Decoder zur Verfügung gestellt.

Diese Befehlsdateien werden mit Hilfe der oben genannten Btx-Zugriffsdateien erzeugt. Eine solche Zugriffsdatei ist für jede Bank einzurichten und hat dabei aus zwei Abschnitten zu bestehen:⁶ Zunächst finden sich Angaben über die Position der Kontostandsanzeige auf der auszulesenden Btx-Seite, der Art der numerischen Darstellung des Kontostandes, z.B. bezüglich der Nachkommastellen- und der Tausendertrennung, der Bankleitzahl, sowie der sogenannten Transferdatei, die eine Überweisung von einem Konto dieser Bank beschreibt.

In einem zweiten Abschnitt folgt eine Auflistung derjenigen Tastenkombinationen, die zur Anwahl der Btx-Seite mit den Kontoinformationen nötig sind. Die konkrete Kontonummer eines Kontos bei der entsprechenden Bank wird durch ein Schlüsselwort (KONTONUMMER) repräsentiert, so daß für ver-

⁵ Vgl. IBM Deutschland GmbH: Btx-API, Definition of a Unique Application Program Interface for Btx-Decoders, Stuttgart 1989.

⁶ Vgl. Anhang A.

schiedene Konten bei einer Bank dieselbe Zugriffsdatei benutzt werden kann.

Konten		Kontostand
Bayerische Hypobank	2345234562 hupo t	
BfG Bank	8234275642 bfg t	
Commerzbank	1284612342 comm t	
Deutsche Bank	7234523637 deut t	
Deutsche Bundespost	3942852346 dbp t	
Dresdner Bank	5345872352 dres t	
LB Schl.-Holst.	4023453457 lbsh t	
Noris Bank	2345234562 noris t	
Verbraucherbank	3452334343 verbr t	
Vereins- u. Westbank	3525462536 vewe t	

←
Ändern
Ins/Del
Entf/Entf
A V
Konto
←
Menü
F1
Hilfe
F5
Farbe
F9
Rechner
F10
Btx
Esc
Ende

Abbildung 2: Kontenstammdatenverwaltung

Weiterhin greift das System auf einen leistungsfähigen Problemlösungsalgorithmus zu, der die Optimierung durchführt und die Lösung bereitstellt. Damit entfällt die Notwendigkeit der Programmierung der Methodenkomponente des Systems.

Die Transaktionskomponente des Systems wird wiederum durch Bereitstellung von Steuerdateien realisiert, die durch den Btx-Decoder interpretiert werden können und so die optimalen Transaktionen einleiten. Der Aufbau dieser Transferdateien ist analog zu den oben beschriebenen Btx-Zugriffsdateien.⁷

Damit ergibt sich insgesamt der in Abbildung 3 dargestellte Aufbau des Programmsystems.

⁷ Vgl. Anhang B.

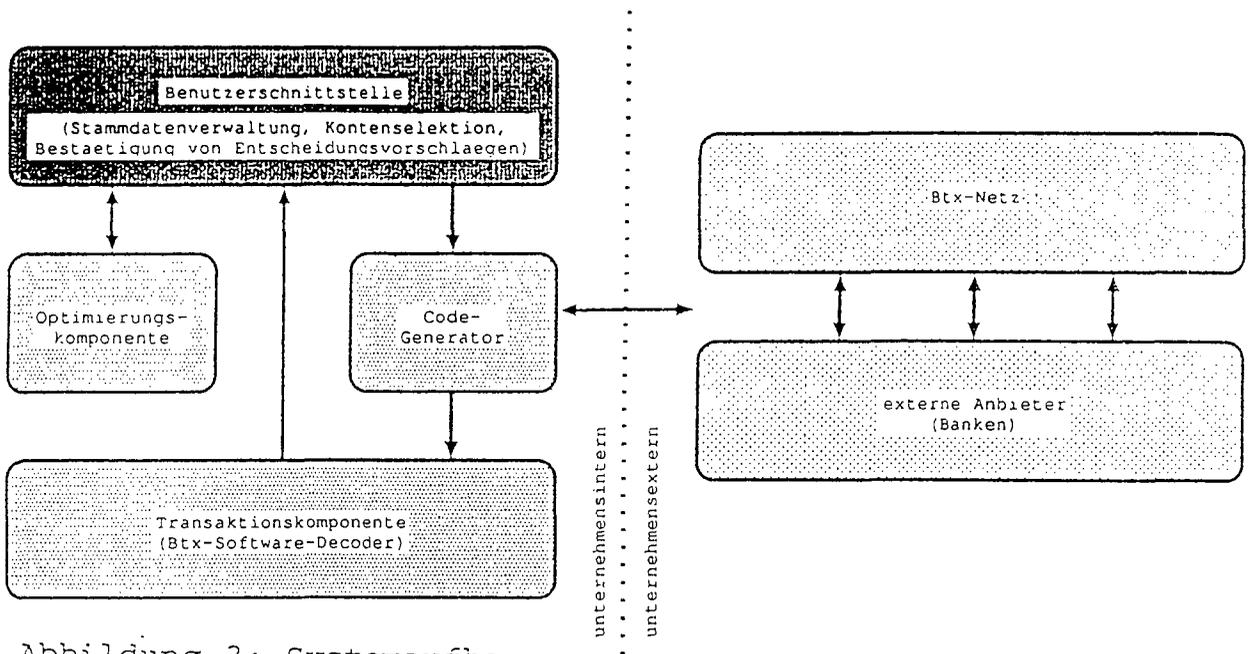


Abbildung 3: Systemaufbau

3.2 Ablauf des Programmsystems

Nach der Selektion der problemrelevanten Konten, bei der auf die Kontenstammdatenverwaltung zugegriffen wird, generiert das System unter Verwendung der Btx-Zugriffsdateien eine Befehlssequenz zur Anfrage an das Btx-System, die durch den Softwaredecoder interpretiert wird und zum Auslesen der Kontostände der selektierten Konten führt. Das Ergebnis dieser Anfrage zeigt Abbildung 4, in der die Salden der einzelnen Konten in einer Übersicht dargestellt und der Gesamtsaldo der Konten ausgewiesen wird.

Kontostände			
Bayerische Hypobank	2345234562	1.000,00 S	
BfG Bank	8234275642		1.200,00 H
Commerzbank	1284612342		12.000,00 H
Saldo			9.200,00 H

Donnerstag, 27.02.1992 12:42:21 Uhr

Abbildung 4: Ergebnis der Anfrage an das Btx-System

Sofern gewünscht, kann der Benutzer nun die Kontendaten für seine Zwecke lokal weiterverarbeiten. Die hier gewählte exemplarische Problemstruktur führt zu einem klassischen Transportproblem, das mit Hilfe eines Standardpakets zur Linearen Optimierung gelöst wird.⁸ Zu diesem Zweck wird eine Anfrage an das Optimierungssystem generiert, die auf eine vordefinierte Problemstruktur zurückgreift und dann eine Lösung anhand der aktuellen Kontobestände in einem bekannten Format bereitstellt.

Nach Auslesen der optimalen Lösung wird diese interpretiert und am Bildschirm wie in Abbildung 5 dargestellt. In diesem Beispiel werden bestimmte Transfers zwischen den involvierten Konten derart vorgeschlagen, daß die Kosten der Glattstellung minimal werden.

Optimierende Transaktionen		
Bayerische Hypobank	2345234562	
↓		
BfG Bank	8234275642	1.000,00 H
.....		
Commerzbank	1284612342	
↓		
Bayerische Hypobank	2345234562	5.000,00 H
.....		
Commerzbank	1284612342	
↓		
BfG Bank	8234275642	7.000,00 H

F10 - Transaktionen ausführen

Abbildung 5: Optimale Lösung des Entscheidungsproblems

Sofern der Nutzer den Systemvorschlag akzeptiert, kann er die optimalen Transaktionen (Taste F10) einleiten. Das System generiert dann die entsprechende durch den Softwarede-

⁸ Das Beispielproblem ist in den Anhängen C und D dokumentiert.

coder ausführbare Befehlsfolge und realisiert die Transaktionen.

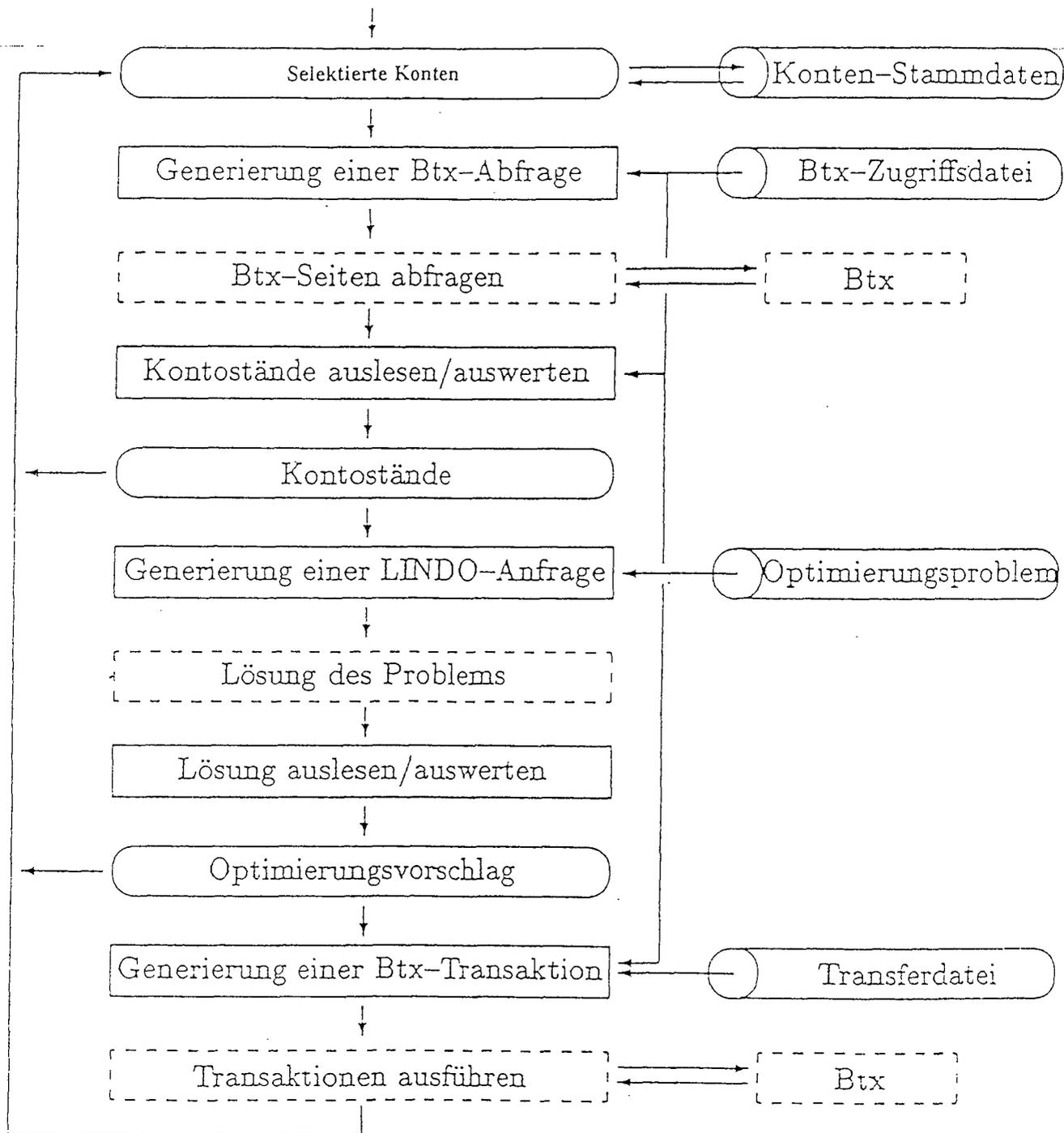


Abbildung 6: Ablauf des Programmsystems*

* Legende:

- [gestrichelte Kästen] Funktionen von Fremdsoftware
- [eckige Kästen] Funktionen dieses Systems
- [abgerundete Kästen] Inputparameter

Insgesamt hat der Systemnutzer damit ein integriertes Softwarepaket zur Verfügung, das die Funktionen "Informieren", "Entscheiden" und "Realisieren" umfaßt. Damit geht der Systemumfang weit über das hinaus, was markt-gängige Cash Management-Systeme üblicherweise anbieten.

Der schematisch dargestellte Ablauf in Abbildung 6 faßt die oben beschriebenen Komponenten übersichtlich zusammen. Es wird deutlich, daß die Informationsphase mit der Ermittlung der Kontostände, die Entscheidungsphase mit dem Optimierungsvorschlag und die Transaktionsphase mit der Ausführung der Transaktionsdatei abschließen. Nach Beendigung jeder dieser drei Phasen kann optional zum Beginn des Gesamtprozesses verzweigt werden.

Daneben findet sich eine Übersicht der verwendeten Dateien, die entweder im Btx-System (gestrichelte Linien) oder auf dem lokalen Rechner (durchgezogene Linien) zur Verfügung stehen.

4. Kritische Würdigung und Ausblick

Die Vorstellung des hier implementierten CMS-Prototyps hat deutlich gemacht, daß eine sinnvolle Einbindung von DSS-Komponenten in solche Systeme problemlos realisierbar ist. Im Mittelpunkt der Betrachtung stand die Zerlegung der Aktivitäten in folgende Phasen:

INFORMATIONSBESCHAFFUNG -> ENTSCHEIDUNG -> TRANSAKTION

Diese erscheint für praktische Zwecke einerseits sowie aus forschersicher Sicht andererseits durch die interessante Verbindung von Informations- und Kommunikationstechnik und betriebswirtschaftlicher Methodik besonders relevant. In didaktischer Hinsicht weist der Modellaufbau sicherlich große motivatorische Vorteile gegenüber konventioneller Unterrichtsgestaltung auf.

Dabei hat sich insbesondere die gewählte Verbindung von Bildschirmtext als ein effizientes Trägersystem zur Haltung von Kontoinformationen mit einem Personalcomputer als dezentraler Einheit für bestimmte Datenbank-, Verwaltungs- und Optimierungsaufgaben, als problemadäquate Konfiguration herausgestellt.

Mit Blick auf die Realitätsnähe der bearbeiteten Problemstellung sind allerdings einige einschränkende Bemerkungen am Platz: Was die Problemgröße anbetrifft, so bestehen Restriktionen bezüglich des für die Kontodatensammlung zur Verfügung stehenden Zeitrahmens sowie der Möglichkeiten zur Lösung des Optimierungsproblems. Hinsichtlich der Programmlaufzeitproblematik sind allerdings wesentliche Infrastrukturverbesserungen durch die Einführung von Btx-ISDN durch die TELEKOM bereits realisiert, die eine effiziente Verwaltung von mehreren hundert solcher Konten realistisch erscheinen läßt. Der Modellgehalt der präsentierten Formulierung entspricht wohl nur *im Prinzip* tatsächlichen Problemstellungen des Finanzmanagements und dient insofern lediglich der exemplarischen Untermauerung der hier verfolgten Zielkonzeption eines vollintegrierten CMS.

Erweiterungen des Systems sind in mehrere Richtungen denkbar. So lassen sich zunächst problemlos Transfer- und Anfragedateien zur Datenkommunikation mit ausländischen Systemen des Btx-Verbunds kreieren, die das System dann zu einem internationalen Cash Management befähigen.¹⁰ Weiterhin ist vor allem an die Beschickung von Schnittstellen zu sonstigen internen Systemen wie der Finanzbuchhaltung zu denken, in der dann etwa die aus der Optimierung generierten Transaktionen zu Buchungen führen, oder zur Portfeuilleverwaltung, sofern Transaktionen z.B. den Handel mit Wertpapieren beinhalten. Außerdem läßt sich die Gesamtkonzeption auf ähnlich gelagerte Problemstellungen - wie z.B. das Konzernclearing - übertragen. Zudem ist an den Einsatz von Verfahren der Künstlichen Intelligenz (KI) zu

¹⁰ Vgl. Hinz, H.: Some Advances in Supporting International Financial Management Decisions, in: Journal of Financial Management and Analysis, Januar/Juni, 1990, S. 50-57.

denken, wie sie etwa für die strategische Planung vorgeschlagen werden.¹¹

Zudem bietet sich die Systemkonzeption für die Entwicklung von Lehr-/Lernsystemen an, die auf eine effiziente informationstechnologische Infrastruktur Rückgriff nehmen wollen. Durch Verknüpfung mit weiteren Informationssystemen wird so ein multimedialer Unterricht ermöglicht, der neben Bildinformationen vor allem auch akustische Signale zur Unterstützung verarbeiten kann.

Insgesamt bleibt zu konstatieren, daß der Systemaufbau für diverse Interessentenkreise deutliche Vorteile mit sich bringt: Neben der Möglichkeit für Professoren, betriebswirtschaftliche Methodik durch Modellintegration zu vermitteln, wird einer der Gründe für die letztlich doch erfolgreiche Verbreitung des Btx-Systems der TELEKOM als Trägermedium und Benutzeroberfläche offenbar. Zudem zeigt die Realitätsnähe der Anwendung der interessierten Praxis, daß informationstechnologische Innovationen im universitären Unterricht praxisnah umgesetzt werden.

¹¹ Vgl. Schmidt, R.: Expertensysteme zur Unterstützung der strategischen Planung, in: K. Spremann; E. Zur (Hrsg.): Informationstechnologie und strategische Planung, Wiesbaden, 1989, S. 255-274.

Literaturverzeichnis

- AMARIS : Softwarebedienungshandbuch für AMARIS-Btx/2, Bielefeld, 1991
- Borland : Benutzerhandbuch für Turbo Pascal, Vers. 6.0, o.O., 1991
- Eistert, T.: Electronic Banking und Cash Management Systeme, Studie zur Wirtschaftsinformatik, Nr. 1, Stuttgart, 1991
- Hinz, H.: Some Advances in Supporting International Financial Management Decisions, in: Journal of Financial Management and Analysis, Januar/Juni, 1990, S. 50-57
- IBM Deutschland GmbH: Btx-API, Definition of a Unique Application Program Interface for Btx-Decoders, Stuttgart, 1989.
- Kimms, A.: MS-DOS für Anwender, WARO-Fachbuchverlag, 2. Aufl., Kiel, 1991
- Kuhlmann, E.: Vom Btx zum Bildinformationssystem, in: L.J. Heinrich;G. Pomberger; R. Schauer (Hrsg.): Die Informationswirtschaft im Unternehmen, Linz, 1991, S. 99-119
- Müller, M.: Vergleichende Analyse existierender Cash Management-Systeme aus Anwendersicht, Diplomarbeit, Lehrstuhl für Finanzwirtschaft, Kiel, 1990
- o.V.: Treasury Workstations, Corporate Finance, Supplement, October 1990, S. 25-26
- Schmidt, R.: Expertensysteme zur Unterstützung der strategischen Planung, in: K. Spremann;E. Zur (Hrsg.): Informationstechnologie und strategische Planung, Wiesbaden, 1989, S. 255-274

Schrage, L.: Programming with LINDO, User's Manual, Palo Alto, 1984

Anhang A
Aufbau von Btx-Zugriffsdateien

Btx-Zugriffsdateien sind reine ASCII-Dateien, die die für den Zugriff auf einen Kontostand bei einer bestimmten Bank notwendigen Informationen enthalten. Sie können mit einem beliebigen Textverarbeitungsprogramm angelegt und bearbeitet werden.

Die ersten sechs Zeilen einer Btx-Zugriffsdatei müssen Informationen über die Position des Kontostandes auf der Btx-Seite, sowie Informationen über die Darstellung des Kontostandes enthalten.

Außerdem muß hier die Bankleitzahl der Bank, und der Name einer Datei, die eine Transaktion von dieser Bank beschreibt, enthalten sein.

Dazu sind die folgenden sechs Parameter in beliebiger Reihenfolge festzulegen:

ZEILE=xxx

Der Wert xxx bestimmt die Zeile der Btx-Seite, in der der Kontostand zu finden ist. Dabei sind die Zeilen von oben nach unten, beginnend bei Zeilennummer 1, durchzuzählen.

SPALTE=yyy

Der Wert yyy bestimmt die Spalte der Btx-Seite, in der das letzte Zeichen des Kontostandes zu finden ist.

Beispiel: 3.327,00 H

^

Die Spalten sind dabei von links nach rechts, beginnend bei Spaltennummer 1, durchzuzählen.

NACHKOMMA=z

Das Zeichen z ist das Zeichen, das als Trennsymbol von Vor- und Nachkommastellen benutzt wird.

TAUSENDER=t

Das Zeichen t ist das Zeichen, das als Trennsymbol hinter Tausenderstellen benutzt wird. Werden Tausenderstellen nicht durch ein Trennsymbol gekennzeichnet, so kann die Angabe des Zeichens t, nicht jedoch die Angabe des Parameters TAUSENDER entfallen.

BLZ=bbbbbbbbb

Die Ziffernfolge bbbbbbbb entspricht der Bankleitzahl der Bank.

TRANSFER=datname

Die Zeichenfolge datname wird als Name einer Textdatei interpretiert, die die Tastenfolge enthält, die zum Überweisen von dieser Bank via Btx notwendig ist.

Hinter den Parameterangaben in einer Btx-Zugriffsdatei sind diejenigen Tastenfolgen anzugeben, die diejenige Btx-Seite einer Bank anwählen, die den Kontostand enthält. Dabei dürfen im allgemeinen mehrere Zeichen in einer Zeile stehen. Ausnahmen bilden hier Btx-Spezialzeichen * und #, die jeweils in einer eigenen Zeile auftreten müssen, um sie von den Schriftzeichen Stern- und Nummernsymbol unterscheiden zu können. Tastenfolgen, hinter denen auf den Aufbau der Btx-Seite gewartet werden muß, sind ebenfalls in einer eigenen Zeile anzugeben.

Soll hinter der Tastenfolge einer Zeile nicht auf den Aufbau der Btx-Seite gewartet werden (= Warten auf ein DCT Signal), so ist an das Ende einer solchen Zeile das Symbol \ zu stellen.

An denjenigen Stellen der Btx-Zugriffsdatei, an denen eine konkrete Kontonummer anzugeben ist, kann stattdessen das Schlüsselwort

KONTONUMMER

in einer eigenen Zeile eingefügt werden. Dieses Wort wird während der Programmausführung durch einen konkreten Wert ersetzt.

Beispiel Btx-Zugriffsdatei:

```
ZEILE=12
SPALTE=40
NACHKOMMA=,
TAUSENDER=.
BLZ=76398621
TRANSFER=test
*\
76523\
#
2
19
#
KONTONUMMER
019
54187
19
1
1
1
```

Anhang B

Aufbau von Btx-Transferdateien

Btx-Transferdateien sind reine ASCII-Dateien und enthalten wie Btx-Zugriffsdateien eine Abfolge von Tasten. Transferdateien beschreiben den Ablauf einer Überweisung von einer Bank. In einer Transferdatei gibt es keine Parameter.

An den Stellen, an denen zur Laufzeit konkrete Werte ersetzt werden sollen, sind folgende Schlüsselworte zu verwenden:

KONTONUMMER

Kontonummer der Bank, von der etwas überwiesen werden soll
(= Bank für die die Transferdatei eingerichtet wird).

EMPFÄNGER-KONTONUMMER

Kontonummer eines Kontos der Bank, an die etwas überwiesen werden soll.

BLZ

Bankleitzahl der Bank, an die etwas überwiesen werden soll.

DM

DM-Betrag, der überwiesen werden soll.

PFENNIGE

Pfennig-Betrag, der überwiesen werden soll.

Beispiel Transferdatei:

```
*\  
76523\  
#  
2  
19  
#  
KONTONUMMER  
019  
54187  
19  
2  
1  
#  
BLZ\  
EMPFÄNGER-KONTONUMMER\  
Harry Hirsch\  
DM\  
PFENNIGE\  
Optimierung\  
#\  
19
```

Anhang C

Optimierungsproblem

Zur Lösung der Optimierungsaufgabe wird in der Prototyp-Implementierung das OR-Softwarepaket LINDO eingesetzt, welches zur Lösung linearer Optimierungsprobleme dient.

Der Benutzer hat eine ASCII-Datei PROBLEM.LP anzulegen, in der das zu lösende LP in LINDO-Syntax formuliert ist. Transferwege zwischen einem Konto i und einem Konto j müssen dabei durch Variablen der Form X_{ij} mit $1 \leq i, j \leq 9$ bezeichnet werden. Dabei repräsentiert ein Index k mit $1 \leq k \leq 9$ das k -te selektierte Konto.

(FUSSNOTE: In der derzeitigen Implementierung ist die Optimierung damit auf neun Konten beschränkt worden, um die programmgesteuerte Erkennung der Indizes i und j einer Variablen X_{ij} zu vereinfachen. Eine Erweiterung auf eine unbeschränkte Anzahl von Konten wäre z.B. dadurch denkbar, daß man Variablen der Form X_{a_b} zuläßt, wobei a und b mehrere Ziffern darstellen können). Kontostände müssen durch folgende Bezeichner dargestellt werden, wobei x für das x -te selektierte Konto steht:

KONTO x - Kontostand

ABS_KONTO x - Absolutbetrag des Kontostandes

NEG_KONTO x - Minuseinsfache des Kontostandes

Die Einführung weiterer Schlüsselworte, wie z.B. LIMIT x , zur Darstellung eines Kreditlimits ist vorstellbar, jedoch zur Zeit nicht implementiert.

In der ersten Zeile der Datei muß sich ein Eintrag der Form BANKEN= anz befinden, um anzudeuten, daß das Problem für eine Anzahl von anz Banken definiert ist.

Zeilen, die mit einem *-Symbol beginnen, werden als Kommentare angesehen und ignoriert.

Beispiel Optimierungsproblem:

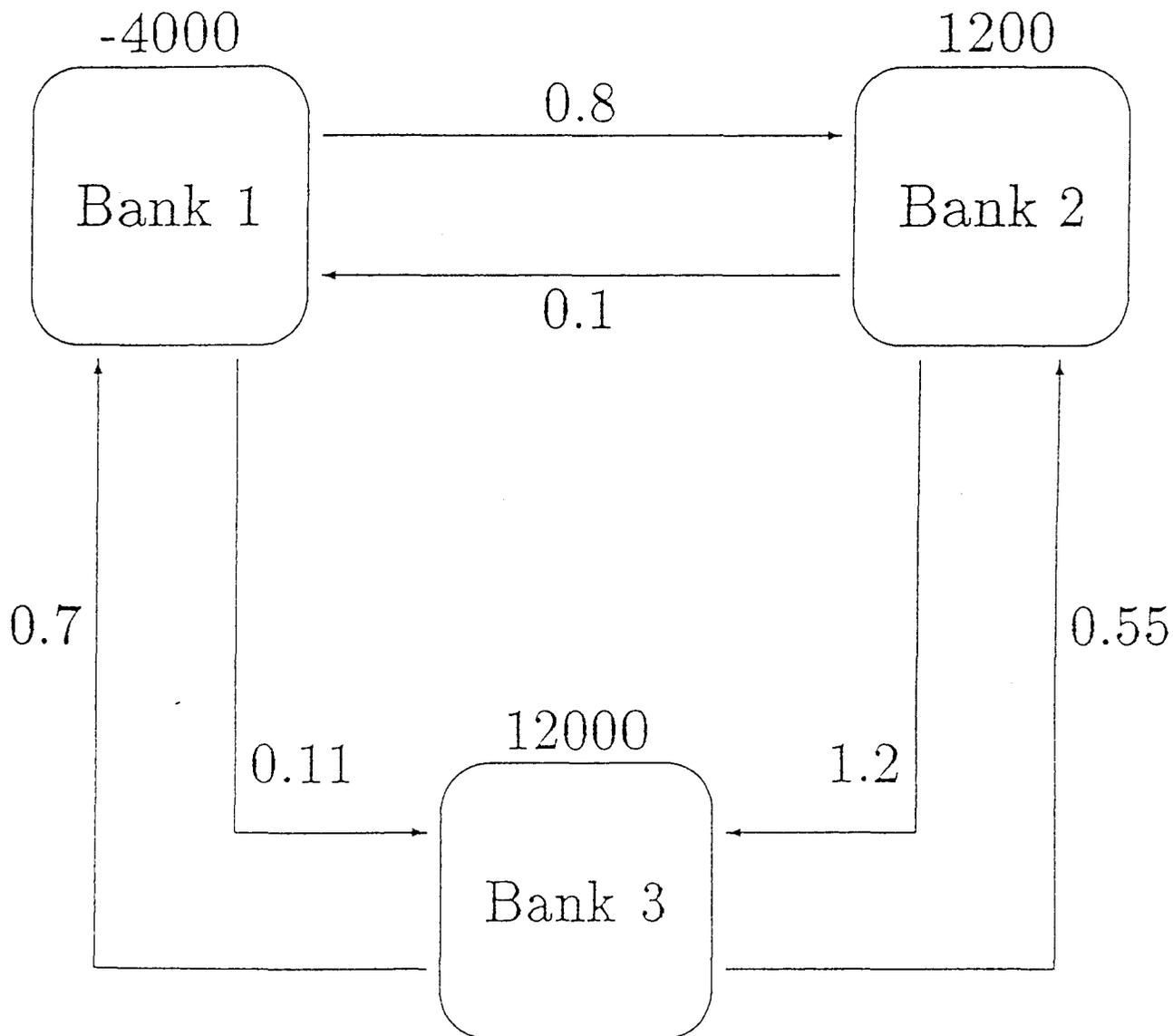
BANKEN=3

```

*****
*** Minimiere die Überweisungskosten zwischen drei Banken.
*** Die Kosten sind proportional zum überwiesenen Betrag.
*****
min 0.11 x13 + 0.8 x12 + 1.2 x23 + 0.1 x21 + 0.55 x32 + 0.7 x31
st
*****
*** Die Konten der Banken 1 und 3 sollen auf Null abgeglichen wer-
den.
*****
KONTO1 + x21 + x31 - x13 - x12 = 0
KONTO3 + x13 + x23 - x32 - x31 = 0
*****
*** Zwischen Bank 2 und 3 werden pro Richtung max. 7000 DM über-
wiesen.
*****
x32 <= 7000
x23 <= 7000
end

```

Anhang D
Transferstruktur des Beispiels



Legende: [Kantenkoeffizienten] Transferkosten je DM
[Knotenkoeffizienten] Anfangsgeldbestände