BERUFSAKADEMIE KARLSRUHE
University of Cooperative Education

Studienbereich Wirtschaft

Studienfach Wirtschaftsinformatik

Entwurf einer revisionssicheren Kundenakte für die AddOns der cormeta ag

---

Diplomarbeit (Exzerpt)

Im Rahmen der Prüfung zum Diplom-Wirtschaftsinformatiker (BA)

Fachrichtung Wirtschaftsinformatik
Kurs: WWI05V1
Ausbildungsstätte: cormeta ag

Verfasser: Dennis Schulmeister
Matrikelnummer: 192932

Betreuender Dozent: Klaus Biedermann
Betrieblicher Betreuer: Dr. Christoph Wegener

Abgabedatum: Montag, 5. Mai 2008

Unterschrift,Verfasser:

Note:

Danksagung

Folgenden Personen möchte ich an dieser Stelle meinen persönlichen Dank aussprechen:

- Herrn Dr. Christoph Wegener für die umfassende Unterstützung während der Anfertigung der Arbeit und die allzeit kritische Auseinandersetzung mit den Inhalten, welche sehr zum Gelingen beigetragen hat.
- Frau Ursula Westermann für die zahlreichen, konstruktiven Dialoge, deren Ergebnisse in die Untersuchung mit einflossen.
- Herrn Michael Rawolle, Herrn Frank Holetzek sowie meinen verbleibenden Kollegen, die in allen Fragen stets ein offenes Ohr für mich hatten.
- Und letztlich meinen Freunden und meiner Familie, die mich nur allzu oft an den Computer ausleihen mussten.

Karlsruhe, im Mai 2008

Dennis Schulmeister

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit mit dem Titel *Entwurf einer revisionssicheren Kundenakte für die AddOns der cormeta ag* in selbstständiger Arbeit ohne Zuhilfenahme anderer als der angegebenen Hilfsmittel erstellt habe. Aus den benutzten Quellen direkt und indirekt übernommene Gedanken habe ich als solche kenntlich gemacht.

Diese Arbeit wurde bisher in gleicher oder ähnlicher Form auch auszugsweise weder veröffentlicht noch bei einer Prüfungsbehörde vorgelegt.

Karlsruhe, 5. Mai 2008 _____

(Unterschrift)

Inhaltsübersicht

	Danksagung	**I**
	Eidesstattliche Erklärung	**II**
	Inhaltsverzeichnis	**IV**
	Abbildungsverzeichnis	**V**
1	**Betriebswirtschaftliche Aspekte**	**VI**
1.1	Der innerbetriebliche Informationsfluss	VI
1.2	Bedeutung der Revisionssicherheit	VIII
2	**Schwerpunkte der technischen Realisierung**	**XIII**
2.1	Asynchrone Methodenaufrufe mit der Ereignissteuerung	XIII
2.2	Objektorientiertes Datenmodell mit Indizierung	XVI
2.3	Dokumentablage mithilfe des Business Document Service	XX
3	**Integration der SAP-WebFlow-Engine**	**XXIII**
3.1	Beschreibung des SAP-Workflowsystems	XXIII
4	**Schlussbetrachtung**	**XXVI**
4.1	Quintessenzen der Untersuchung	XXVI
4.2	Resumé	XXVIII
A	**Quellenangaben**	**A**

Inhaltsverzeichnis

	Danksagung	**I**
	Eidesstattliche Erklärung	**II**
	Inhaltsübersicht	**III**
	Abbildungsverzeichnis	**V**
1	**Betriebswirtschaftliche Aspekte**	**VI**
1.1	Der innerbetriebliche Informationsfluss	VI
1.1.1	Bedeutung des innerbetrieblichen Informationsflusses	VI
1.1.2	Realisierung eines Informationsflusses mit Workflows	VII
1.2	Bedeutung der Revisionssicherheit	VIII
1.2.1	Rechtliche Grundlagen	VIII
1.2.2	Leitsätze der revisionssicheren, elektronischen Archivierung	X
2	**Schwerpunkte der technischen Realisierung**	**XIII**
2.1	Asynchrone Methodenaufrufe mit der Ereignissteuerung	XIII
2.1.1	Funktionsweise der Ereignissteuerung	XIII
2.1.2	Anwendungsbeispiel zur Ereigniserzeugung	XIV
2.2	Objektorientiertes Datenmodell mit Indizierung	XVI
2.2.1	Objektorientierte Repräsentation der Einträge und Metadaten	XVI
2.2.2	Transfer der Klassen in die Datenbank	XVII
2.2.3	Verwaltung und Indizierung aller Akteninhalte	XVIII
2.3	Dokumentablage mithilfe des Business Document Service	XX
3	**Integration der SAP-WebFlow-Engine**	**XXIII**
3.1	Beschreibung des SAP-Workflowsystems	XXIII
4	**Schlussbetrachtung**	**XXVI**
4.1	Quintessenzen der Untersuchung	XXVI
4.2	Resumé	XXVIII
A	**Quellenangaben**	**A**

Abbildungsverzeichnis

Abb. 1: Direkte und Indirekte Kommunikation XIV
Abb. 2: Auslösen eines Ereignisses XV
Abb. 3: Polymorphe Datenbankoperationen XVIII
Abb. 4: ER-Diagramm der Indextabellen XIX
Abb. 5: Signaturen zur Beschreibung von Komponenten XXI
Abb. 6: Architektur des SAP-Workflowsystems XXIV

1 Betriebswirtschaftliche Aspekte

Dieses Kapitel beleuchtet die Kundenakte unter dem Aspekt des betriebswirtschaftlichen Nutzens und steckt dabei das Aufgabengebiet der Kundenakte ab. Ziel des Vorgehens ist die Gewinnung von Anforderungen, welche durch die Kundenakte umgesetzt werden müssen.

1.1 Der innerbetriebliche Informationsfluss

1.1.1 Bedeutung des innerbetrieblichen Informationsflusses

Die Information ist die Basis einer jeden Entscheidung. Denn ohne eine ausreichende Informationsbasis kann keine Entscheidung getroffen werden und die Qualität einer Entscheidung hängt maßgeblich von den Informationen ab, auf denen sie beruht. Dabei ist es prinzipiell wünschenswert, dass jegliches Entscheidungsproblem ein gut strukturiertes Entscheidungsproblem[1] darstellt, das allein auf vollkommenen Informationen gelöst werden kann. Es ist also erstrebenswert, die Konsequenzen jeder Handlungsalternative a priori abschätzen zu können, wobei nur primäre Informationen ausgewertet werden, deren Informationsgehalt vollends zur Problemlösung beiträgt. Somit kommt dem Vorhandensein von Informationen sowohl in quantitativer als auch qualitativer Hinsicht eine große Bedeutung zu.

Natürlich ist der gewünschte Idealfall in der Realität nur selten gegeben, so dass die genauen Auswirkungen einer Entscheidung nur abgeschätzt werden können, weil oftmals nur unvollständige Informationen mit unbestimmtem Informationsgehalt vorliegen. Doch selbst wenn die notwendigen Informationen vorhanden sind, muss erst das Wissen um ihre Existenz und um ihren Zugriff bestehen. Eine vorhandene Information, die nicht wiedergefunden kann ist nicht mehr wert als eine fehlende Information. In der Tat ist der Unterschied zwischen beiden nur gering.

Die Relevanz dieser Thematik ergibt sich, wenn man einen Blick auf den Kreditmanagementprozess wirft, den die AddOns abdecken. Anhand einer Umfrage unter 21 Großunternehmen stellt [BROE00] (S. 20-23) fest, dass sich drei Organisationsformen bezüglich der Verantwortlichkeiten innerhalb des Prozesses unterscheiden lassen: Das zentrale Kreditmanagement, das dezentrale Kreditmanagement und das teilzentrale Kreditmanagement.

1 Vgl. [ADAM96] (S. 9-10) zu den Kriterien eines gut strukturierten Entscheidungsproblems.

Das zentrale Kreditmanagement zeichnet sich dadurch aus, dass nahezu der komplette Kreditmanagementprozess von einer Fachabteilung durchgeführt wird, welche nicht nur alle Entscheidungen trifft, sondern auch für die anschließende Durchführung dieser verantwortlich zeichnet. In der Regel ist diese Abteilung sehr klein, so dass zumindest bezüglich der internen Informationen kein Wissensmangel besteht.

Im Gegensatz dazu sind das dezentrale und das teilzentrale Kreditmanagement oft historisch gewachsen, wobei eine rege Aufgabenteilung zwischen den verschiedenen Bereichen vorherrscht, welche sich jeweils nur auf einen Teil des Kreditmanagementprozesses konzentrieren. Dabei kommt es nicht selten vor, dass eine Abteilung die Entscheidungen einer anderen Abteilung durchführen muss und dass als Grundlage einer Entscheidung Informationen aus den anderen Abteilungen herangezogen werden[2]. Hier ist der Informationsfluss wesentlicher Erfolgsfaktor für die Effizienz des gesamten Prozesses.

Somit wird lässt sich die Bedeutung des unternehmensinternen Informationsaustauschs auch für die AddOns erkennen. Sie alle erzeugen Informationen, die in den nachfolgenden Schritten des Kreditmanagementprozesses von Bedeutung sein können. Für ein zentrales Kreditmanagement stellt dies kein Problem dar, da sich die Nutzung der AddOns im Wesentlichen auf einen Fachbereich beschränkt. In allen anderen Fällen muss die abteilungsübergreifende Kommunikation der Informationen sichergestellt werden, weil sie nur dann im verbleibenden Prozessablauf verwendet werden können. Teilweise wird dies durch die AddOns selbst durchgeführt, indem spezifische Schnittstellen zwischen den AddOns zur gegenseitigen Benachrichtigung und Abfrage ausgenutzt werden. Dies ist allerdings nur dann praktikabel, wenn die Prozesshoheit von einem AddOn an ein anderes übergeben werden soll. Eine allgemeingültige Lösung ist dadurch nicht gegeben.

1.1.2 Realisierung eines Informationsflusses mit Workflows

Im Umfeld der Festlegung von Zahlungszielen stellt [KOKA00] (S. 123-124) fest, dass durch die Definition einheitlicher, betriebsinterner Vorgehensweisen, welche auch die geschäftspolitischen Grundsätze der Unternehmung sowie klare, organisatorische Regelungen berücksichtigen, sowohl die Rechnungsstellung als auch die nachgelagerte Kontrolle des Zahlungseingangs wesentlich erleichtert wird. Dabei lässt sich eine Relation zur Unternehmensgröße erkennen, in dem Sinne, dass derartige Regelungen bei kleineren Unternehmen häufig fehlen, während sich größere Unternehmen zunehmend an den Gepflogenheiten der

2 Vgl. hierzu die restlichen Umfrageergebnisse, die [BROE00] auf den Seiten 12-19 präsentiert.

eigenen Branche orientieren. In diesem Zusammenhang gewinnt auch die Frage an Bedeutung, welche Instanz über von den üblichen Regeln abweichende Konditionen entscheiden darf. Erwartungsgemäß obliegt diese Aufgabe der Geschäftsführung, wobei auch hier erkannbar ist, dass bereits bei Unternehmen ab 100 Mitarbeitern komplexere Zuständigkeiten bestehen.

Solcherlei Szenarios sind prädestiniert für die Abwicklung durch Workflows, deren traditionelles Einsatzgebiet laut [RICK02] (S. 32) eben die Realisierung von wiederholbaren Entscheidungsprozessen ist. Insbesondere sind Workflows gerade deshalb gut geeignet, da sie die Ermittlung von Sachbearbeitern in Abhängigkeit ihrer Aufgaben oder ihrer hierarchischen Stellung unterstützen, so dass die statische Adressierung von Empfängern vermieden wird. Doch auch ohne konkreten Handlungsbedarf schlägt [BRAH03] (S. 159-164) den Einsatz von Informationsworkflows vor, welche dazu genutzt werden können, im SAP-System erzeugte Informationen unmittelbar zu kommunizieren. (...)

1.2 Bedeutung der Revisionssicherheit

1.2.1 Rechtliche Grundlagen

Obwohl die Revisionssicherheit bei der historischen Dokumentation von handelsrechtlichen Unterlagen von großer Bedeutung ist, bietet der nationale Gesetzgeber keine direkte Definition, welche zur Interpretation des Begriffs herangezogen werden kann. Stattdessen bestehen im Handelsgesetzbuch und in der Abgabenordnung allgemeine Regelungen über den Umgang mit Handelsbriefen, die sich zwar primär auf Dokumente in Papierform beziehen, aber sinngemäß auf digitale Daten übertragen werden können.

Grundlage dieser Regelungen ist der §238 Abs. 1 [HGB], aus dem die allgemeine Buchführungspflicht resultiert, mit der Maßgabe, dass sich alle Geschäftsvorfälle in ihrer Entstehung und Abwicklung her verfolgen lassen können müssen, wodurch die Nachvollziehbarkeit und insbesondere die Beweisfähigkeit der Buchführung deklariert wird. Sinngemäß verpflichten §238 Abs. 2 [HGB] sowie §257 Abs. 1 [HGB] den Kaufmann, eine wortgetreue Wiedergabe aller versandten und die Originale aller empfangenen Handelsbriefe aufzubewahren, wobei §257 Abs. 2 [HGB] diese als sämtliche Schriftstücke definiert, die ein Handelsgeschäft tangieren. Hierbei verlangt §257 Abs. 5 [HGB] eine Aufbewahrungsfrist von sechs Jahren ab Ablauf des Kalenderjahres.

Bezüglich der Revisionssicherheit der Aufzeichnungen erscheint §239 Abs. 2 [HGB] interessant. Dieser legt fest, dass alle Eintragungen vollständig, wahrheitsgemäß, zeitgerecht und geordnet vorgenommen werden müssen. Dabei sind gemäß §239 Abs. 3 [HGB] keinerlei Manipulationen bereits bestehender Einträge zugelassen, welche den ursprünglichen Inhalt verwischen oder ungewiss lassen, ob es sich überhaupt um Änderungen handelt und wenn ja, wann diese getätigt wurden.

Über die Besonderheiten von Datenverarbeitungssystemen trifft das Handelsgesetzbuch bis auf die Aussage der §239 Abs. 4 und §257 Abs. 1, dass elektronisch verwaltete Buchdaten innerhalb der allgemeinen Aufbewahrungsfristen entsprechend den Grundsätzen ordnungsgemäßer Buchführung zu führen sind und dass sie innerhalb einer angemessenen Frist, welche nicht näher definiert wird, wieder lesbar gemacht werden können müssen, keine Aussage. Selbiges trifft auf die Abgabenordnung zu, deren Regelungen der §145, §146 und §147 [AO] auch nicht näher ins Detail gehen. Grundsätzlich handelt es sich bei ihnen ohnehin um Abschriften der HGB-Regelungen, welche um die Bedürfnisse der Steuerprüfung erweitert wurden. Zum Beispiel ist der §147 [AO] analog zu §257 [HGB] formuliert, mit der Ergänzung der Absätze 5 und 6, die der Finanzbehörde den Zugriff auf steuerrelevante Daten einräumen.

Lohnenswerter erscheint darum ein Blick in die *Grundsätze zum Datenzugriff und die Prüfbarkeit digitaler Unterlagen*, wie sie in einem BMF-Schreiben des Bundesfinanzministeriums vom 16. Juli 2001 veröffentlicht wurden. Sie entstammen ebenfalls dem Hintergrund der steuerrechtlichen Betriebsprüfung, konkretisieren allerdings die allgemeinen Anforderungen des Handelsgesetzbuches und der Abgabenordnung hinsichtlich des Zugriffs auf die digitalen Unterlagen. Insbesondere Abschnitt II.2 befasst sich mit der Prüfbarkeit aufbewahrungspflichtiger Daten und stellt eine Verbindung zu den *Grundsätzen ordnungsgemäßer DV-gestützter Buchführungssysteme* – ebenfalls vom Bundesfinanzministerium – her, indem gefordert wird, dass sämtliche digitalisierte Daten, welche im Sinne des §147 Abs. 1 [AO] aufbewahrungspflichtig sind und nicht in Papierform vorliegen, gemäß den GoBS behandelt werden. Insbesondere muss dabei der Originalzustand der Daten erkennbar bleiben, so dass nachträgliche Veränderungen zu vermeiden sind. Darüber hinaus ist der Eingang, die Archivierung, die Konvertierung und die Weiterverarbeitung der Daten zu protokollieren.

Die GoBS wiederum beschränken sich zwar auf die digitale Buchhaltung, doch wie [KAMP03] (S. 10) anführt, wird gleich im ersten Absatz darauf hingewiesen, dass die selben

Anforderungen, die der Gesetzgeber an ein kaufmännisches Buchhaltungssystem stellt, gleichermaßen auf ein betriebliches Dokumentenmanagementsystem übertragen werden können. So referenziert Sektion 1.2 [GoBS] direkt die §238, §239 [HGB] und §145, §146 [AO] und ergänzt:

- Jedweder Geschäftsvorfall muss richtig, vollständig und zeitgerecht erfasst sein und sich in seiner Entstehung und Abwicklung verfolgen lassen.
- Alle Geschäftsvorfälle sind so zu verarbeiten, dass sie überblicksartig geordnet darstellbar sind.
- Einem Sachverständigen muss es möglich sein, sich binnen kurzer Zeit zurechtzufinden und einen Überblick zu gewinnen.
- Eine Verfahrensdokumentation soll das genaue Vorgehen beschreiben und es muss gewährleistet sein, dass sich die Software gemäß der Dokumentation verhält.

Darüber hinaus betont Passus 5.5.1 die Wichtigkeit von Zugriffs- und Zugangskontrollen. Diese sind derart auszugestalten, dass selbst berechtigte Personen nur in dem Umfang Zugriff auf die gespeicherten Daten erlangen können, wie es zur Erfüllung ihrer Aufgaben notwendig ist.

1.2.2 Leitsätze der revisionssicheren, elektronischen Archivierung

Wenngleich die rechtlichen Bestimmungen keine allgemeingültige Definition der Revisionssicherheit liefern mögen, lassen sich ihre Intensionen wie folgt zusammenfassen:

- Vollständigkeit: Die Dokumentation aller relevanten Informationen hat fortlaufend und lückenlos zu erfolgen. Es dürfen keine Informationen ausgelassen werden.
- Ordnungsmäßigkeit und Verfahrenssicherheit: Die Dokumentation hat nach reproduzierbaren Verfahren zu erfolgen, welche selbst in einer Verfahrensanweisung beschrieben werden. Darüber hinaus ist dem Datenverlust durch geeignete Sicherungsmaßnahmen Vorsorge zu tragen.
- Schutz vor Verfälschung: Einmal archivierte Daten dürfen nicht nachträglich verfälscht werden. Jegliche Änderungen sind als Kopie zu kennzeichnen, ohne dass das Original verloren geht. Ebenfalls müssen die durchgeführten Änderungen derart protokolliert werden, dass der Originalzustand wiederhergestellt werden kann.

- **Nachvollziehbarkeit und Überprüfbarkeit:** Als Folge der obigen Punkte ergibt sich, dass nicht nur das Archivierungsverfahren überprüfbar ist, sondern dass auch die Zusammenhänge der erfassten Daten jederzeit und unverfälscht nachvollzogen werden können.

- **Zugriffskontrolle:** Im Falle der Archivierung von sensiblen Daten, wie zum Beispiel Buchungsdaten, ist der Zugriff auf die Daten bezüglich des Personenkreis und der einsehbaren Daten soweit wie möglich einzuschränken. Einer berechtigten Person sollen nicht mehr Zugriffe erlaubt sein, als für die Erfüllung ihrer Aufgaben notwendig.

Dabei trifft der Gesetzgeber keine konkreten Aussagen über die Herkunft oder Beschaffenheit der aufzubewahrenden Dokumente, so dass die Wirksamkeit der gesetzlichen Regelungen allein an den funktionalen Aspekt geknüpft ist. Das Handelsgesetzbuches bezieht sich auf alle Handelsbriefe und sämtliche mit der Erstellung des Jahresabschlusses zusammenhängende Dokumente. Die Abgabenordnung und die GDPdU fokussieren eine rein steuerrechtliche Revision und die GoBS betrachten nur buchhalterische Dokumente.

Vor diesem Hintergrund hat der *Verband Organisationssysteme und Informationssysteme* den Begriff der revisionssicheren, elektronischen Archivierung geprägt, der die gesetzlichen Bestimmungen verallgemeinert. Vom selben Verband stammen auch die folgenden zehn Merksätze, welche die Anforderungen an die Revisionssicherheit subsumieren ([KAMP03] S. 24-26):

1. Jedes Dokument muss unveränderbar archiviert werden.

2. Es darf kein Dokument auf dem Weg ins Archiv oder im Archiv selbst verloren gehen.

3. Jedes Dokument muss mit geeigneten Retrievaltechniken wieder auffindbar sein.

4. Es muss genau das Dokument wiedergefunden werden, das gesucht worden ist.

5. Kein Dokument darf während seiner vorgesehenen Lebenszeit zerstört werden können.

6. Jedes Dokument muss in genau der gleichen Form, wie es erfasst wurde, wieder angezeigt und gedruckt werden können.

7. Jedes Dokument muss zeitnah wiedergefunden werden können.

8. Alle Aktionen im Archiv, die Veränderungen in der Organisation und Struktur bewirken, sind derart zu protokollieren, dass die Wiederherstellung des ursprünglichen Zustandes möglich ist.

9. Elektronische Archive sind so auszulegen, dass eine Migration auf neue Plattformen, Medien, Softwareversionen und Komponenten ohne Informationsverlust möglich ist.

10. Das System muss dem Anwender die Möglichkeit bieten, die gesetzlichen Bestimmungen (BDSG, AO, etc.) sowie die betrieblichen Bestimmungen des Anwenders hinsichtlich Datensicherheit und Datenschutz über die Lebensdauer des Archivs sicherzustellen.

2 Schwerpunkte der technischen Realisierung

Der folgende Teil der Diplomarbeit fokussiert die Kernbereiche der Kundenakte mit Blick auf ihre mögliche Implementierung. Dabei werden ausgehend vom architektonischen Gesamtmodell der Kundenakte konkrete Lösungsvorschläge unterbreitet und ihre Anwendung diskutiert. Davon ausgenommen ist die Integration des SAP-Workflowsystems, da diese in einem eigenen Kapitel behandelt wird.

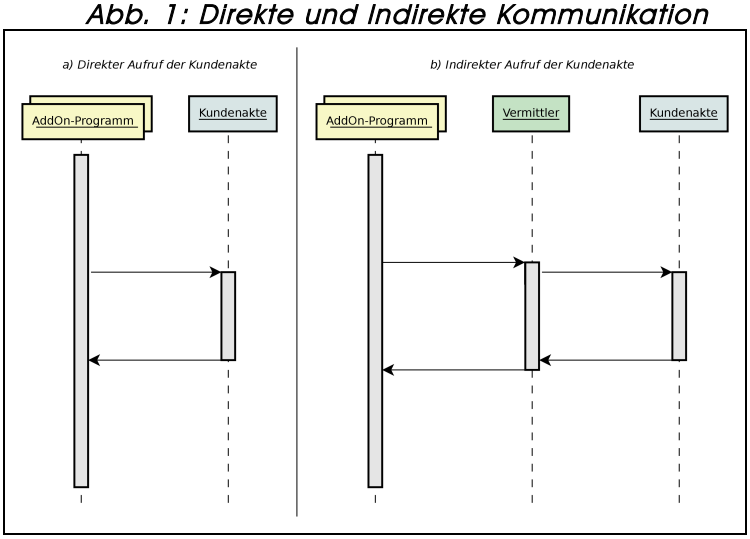
2.1 Asynchrone Methodenaufrufe mit der Ereignissteuerung

2.1.1 Funktionsweise der Ereignissteuerung

Das Funktionieren der Kundenakte setzt voraus, dass für die AddOns eine sichere Möglichkeit besteht, Einträge an die Kundenakte zu melden. Da aus Sicht der AddOns die Existenz der Kundenakte jedoch nicht vorausgesetzt werden kann, darf das Funktionieren der AddOns bei Nichtvorhandensein der Kundenakte nicht beeinträchtigt werden. Basierend auf dieser Überlegung wird darum die Anforderung gestellt, nur eine indirekte Kommunikation zwischen den AddOns und der Kundenakte zuzulassen, welche einen im SAP-Standard vorhandenen Vermittler einsetzt, der die Sicherheit der Kommunikation gewährleistet.

Im Kontext von verteilten Systemen schlägt [TANE03] (S. 122-127 sowie S. 132-135) vor, dass diese Ziele durch den Austausch von Nachrichten erreicht werden können. Denn im Gegensatz zu andersartigen Kommunikationstechnologien, wie dem direkten Aufruf einer entfernten Prozedur oder Methode, kommt dabei immer ein Kommunikationssystem zum Einsatz, das die übermittelten Nachrichten sowohl asynchron als auch persistent behandelt. Das heißt, der Sender wird während der Übermittlung seiner Nachrichten nicht blockiert oder andersartig beeinträchtigt, während das Kommunikationssystem autonom versucht, die Nachrichten zuzustellen. Sollte dies vorübergehend nicht möglich sein, werden die Nachrichten zwischengespeichert und verschiedene Heuristiken angewandt, um die Zustellung dennoch zu erreichen. Somit hat der Sender zwar nicht Gewissheit, dass seine Nachrichten tatsächlich empfangen werden, das System garantiert aber, dass die Nachrichten empfangen werden können. [ANGE03] (S. 126-127) fasst dies unter dem Begriff *Guaranteed Delivery* zusammen.

Abb. 1: Direkte und Indirekte Kommunikation



Quelle: Eigene Darstellung

An diesem Punkt setzt die Ereignisbehandlung von SAP an, welche einen systeminternen Nachrichtendienst nachbildet und auch verwendet wird, um Workflows auf Basis von Ereignissen anzustoßen. Sie stellt einen anwendungsübergreifenden Ereignismanager bereit, der es Anwendungen ermöglicht, Ereignisse zu triggern, ohne dass diese von derselben Anwendung bearbeitet werden. Die Behandlung der Ereignisse erfolgt immer in einem eigenen Programmkontext durch so genannte Ereignisverbraucher, wobei es sich konkret um Funktionsbausteine, Methoden von Business Objekten und Methoden von ABAP-Objekten handeln kann.

Nach der Beschreibung von [MEND00] (S. 225) handelt der Ereignismanager nach einer strikten Event-Condition-Action-Taktik. Das heißt, für jedes auftretende Ereignis werden zunächst alle in Frage kommenden Verbraucher ermittelt. Anschließend wird für jeden Verbraucher durch spezielle Funktionsbausteine geprüft, ob die Verarbeitung tatsächlich angestoßen werden soll und im letzten Schritt werden alle Verbraucher aktiviert, deren Prüfung positiv ausfiel. Wenn kein Verbraucher gefunden wird, der bereit ist, das Ereignis zu behandeln, wird es verworfen.

2.1.2 Anwendungsbeispiel zur Ereigniserzeugung

Das folgende Beispiel zeigt, wie ein anwendungsübergreifendes Ereignis ausgelöst werden kann. Hierfür wird ein zu *if_swf_evt_event* typkompatibles Objekt verwendet, das von der statischen Methode *get_instance* der Klasse *cl_swf_evt_event* generiert wird, wobei die Methode in Form dreier Parameter wissen muss, um welches Ereignis es sich handelt. Hierfür

müssen der Ereignistyp[3], die definierende Klasse[4] und der Ereignisname benannt werden. Daraufhin können Informationen über das Ereignis und seine Behandler erfragt, Parameter an das Ereignis übergeben und das Ereignis mit der Methode *raise* ausgelöst werden. Nennenswert dabei ist, dass die Erzeugung des Objekts in jedem Fall funktioniert, selbst wenn die definierende Klasse des Ereignisses nicht existiert oder kein Ereignisverbraucher mit dem Ereignis registriert ist. In diesem Fall wirft die Methode *raise* eine Ausnahme, welche vom aufrufenden Programm abgefangen und behandelt werden kann.

Abb. 2: Auslösen eines Ereignisses

```
* Referenz auf das Event-Objekt
DATA: event TYPE REF TO if_swf_evt_event.

* Ereignisobjekt erzeugen
CALL METHOD cl_swf_evt_event=>get_instance
  EXPORTING
    im_objcateg        = 'CL'
    im_objtype         = 'ZDES_CL_KAKTE'
    im_event           = 'NEW_MESSAGE'
    im_objkey          = 'J6M'
*   IM_EVENT_CONTAINER =
  RECEIVING
    re_event           = event.

* Ereignis auslösen und Fehler behandeln
TRY.
    event->raise( ).
  CATCH cx_swf_evt_invalid_objtype .
    MESSAGE i000(43) WITH 'Ungültiges Objekt.'.
  CATCH cx_swf_evt_invalid_event .
    MESSAGE i000(43) WITH 'Ungültiges Ereignis.'.
ENDTRY.

* Speicher wieder freigeben
FREE event.
```

Quelle: Eigene Darstellung

Eine Besonderheit, die aus dem Beispiel nicht hervor geht, ist die Übergabe von Parametern an das aufgerufene Ereignis. In der Tat sind Ereignisse mit regulären Methoden sehr verwand, weshalb sowohl in der ABAP-Notation als auch im Class Builder beide beinahe identisch definiert werden[5]. Demgemäß können Ereignisse Aufrufparameter besitzen, welche jedem Ereignisverbraucher bei seiner Ausführung mitgegeben werden. Lediglich die Rückgabe von Werten ist aufgrund des asynchronen Aufrufs der Verbraucher nicht möglich.

Wünscht das rufende Programm, ein Ereignis bei seiner Initiierung mit Parametern zu versorgen, muss es sich hierzu eines Parametercontainers bedienen, da die Methode *raise* keine Möglichkeit der direkten Übergabe vorsieht. Dabei handelt es sich um ein besonderes

3 Ereignis eines Business Objekt oder einer ABAP-Klasse

4 Ereignisse gehören immer zu einer Instanz einer ABAP-Klasse oder eines Business Objekts, da sie hauptsächlich dazu verwendet werden, sogenannte Beobachter über Zustandsänderungen des Objekts zu informieren. Eine allgemeine Beschreibung des Beobachter-Patterns findet sich in [GAMM04] (S. 287-291).

5 Vgl. zur Definition von Ereignissen [WOLF07] S. 65 und S. 99

Objekt, welches von der Methode *get_event_container* des Eventobjekts erzeugt wird und einfach eine Liste der zu übergebenen Parameter und ihrer Werte darstellt. Als solche stellt es Methoden wie *set*, *get* oder *clear* zur Manipulation seiner Inhalte zur Verfügung. (...)

2.2 Objektorientiertes Datenmodell mit Indizierung

2.2.1 Objektorientierte Repräsentation der Einträge und Metadaten

Basierend auf den betriebswirtschaftlichen Vorüberlegungen sieht das architektonische Gesamtbild der Kundenakte eine Unterscheidung zwischen originären Einträgen der Kundenakte, welche in einem Verbund von Archivtabellen abgelegt werden können, und Dokumenten, welche mittels des Business Document Service ausgelagert werden können, vor. Bei beiden handelt es sich um Einträge der Kundenakte, welche von der Indexdatenbank erfasst werden müssen. Hinzu kommen Sequenzen, die ebenfalls in der Indexdatenbank enthalten sein müssen, da sie, um geschachtelt werden zu können, ebenfalls als Einträge behandelt werden müssen.

Obwohl sich all diese Eintragsarten in wesentlichen Punkten voneinander abheben, besitzen sie als indizierte Einträge der Kundenakte zahlreiche Gemeinsamkeiten, unter denen sie gebündelt betrachtet werden können. Zum Beispiel besitzen sie alle eine Reihe von Verwaltungsdaten wie den Zeitpunkt der Erzeugung, das Kürzel der auslösenden Softwarekomponente oder die Metamerkmale aus der Indexdatenbank. In einem objektorientierten Modell, das die Trennung der Verantwortlichkeiten akzentuiert, werden derartige Sachverhalte durch Erbschaftsbeziehungen ausgedrückt. Das heißt, es wird eine Oberklasse aller Einträge gebildet, unter welcher die gemeinsamen Attribute und Methoden aller Einträge zusammengefasst werden können. Von dieser wird je Eintragsart eine Klasse abgeleitet, welche die allgemeine Beschreibung eintragsspezifisch komplettiert. Dabei sollte die Mutterklasse, da es keine generischen Einträge geben kann, als abstrakte Klasse definiert werden, von der selbst keine Objektinstanzen erzeugt werden können.

Die Instanzen der Klassen repräsentieren die Einträge der Kundenakte zur Laufzeit eines sie verwendenden Programms. Zu jeder Klasse sollte es darum mindestens eine Tabelle auf der Datenbank geben, welche je persistenter Klasseninstanz einen Eintrag beinhaltet. Dies trifft auch auf die Mutterklasse zu, obwohl von dieser keine eigenständigen Instanzen erzeugt werden sollten. Der Grund hierfür ist zum einen, dass jede Objektinstanz der Subklassen indirekt auch eine Instanz der Mutterklasse ist. Zum anderen muss für jeden Eintrag die Möglich-

keit der Indizierung bestehen, so dass diese Funktionalität von der Mutterklasse erbracht werden kann. Sie stellt somit die Verbindung von den Bestandsdatenbanken zur Indexdatenbank her. Allerdings muss die direkte Entsprechung eines Tabelleneintrages mit einer Klasseninstanz an dieser Stelle aufgelockert werden. Denn durch die Mutterklasse sollen natürlich alle Metamerkmale eines Eintrages verwaltet werden, nicht nur ein einziger.

2.2.2 Transfer der Klassen in die Datenbank

Sowohl bei der Objekterzeugung als auch bei der Ablage von Objektdaten auf der Datenbank muss sichergestellt sein, dass jede Klasse, von der eine Objektinstanz abhängig ist, die Möglichkeit erhält, auf die Datenbank zuzugreifen. Denn andernfalls kann die Vollständigkeit der persistierten Daten nicht gewährleistet werden. Zu einem gewissen Grade steht dies im Gegensatz zur objektorientierten Generalisierung, aufgrund derer die Methoden zum Datenbankzugriff bereits in der obersten Mutterklasse angesiedelt werden müssen, so dass sie von den abhängigen Unterklassen bei jeweils gleicher Aufrufsignatur domänenspezifisch implementiert werden können[6]. Da aber die Subklassen nicht nur bezüglich ihrer Funktionalitäten sondern auch bezüglich ihrer Dateninhalte auf ihrer Superklasse aufbauen, darf, um die Trennung der Verantwortlichkeiten nicht zu gefährden, die in der Superklasse enthaltene Implementierung der Datenbankmethoden trotz Überdefinition nicht verloren gehen.

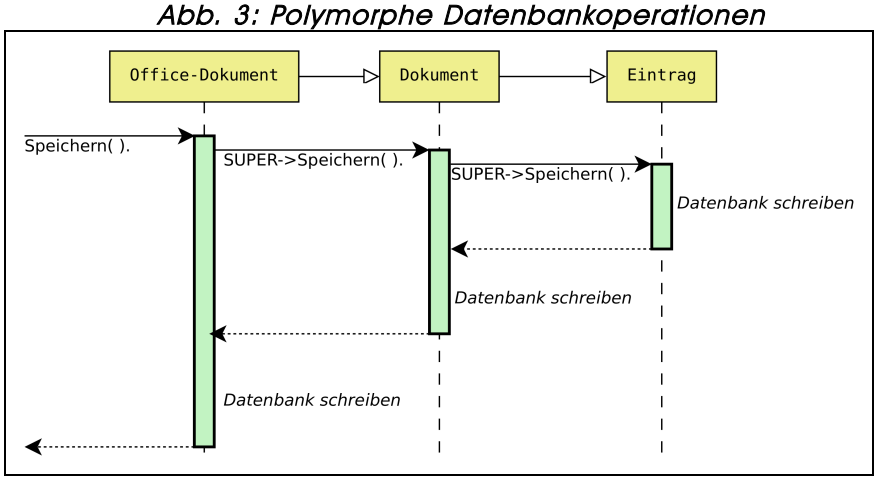
Vom ABAP-Laufzeitsystem wird dies insofern berücksichtigt, als dass jedes Objekt ein automatisches Attribut namens *super* besitzt, welches eine Referenz auf den geerbten Teil des Objekts darstellt. Sein Typ entspricht deshalb immer der Superklasse des Objekts und alle damit durchgeführten Aufrufe beziehen sich stets auf die Implementierung der Superklasse, unabhängig davon, ob eine polymorphe Überdefinition vorliegt. (Vgl. [WOLF07] S. 36/58).

Mit Blick auf die Datenbankmethoden ergibt sich also, dass jede Methode, bevor sie ihre eigenen Operationen durchführt, via *super* ihre namensgleiche Umsetzung der Mutterklasse ausführen muss. Denn auf diese Weise wird gewährleistet, dass auf die Datenbank in umgekehrter Aufrufreihenfolge stets von der allgemeinen zur speziellen Klasse hin zugegriffen wird, ohne dass eine Ebene der Vererbungshierarchie übersprungen wird. Ebenso wird sichergestellt, das die zwischen den korrelierenden Tabellen bestehenden Fremdschlüsselbeziehungen, welche notwendig sind, um die Vererbungsmimik auf das relationale Datenbank-

6 Dieses Vorgehen wird Überdefinition genannt und hat die Polymorphie der geerbten Methoden zur Folge. Sowohl nach Aussage von [BÖHM93] (S. 395-396) als auch von [WOLF07] (S. 36/58) stellt es eines der Kernkonzepte der Objektorientierung dar, um die Wiederverwendbarkeit von Quellcode zu erhöhen.

schema zu übertragen, nicht verletzt werden, da davon ausgegangen werden kann, dass sich diese analog zum objektorientierten Modell von der speziellen zur allgemeinen Tabelle hin orientieren.

Abb. 3: Polymorphe Datenbankoperationen



Quelle: Eigene Darstellung

Die obige Abbildung visualisiert das Verfahren für den Fall dreier Hierarchieebenen, weshalb im oberen Teil der Graphik die Klassennamen mitsamt ihren Erbschaftsbeziehungen eingezeichnet sind. Der untere Teil zeigt den zeitlichen Verlauf aller Methodenaufrufe und Datenbankzugriffe, der sich aus dem externen Aufruf der Speichern-Methode eines Objekts ergibt. Entsprechend des beschriebenen Mechanismus ruft jede Methode erst die Implementierung der eigenen Oberklasse auf, bevor sie selbst auf die Datenbank zugreift, was dazu führt, dass von der allgemeinen Eintragsklasse aus beginnend, die Datenbank in umgekehrter Reihenfolge beschrieben wird.

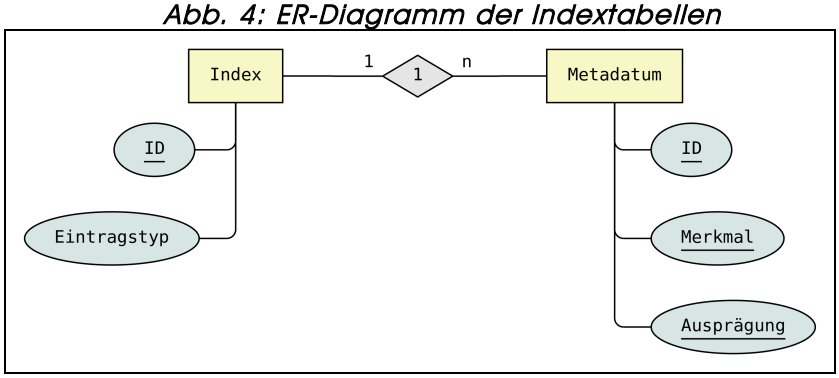
2.2.3 Verwaltung und Indizierung aller Akteninhalte

Ein objektorientiertes Datenmodell, das regen Gebrauch von Vererbungsmechanismen macht, bietet der Kundenakte zwar die nötige Flexibilität, um verschiedene Eintragsarten unter einem ganzheitlichen Konzept zu verwalten, es impliziert aber auch, dass für die Ablage der Eintragsdaten mehrere Tabellen zum Einsatz kommen, deren Datensätze nur in Kombination einen vollständigen Eintrag ausmachen. Glücklicherweise stellt das zuvor beschriebene, polymorphe Zugriffsverfahren sicher, dass für alle Datenbankoperationen die richtigen Tabellen angesprochen werden, doch ein Gesichtspunkt wird dabei außer Acht gelassen. Nämlich die Frage, anhand welchen Kriteriums beim Auslesen eines bestehenden Akteneintrags determiniert werden kann, welche Klasse letztlich instantiiert werden muss, um den Eintrag zu repräsentieren. Denn nur wenn die zu einem Eintrag adäquate Klasse einge-

setzt wird, kann die polymorphe Abbildung zwischen dem relationalen Datenbankschema und seiner objektorientierten Verschalung fehlerfrei funktionieren.

Naturgemäß kann diese Aufgabe nicht von den repräsentierenden Klassen während ihrer Instanzerzeugung beantwortet werden, da zu diesem Zeitpunkt die Auswahl der verwendeten Klasse bereits getroffen wurde. Die Entscheidung muss also ausgelagert werden, in dem Sinne dass eine speziell für die Objekterzeugung zuständige Einheit – ein sogenannter Erbauer[7] – verwendet wird, welcher als Blackbox funktioniert und immer das richtige Objekt generiert. Somit wird dafür Sorge getragen, dass die Entscheidung über das zu erzeugende Objekt nur von einer zentralen Implementierung getroffen wird, doch das Entscheidungsproblem wird dadurch noch nicht gelöst. Endgültige Klarheit kann darum nur die Datenbank selbst schaffen, indem sie eine Indextabelle vorhält, welche einerseits verwendet wird, um jeden Eintrag mit einer eindeutigen Kennnummer zu versehen und welche andererseits den Typus eines jeden Eintrags speichert.

Abb. 4: ER-Diagramm der Indextabellen



Quelle: Eigene Darstellung

Doch an dieser Stelle endet die Indizierung nicht, denn eine weitere Facette ihrer stellt die Suche nach den abgelegten Einträgen anhand übergreifender Kriterien dar, welche unabhängig von der Eintragsart auf alle Akteneinträge zutreffen können. Insbesondere handelt es sich hierbei um Kriterien, welche über den reinen Informationsgehalt der Einträge hinausgehen, diesem aber erst seine betriebswirtschaftliche Bedeutung verleihen, wie zum Beispiel die Verknüpfung zu einem Geschäftspartner oder zu einer zusammenhängenden Kette von Ereignissen. Derartige Informationen können auf vielfältige Weise in der Datenbank abgelegt werden, zum Beispiel in Form von dedizierten Verknüpfungstabellen mit ausgeprägten Fremdschlüsselbeziehungen. Allerdings müssen dabei die Anforderungen einer zentralen Suchfunktion berücksichtigt werden. Denn diese muss in erster Linie unkompliziert

7 Vgl. die Beschreibung des Erbauer-Patterns aus [GAMM04] (S. 119-123)

und performant sein, sie muss aber auch flexibel genug sein, um die Formulierung aussagekräftiger Suchanfragen zu gestatten, welche in der Lage sind, bereits die ersten Treffermengen hinreichend präzise einschränken.

Aus diesem Grund wird anstelle der Verwendung von heterogenen Verknüpfungstabellen angeregt, mithilfe von Metamerkmalen eine Art Schlagwortverzeichnis aufzubauen, das die Möglichkeit bietet, zu jedem Akteneintrag mehrere Metamerkmale zu hinterlegen, die durch eine Wertausprägung näher spezifiziert werden können. Ihre Ablage kann in einer singulären Tabelle erfolgen, welche je Eintrag in der Kundenakte und je zutreffendem Metamerkmal einen Datensatz vorsieht. Somit gestaltet sich die Abfrage flexibel genug, um verschiedene Aspekte in eine Suchanfrage einfließen zu lassen, es entsteht aber auch kein Laufzeitengpass, da zur Bildung der Trefferliste keine Mengenoperationen benötigt werden, um verschiedene Teilmengen aus unterschiedlichen Tabellen zu harmonisieren. (...)

2.3 Dokumentablage mithilfe des Business Document Service

(...) Die Besonderheit des Business Document Service besteht (...) darin, dass er aus Sicht einer ihn verwendenden Anwendung völlig in sich abgeschlossen ist und nur über die Zugriffsklassen *cl_bds_document_set* und *document_proxy* angesprochen wird.[8] Wenngleich also die inneren Vorgänge des Business Document Service sehr komplex sein mögen, ist weder seine interne Implementierung noch sein Customizing von Belang für die Kundenakte, da diese keine Auswirkungen auf seine programmatische Bedienung haben.

Dabei werden diverse Aufgabenfelder durch ihn abgedeckt. Zum einen stellt er eine zentrale Verwaltung aller betriebswirtschaftlichen Dokumente dar, ohne dass diese in der operativen Datenbank des SAP-Systems enthalten sein müssen. Viel mehr erfolgt die physische Dokumentablage mit dem ArchiveLink, welcher neben der SAP-Datenbank auch auf externe Content Server und Archivsysteme zugreifen kann. Zum anderen bietet er die Möglichkeit, von allen Dokumenten mehrere, konkurierende Versionen und Varianten zu verwalten. Und letztlich stellt er eine Verknüpfung her zwischen den Dokumenten und den betriebswirtschaftlichen Objekten, zu welchen die Dokumente gehören.

Vor SAP-Release 4.7 bezog sich diese Verknüpfung immer auf ein Business Objekt, seit Release 4.7 können jedoch auch ABAP-Klassen herangezogen werden. In Folge dessen werden exakt drei Werte benötigt, um eine gültige Verknüpfung herstellen zu können:

8 Für ein ausführliches Beispiel zu deren Verwendung siehe [WOLF07] S. 407-428.

- **Der Objekttyp** gibt an, ob sich die Verknüpfung auf ein Business Objekt, eine ABAP-Klasse oder auf ein nicht spezifiziertes Objekt bezieht.
- **Der Objektname** gibt den Namen des Business Objekt oder der ABAP-Klasse an.
- **Der Objektschlüssel** stellt eine einheitliche Kennnummer dar, welche die Objektinstanz eindeutig kennzeichnet.

Für die Kundenakte bietet sich hier an, die Mutterklasse aller Einträge als Verknüpfungsziel anzugeben, wobei der Objektschlüssel in diesem Falle die in der Kundenakte verwendete Kennnummer des Eintrags sein sollte. Dementsprechend muss die Kundenakte keinen Gebrauch von der Versionsverwaltung des Business Document Service machen, weil aufgrund der geforderten Vollständigkeit der Einträge und dem Schutz vor Verfälschungen keine Eintragsversionen verwaltet werden. Neue Versionen eines Sachverhalts sollten immer zu einem neuen Eintrag mit eigener Kennnummer führen.

Eine Eigenheit des Business Document Service besteht in der Verwendung von sogenannten Komponenten und Signaturen. Als Komponenten werden dabei alle Teile eines Dokuments bezeichnet, da der Business Document Service die Verwaltung von mehrteiligen Dokumenten unterstützt. Mithin besteht jedes Dokument aus mindestens einer Komponente, wobei zu jeder Komponente ein Satz von Signaturen gehört, welche die Komponente näher beschreiben. In diesem Sinne sind die Signaturen vergleichbar mit den angeregten Metamerkmalen der Kundenakte und auch ihre Funktionsweise gestaltet sich analog.

Abb. 5: Signaturen zur Beschreibung von Komponenten

Signatur	Bedeutung
BDS_DOCUMENT_CLASS	Dokumentart, Fachliche Beschreibung, Pflichtwert
BDS_DOCUMENT_TYPE	Dokumenttyp, Technische Beschreibung, Pflichtwert
BDS_KEYWORD	Schlagwörter. Können beliebig oft verwendet werden.
DESCRIPTION	Beschreibung
LANGUAGE	Sprachschlüssel
CREATED_BY	Benutzernamen des Erstellers
CREATED_AT	Zeitstempel der Erzeugung
LAST_CHANGED_BY	Benutzernamen des letzten Änderers
LAST_CHANGED_AT	Zeitstempel der letzten Änderung

Quelle: Eigene Darstellung

Dabei wird anhand der Signaturen eine weitere Besonderheit des Business Document Service evident, welche von der Kundenakte berücksichtigt werden muss: Die Unterscheidung zwischen der technischen Beschreibung eines Dokuments und seiner fachlichen Bedeutung. Laut [WOLF07] (S. 408-410) stellt dies ein zweistufiges Definitionskonzept dar, wobei die technische Beschreibung allein auf den MIME-Type[9] des Dokuments abzielt. Die Dokumentart hingegen ist an den technischen Dokumenttyp gebunden, beschreibt diesen aber aus betriebswirtschaftlicher Sicht. Leider ist die Beziehung zwischen dem Dokumenttyp und der Dokumentart eine 1:n-Beziehung, so dass hinter einer Dokumentart nicht mehrere technische Typen stehen können, womit die Definition einer einheitlichen Dokumentart für die Einträge der Kundenakte verhindert wird. Da das Standard-Customizing des ArchiveLink allerdings viele Standardarten für die gängigen Office- und Dokumentdateien kennt, kann auf diese bedenkenlos zurückgegriffen werden.

9 Der MIME-Type ist eine standardisierte Identifikation des Dateiformats. Seine Ursprünge liegen in den frühen Nachrichtendiensten des Internets und auch heute spielt er über dieses Umfeld hinaus eine bedeutende Rolle.

3 Integration der SAP-WebFlow-Engine

Dieses Kapitel bildet den Abschluss der vorliegenden Diplomarbeit. Es wird darin untersucht, inwiefern die WebFlow-Engine des SAP-Web Application Server als Grundlage für die Umsetzung mehrstufiger Entscheidungsprozesse mit Integration der Kundenakte herangezogen werden kann. Dabei wird besonders auf die technische Integration der drei Komponenten AddOns, Kundenakte und WebFlow-Engine eingegangen.

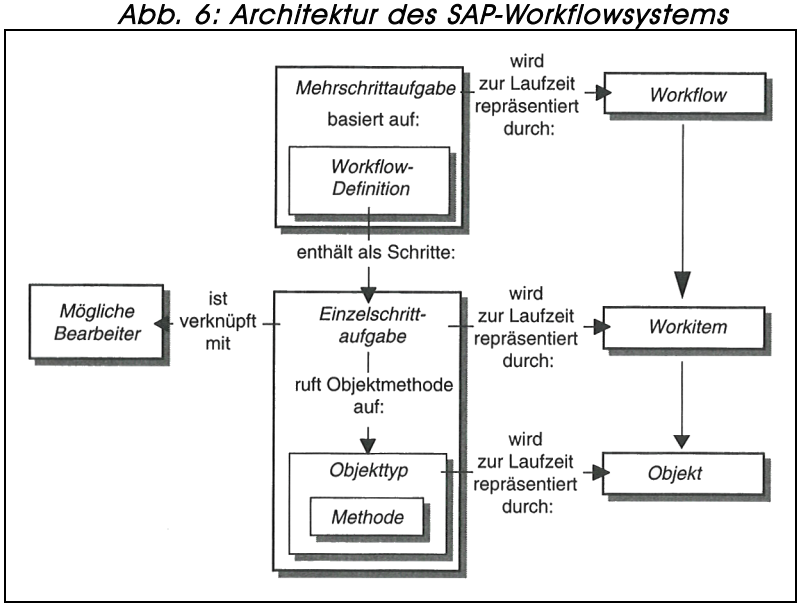
3.1 Beschreibung des SAP-Workflowsystems

Das Workflowsystem von SAP basiert auf der WebFlow-Engine, welche als direkter Nachfolger der Business Workflow Engine zum Lieferumfang von SAP ERP gehört. Sie zeichnet sich laut [RICK02] (S. 45-51) dadurch aus, dass sie direkt in den Web Application Server integriert ist und somit auf viele Funktionalitäten dessen zurückgreift, ohne sie zu duplizieren. Zu diesen gehören die technischen Kommunikationsmöglichkeiten und Sicherheitsfunktionen, das Organisationsmanagement aus dem Modul Human Resources sowie die Möglichkeit, nahezu jede SAP-Transaktion ansteuern zu können.

Die Workflows, die sie ausführt, werden von [BRAH03] (S. 29-30) als vorzeitig definierte Prozesse beschrieben, welche die Arbeit mehrerer Beteiligter an betriebswirtschaftlichen Objekten wie Bestellanforderungen, Rechnungen oder Materialstämmen koordinierten. Ihr klassisches Anwendungsgebiet sind laut [RICK02] (S. 32) mehrstufige Entscheidungsprozesse, doch im Grunde genommen lassen sich jegliche Abläufe mit ihnen automatisieren, unabhängig davon, in welchem Grade Endanwender daran beteiligt sind und unabhängig davon, was das Ziel der Prozesse ist. Dementsprechend beschreibt [ANGE03] (S. 142) Workflows schlicht als Verkettung einzelner Transaktionen auf der Basis von auslösenden Ereignissen, wobei konditionale Handlungsabläufe und die Weitergabe von Daten zwischen den Verarbeitungsschritten unterstützt werden und jederzeit angehalten werden kann, um eine Benutzeraktion zu erfragen.

Wie das Workflowsystem funktioniert, zeigt [MEND00] auf den Seiten 21-24, denen auch die nachfolgende Zeichnung entnommen ist. Seine Beschreibung deckt sich mit den Meinungen von [BRAH03] (S. 79-80) und [RICK02] (S. 51-53), welche das Thema unter anderen Gesichtspunkten beleuchten:

Abb. 6: Architektur des SAP-Workflowsystems



Quelle: Aus [MEND00] (S. 21)

Grundsätzlich muss zwischen der Definitionszeit eines Workflows und seiner Laufzeit unterschieden werden. Die Definitionszeit entspricht seiner Entwicklung, während der er durch die Workflowdefinition beschrieben wird, die Laufzeit hingegen entspricht seiner Ausführung, während der er durch eine Workflowinstanz beschrieben wird. Von dieser kann es im Gegensatz zu seiner Definition beliebig viele geben. Die Workflowdefinition wird auch Mehrschrittaufgabe genannt, denn sie beschreibt einen vollständigen Geschäftsprozess und besteht aus der sachlogischen und zeitlichen Anordnung seiner Aktionen und strukturgebenden Elemente. Bei diesen handelt es sich entweder um Einzelschrittaufgaben[10], welche zur Ausführung von Objektmethoden führen, oder um Elemente der Ablaufsteuerung, wie Bedingungen, Schleifen oder Benutzerentscheidungen.

Alle Einzelschrittaufgaben werden zur Laufzeit durch Workitems repräsentiert, welche, sofern sie das Einschreiten eines Anwenders erfordern, den zuständigen Sachbearbeitern zugestellt werden. Die Ermittlung dieser erfolgt mit Zuständigkeitsregeln, die erst während der Laufzeit ausgewertet werden. Da diese unabhängig von der Workflowdefinition sind, gestatten sie eine schnelle Reaktion auf sich ändernde Zuständigkeiten, ohne dass Änderungen an den Geschäftsprozessen notwendig sind. Somit bilden die Einzelschrittaufgaben den Schnittpunkt zwischen den funktionalen Aspekten eines Workflows und seinen organisatorischen Aspekten, indem sie determinieren, welche Aktionen von welchen Akteuren vorgenommen

10 Der Begriff wird in der Literatur etwas unscharf verwendet. Tatsächlich werden in der Workflowdefinition Aktivitäten eingesetzt, welche der Literaturbeschreibung der Einzelaufgaben entsprechen. Sie kapseln den Aufruf einer Objektmethode jedoch nicht direkt, sondern führen eine separat definierte Aufgabe aus, welche neben dem Methodenaufruf noch die erklärenden Texte der Workitems beinhaltet.

werden, ohne jedoch eine Aussage darüber zu treffen, wie die Aktionen durchgeführt werden. Dies überlassen sie den von ihnen aufgerufenen Objektmethoden, welche in Form von Programmcode vorliegen.

Die Menge der offenen Workitems bildet die Worklist der Sachbearbeiter und entspricht somit inhaltlich den bisherigen Arbeitsvorräten der CREFOsprint-AddOns. Entsprechend stellt nach Aussage von [MEND00] (S. 29-37) ein Workitem immer eine konkrete Aufgabe dar und wird durch das zu bearbeitende, betriebswirtschaftliche Objekt sowie die Menge der möglichen Sachbearbeiter näher spezifiziert. Es wird allen zuständigen Sachbearbeitern zugestellt, von denen es aber nur einer in Bearbeitung nehmen kann. Dabei unterliegt es den Stati

- *Bereit*,
- *Angenommen*,
- *In Arbeit* und
- *Beendet*.

Darüber hinaus kann es von den Anwendern veranlasste AdHoc-Anhänge wie Notizen oder Dokumentanlagen besitzen und ähnlich einer E-Mail verschickt werden. Die Nähe zur E-Mail wird gar noch deutlicher anhand der Tatsache, dass es eine Kurzbeschreibung und eine Aufgabenbeschreibung als Langtext beinhaltet. Doch im Gegensatz zur E-Mail besitzt es keine Sendezeit, sondern mehre Start- und Endtermine, welche nach gewünschten, realen und spätesten Terminen unterschieden werden. (...)

4 Schlussbetrachtung

4.1 Quintessenzen der Untersuchung

Aufbauend auf den konstitutiven Prozessen des SAP-Forderungsmanagements entwickelt und vertreibt die cormeta ag mehrere AddOns, welche die Module FI und SD um diejenigen Funktionalitäten ergänzen, welche ihnen zu einer vollständigen Abdeckung des Kreditmanagementprozesses nach [BROE00] (S. 9-11) fehlen. Dabei konzentrieren sich die AddOns de facto auf die operative Ausübung des Kreditmanagements, ohne eine zusammenhängende Dokumentation aller durchgeführten Aktionen und Entscheidungen vorzunehmen. Zwar werden die Arbeitsvorräte der CREFOsprint-AddOns fortlaufend mit Meldungen über die stattfindenden Handlungen versorgt, doch greift ihr Konzept an mehreren Stellen zu kurz:

Zum einen beschränken sie sich auf jeweils nur ein AddOn der CREFOsprint-Familie, so dass eine AddOn-übergreifende, den gesamten Kreditmanagementprozess umspannende, Meldungsbearbeitung fehlt. Zum anderen liegt ihr Schwerpunkt weder auf der allumfassenden Dokumentation aller Geschäftsvorfälle, noch sind ihre Inhalte für historische Auswertungen vorgesehen. Darüber hinaus besteht als einzige Bearbeitungsmöglichkeit ihrer Einträge der Absprung in eine themenverwandte Transaktion, ohne dass die Ergebnisse der Bearbeitung dem auslösenden Ereignis zugeordnet dokumentiert werden. Insgesamt kann darum gesagt werden, dass sie den Ansprüchen der revisionssicheren Archivierung wie sie der *Verband Organisationssystem und Informationssysteme* hauptsächlich aus den Regelungen des Handelsgesetzbuches und der Abgabenordnung ableitet, nicht genügen. (Vgl. Insbesondere [KAMP03] S. 24-26)

Da aber das Vorhandensein von Informationen Prämisse für fundierte Entscheidungen ist, kommt dem Wissensmanagement eine besondere Relevanz zu. Mit Blick auf die AddOns bedeutet dies, dass in Form der Kundenakte ein Informationspool etabliert werden muss, welcher sich aus allen in den AddOns anfallenden Informationen und Erkenntnissen zusammensetzt. Dabei sollte auch der innerbetriebliche Informationsfluss gefördert werden, indem entsprechend der Empfehlung von [BRAH03] (S. 159-164) Informationsworkflows eingesetzt werden, um neue Akteneinträge unternehmensweit zu kommunizieren.

Lässt man dabei die Möglichkeit zu, dass Workflows über den informativen Charakter hinaus komplexe Geschäftsprozesse mit mehreren Akteuren koordinieren können (vgl. z.B. [BRAH03] S. 29-30), stellt sich die Frage nach dem verbleibenden Zweck der Arbeitsvorräte, insbesondere da ihre Funktionalitäten vom SAP-Standard nicht nur abgedeckt sondern in punkto Meldungsbearbeitung weit übertroffen werden. Allerdings muss dabei die a posteriori verständliche Dokumentation aller Workflowschritte berücksichtigt werden, in dem Sinne, dass erkennbar sein muss, welche Schritte als Reaktion auf einen Eintrag durchgeführt wurden. Hierfür stehen mehrere Lösungswege zur Verfügung: Einerseits kann die Kundenakte zur Eintragserzeugung direkt aus einer Workflowinstanz aufgerufen werden. Andererseits muss sichergestellt werden, dass die durch laufende Workflows entstehenden Mitteilungen der AddOns in ihrem Entstehungskontext festgehalten werden.

Ungeachtet der konzeptionellen Unzulänglichkeiten der Arbeitsvorräte kann ihre technische Realisierung als Grobvorlage für die Umsetzung einer Kundenakte betrachtet werden. So wird für die Kundenakte ebenfalls eine normierte Eingangsschnittstelle benötigt, welche den AddOns die einfache Erzeugung neuer Einträge ermöglicht, obgleich hierfür auf die SAP-Ereignissteuerung[11] anstelle von Funktionsbausteinen zurückgegriffen werden sollte, um latente Abhängigkeitsprobleme zu umgehen. Ebenfalls muss die Möglichkeit der sachlichen Aggregation mehrerer Einträge zu zusammenhängenden Sequenzen übernommen werden. Um dabei auch komplexe Kausalitätsketten zu präservieren, ist die rekursive Schachtelung der Sequenzen allerdings unabdingbar – eine Tatsache, die von der vorgeschlagenen Workflowintegration auch zur Vereinfachung der Implementierung ausgenutzt wird.

Abweichend vom Datenmodell der Arbeitsvorräte wird, nicht zuletzt um die Verfahrenssicherheit der Kundenakte zu gewährleisten, nahe gelegt, das relationale Datenbankschema der Kundenakte in einem objektorientierten Modell zu verschalen, das die Abbildung von Erbschaftsbeziehungen und die Integration von Geschäftslogik ermöglicht, da auf diese Weise die Anbindung konzeptionell divergierender Ablagesysteme für die unterschiedlichen Eintragsarten vereinfacht wird. In diesem Zusammenhang sei insbesondere auf die Ablage von originären AddOn-Daten auf der Datenbank und von Dokumenten mit dem Business Document Service verwiesen. Damit dem Anwender jedoch eine von diesen technischen Feinheiten indifferente Suchfunktion angeboten werden kann, muss eine auf Metadaten basierende, von der Datenablage abgehobene Indexdatenbank geschaffen werden, welche der Katalogisierung aller Akteneinträge dient. (Vgl. [KAMP03] S. 27). Dabei muss allerdings

11 Vgl. [TANE03] S. 122-127 sowie S. 132-135 in Verbindung mit [MEND00] S. 225

das Berechtigungskonzept berücksichtigt werden, so dass die Menge der abrufbaren Daten auf ein erforderliches Minimum beschränkt werden kann.

Selbstverständlich sind an den AddOns Anpassungen notwendig, um die von ihnen erzeugten Informationen in der Kundenakte zu protokollieren. Allerdings sind die AddOns in ihrer gegenwärtigen Gestalt nur bedingt für die Kooperation mit dem SAP-Workflowsystem vorbereitet, was am durchweg objektorientierten Ansatz der WebFlow-Engine liegt. Um die Kompatibilität zum Workflowsystem herzustellen, ist es für die AddOns unverzichtbar, ihre Kernfunktionen in Business Objekten zu verpacken, da nur diese von der WebFlow-Engine angesprochen werden können.

4.2 Resumé

Obwohl die Untersuchung von Umsetzungsmöglichkeiten einen signifikanten Teil der vorliegenden Diplomarbeit ausmacht, können in ihrem Rahmen nur punktuelle Vorschläge für die Realisierung der Kundenakte unterbreitet werden. Folglich ist für die tatsächliche Entwicklung ein eigenständiges Projekt erforderlich, in welchem die vorgezeichneten Realisierungswege vervollständigt und vollkommen zu ende modelliert werden. Dies hat zur Folge, dass die Umsetzung der Kundenakte mit einer großen, zeitlichen Investition verbunden ist, sowohl um die notwendigen Anpassungen an den AddOns durchzuführen, als auch um die Kundenakte selbst zu realisieren. Dem steht allerdings der Mehrwert gegenüber, den die Kundenakte den AddOns hinzufügt:

Durch den konsequenten Einsatz der SAP-WebFlow-Engine im Zusammenhang mit der Kundenakte können die bisher separat entwickelten Arbeitsvorräte durchgehend vom SAP-Standard abgelöst werden, ohne dass dadurch funktionale Einbusen entstehen. Stattdessen wird der Funktionsumfang der AddOns erweitert, indem ihre Einbettung in systemgestützte Geschäftsprozesse und mehrstufige Entscheidungsvorgänge ermöglicht wird. Darüber hinaus wird die revisionssichere Aufzeichnung aller in diesem Zusammenhang entstehenden Informationen gewährleistet und somit die Pflege einer zentralen Wissensbasis unterstützt.

Ein weiterer Effekt, der sich ergibt, ist, dass die Entwicklung der AddOns fortan stärker auf ihre Kernfunktionalitäten ausgerichtet werden kann, während die ausgelagerten Funktionalitäten mit jeder Weiterentwicklung der Kundenakte allen AddOns einheitlich zur Verfügung stehen.

A Quellenangaben

ADAM96: Dietrich Adam (1996), Planung und Entscheidung: Modelle - Ziele - Methoden, 4. Auflage, Gabler Verlag, Wiesbaden

ANGE03: Axel Angeli (2003), Technische Integration von SAP-Systemen, 1. Auflage, SAP PRESS / Galileo Press GmbH, Bonn

AO: Autor unbekannt (2007), Wichtige Steuergesetze (Abgabenordnung), 55. Auflage, Neue Wirtschafts-Briefe GmbH & Co. KG, Herne

BÖHM93: Rolf Böhm et al. (1993), System-Entwicklung in der Wirtschafts-Informatik, 2. überarbeitete und erweiterte Auflage, Verlag der Fachvereine an den schweizerischen Hochschulen und Techniken AG, Zürich

BRAH03: Markus Brahm et al. (2003), Workflow Management mit SAP WebFlow, 1. Auflage, Springer-Verlag, Berlin

BROE00: Herbert Broens (2000), Organisation des Kreditmanagements, 1. Auflage, Verband der Chemischen Industrie e.V., Frankfurt

GAMM04: Erich Gamma et al. (2004), Entwurfsmuster: Elemente wiederverwendbarer objektorientierter Software, 1. Auflage, Addison-Wesley Verlag, München

GoBS: Bundesfinanzministerium (1995), Grundsätze ordnungsgemäßer DV-gestützter Buchführungssysteme (GoBS), Berlin, http://www.bundesfinanzministerium.de/nn_314/DE/BMF__Startseite/Service/Downloads/Abt__IV/BMF__Schreiben/015,templateId=raw,property=publicationFile.pdf, Abruf am 22.04.2008 um 13:54:01, Beigefügt unter dem Dateinamen GoBS.pdf

HGB: Autor unbekannt (2006), Handelsgesetzbuch, 44. Auflage, Deutscher Taschenbuch Verlag GmbH & Co. KG, München

KAMP03: Dr. Ulrich Kampffmeyer (2003), Revisionssichere Archivierung und Dokumentenmanagement im Licht neuer rechtlicher Anforderungen, Hamburg, http://www.project-consult.net/Files/audicon_GDPdU_052003.pdf, Abruf am 22.04.2008 um 13:49:51, Beigefügt unter dem Dateinamen KAMP03.pdf

KOKA00: Ljuba Kokalj et al. (2000), Zahlungsverzug und Forderungsmanagement in mittelständischen Unternehmen, 1. Auflage, Deutscher Universitäts-Verlag GmbH, Wiesbaden

MEND00: Ulrich Mende et al. (2000), SAP Business Workflow: Konzept, Entwicklung, Anwendung, 2. aktualisierte Auflage, Addison-Wesley Verlag, München

RICK02: Alan Rickayzen et al. (2002), Workflow-Management mit SAP: Effektive Geschäftsprozesse mit SAPs WebFlow Engine, 1. Nachdruck 2003, SAP PRESS / Galileo Press GmbH, Bonn

TANE03: Andrew Tanenbaum et al. (2003), Verteilte Systeme: Grundlagen und Paradigmen, 1. Auflage, Pearson Studium / Prentice Hall, München

WOLF07: Frank Wolf (2007), ABAP Objects, 1. Auflage, dpunkt.verlag GmbH, Heidelberg