

# Electronic Contracting im Internet<sup>1</sup>

M. Merz<sup>+</sup>, F. Griffel<sup>\*</sup>, M. Boger<sup>\*</sup>, H. Weinreich<sup>\*</sup>, W. Lamersdorf<sup>\*</sup>

+ Ponton Hamburg, merz@ponton-hamburg.de

\* VSYS - Verteilte Systeme, Fachbereich Informatik, Universität Hamburg,

{merz | griffel | boger | lweinrei | smueller | lamersd}@informatik.uni-hamburg.de

## Abstract

Heute bietet das Internet nicht nur für Großunternehmen sondern auch für kleine Unternehmen Möglichkeiten zur automatisierten Kooperation. Im Bereich des Business-to-Business-Commerce wurden jedoch bisher kleine Unternehmen aufgrund relativ hoher Transaktionskosten eher davon abgehalten, Handelstransaktionen über das Internet durchzuführen. Für solche Anwendungsbereiche wird in diesem Beitrag das COSMOS-Projekt (Common Open Service Market fOr SMEs) und seine Architektur vorgestellt, mit deren Hilfe die Vermittlung von Transaktionspartnern, der Aushandlungsprozeß eines Vertrages sowie das Unterzeichnen und schließlich die Abwicklung der vertraglich vereinbarten Leistungen durch eine verteilte Kollaborationsanwendung unterstützt wird. Dabei basiert die COSMOS-Architektur auf einem einheitlichen Vertragsmodell und unterstützt diese Prozesse in integrierter Form. Architekturell erfolgt der Systementwurf in Anlehnung an den CORBA Business Object Component Architecture (BOCA).

## 1 Einführung

Jede kommerzielle Handelstransaktion, die über organisatorische Grenzen hinweg abgewickelt wird, führt direkt oder indirekt zu einem Vertrag, der zwischen den beteiligten Parteien geschlossen wird. Dieser Vertrag hält deren Verpflichtungen, die dort definierten Leistungen zu erfüllen, fest. Gleichzeitig definiert er die jeweiligen Rechte, die den anderen Parteien daraus erwachsen. Diese austauschbaren Leistungen sind dabei Dienste, Güter, Rechte oder Zahlungen.

In der Literatur werden für eine derartige Transaktion drei Phasen unterschieden [Schm95]:

- In der *Informationsphase* beobachten Teilnehmer den Markt und unterbreiten möglichen Partnern Angebote. Während der Produktsuche werden dabei Produktspezifikationen - also Preise, Qualitätsmerkmale, etc. - zur Evaluation herangezogen.

---

<sup>1</sup> Das in diesem Beitrag präsentierte Projekt „COSMOS“ wird im Rahmen des Vertrages Nr. 26850 (Esprit) von der Europäischen Kommission gefördert

- In der *Verhandlungsphase* treten potentielle Partner zunächst in Verbindung, um über Angebote und Gegenangebote diese Spezifikationen iterativ anzupassen. Dieser Verhandlungsprozeß führt entweder zu einem Zustand der Einigung, oder er wird abgebrochen.
- Schließlich stellt die Vertragsunterzeichnung den Übergang zur *Abwicklungsphase* dar. Diese Phase kann zwischen wenigen Sekunden und mehreren Jahren dauern.

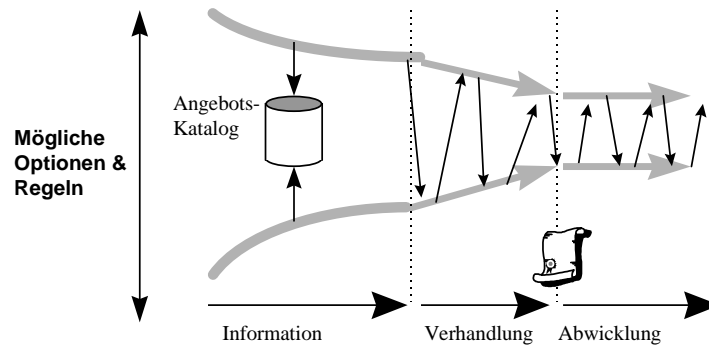


Abb. 1: Phasen einer Handelstransaktion

### Aufbau des Papiers

Der Rest dieses Beitrags ist wie folgt organisiert: Abschnitt 2 motiviert das Gebiet des elektronischen Vertragsdienstes und stellt seine Schwerpunkte dar. Anschließend wird die COSMOS-Architektur im Abschnitt 3 dargestellt. Abschnitt 4 befaßt sich mit dem COSMOS-Objektmodell für elektronische Verträge sowie der komponentenbasierten Modellierung ihrer Bestandteile. Im Abschnitt 5 werden aktuelle Schritte zur Implementierung des Systems erläutert. Weitere Literatur – auch zu hier nicht behandelten Fragestellungen – sind der COSMOS-Home-Page zu entnehmen [COSMOS99].

## 2 Electronic Contracting

Die Metapher des Vertrages ist besonders geeignet, die Verbindung zwischen dem Phasenmodell der ökonomischen Ebene und einer systemtechnischen Unterstützung über alle Phasen herzustellen: Er repräsentiert als zusammenhängendes Dokument die zuvor gesammelte Information, d.h., ausgehandelte Rechte und Pflichten sowie einen Plan zur Erfüllung der fixierten Leistungen. Insbesondere wird die gegenseitige Verpflichtung durch das Leisten der Unterschrift aller Parteien verbindlich und verifizierbar [HaBi93].

Diese Natur des herkömmlichen Vertrages erscheint geeignet, ihn nicht nur auf elektronischem Wege zu erstellen, sondern auch zur Steuerung des Leistungsaustausches heranzuziehen. In diesem Zusammenhang ist insbesondere auf die Arbeiten von Milo-

sevic et al. zu verweisen, in denen eine Architektur zur Verwaltung elektronischer Verträge als Einsatzszenario für das Open Distributed Processing (ODP) diskutiert wird [Milo95, MiAC96].

## 2.1 Reduzierung der Transaktionskosten

Unter realistischen Bedingungen entstehen durch die Informationsphase sowie den Prozeß der Vertragsaushandlung hohe Transaktionskosten. Üblicherweise treten diese Kosten in Form von Personalkosten für die Recherche, Ausarbeitung von Verträgen sowie die Einbindung Dritter auf. Die üblichen Kostenarten sind dabei:

- *Informationskosten.* Sie treten aufgrund mangelhafter Transparenz beim Verschaffen der erforderlichen Marktübersicht auf. Verlage, Analysten und andere Dienstleister profitieren aufgrund des unvollständigen Wissens der Marktteilnehmer von dieser Intransparenz.
- *Verhandlungskosten.* Eine Verhandlung kann ein langwieriger Prozeß sein, der nicht nur interne Personalressourcen bindet, sondern auch durch mögliche Mißverständnisse hinsichtlich der Vertragessemantik und seiner Handhabung sowie durch die Einbindung Dritter fehleranfällig ist. Schließlich erfordert der Prozeß des Signierens besondere Sicherheits- und Vertrauensmechanismen – insbesondere wenn mehr als zwei Parteien oder mehr als ein Vertrag involviert sind.
- *Abwicklungskosten.* Vertragsparteien könnten bei der Erbringung der vereinbarten Leistung in Verzug geraten, so daß zusätzliche Maßnahmen zur Überwachung der Leistungserfüllung erforderlich sind, die ihrerseits die Transaktionskosten steigern.

Als Situationsbeschreibung kann somit festgehalten werden, daß in der Praxis Transaktionskosten erheblich reduziert werden könnten, wenn ein Softwaresystem eine angemessenere Balance zwischen Automatisierung, Standardisierung, einem verbesserten Geschäftsmodell sowie der Regulierung juristischer Rahmenbedingungen bietet. Insbesondere bei der Nutzung des Internet ist daher eine verteilte Anwendung gefordert, die nicht nur einen einfachen technischen Zugang zum System schafft, sondern gleichzeitig auch organisatorischen und juristischen Anforderungen der Marktteilnehmer gerecht wird.

## 2.2 Schwerpunkte des Electronic Contracting

Diese Ausgewogenheit bedarf einer angemessenen Dienstinfrastruktur mit wohldefinierten Schnittstellen zwischen den beteiligten Komponenten sowie den zumeist schon bestehenden Softwaresystemen der beteiligten Parteien. Der bisher wohl prominenteste und älteste Ansatz zur Integration von Geschäftsprozessen zwischen Unternehmen ist der Elektronische Datenaustausch (EDI). Hier beschränkt sich die Standardisierung auf die Definition und den Austausch passiver Nachrichtenobjekte, so daß es Anwendern überlassen bleibt, diese Daten sinnvoll zu verarbeiten bzw. den richtigen Zeitpunkt für ihren Austausch zu wählen.

Somit integriert EDI nicht die beteiligten Unternehmen sondern macht sie lediglich interoperabel. Dieses wird durch ein hohes Maß an Standardisierung und Formalisierung erreicht, jedoch auf Kosten der Unterstützung neuer oder individueller Geschäftsprozesse, für die jeweils spezialisierte EDI-Nachrichtenformate definiert werden müssen. Die Nutzung von EDI ist damit für kleine und mittlere Unternehmen unrentabel.

Andererseits bieten heutige Internet-Technologien weitreichende Möglichkeiten zur Interoperation, jedoch mangelt es hier wiederum häufig an der nötigen Standardisierung, die zur effizienten Kooperation zwischen Unternehmen erforderlich ist.

Während sich nun EDI mit der Datendefinition auf die Festlegung von Nachrichtentypen befaßt, wird beim COSMOS-Projekt das einfache und integrierte Bearbeiten, Signieren und Ausführen von Verträgen unterstützt. Die Zielsetzung besteht darin, Einzelpersonen und kleinen Unternehmen mit wenig Aufwand ein verteiltes System bereitzustellen, welches als begleitendes Instrument zur Abwicklung von Handelstransaktionen eingesetzt werden kann. Ein besonderes Schwergewicht wird dabei auf die Verhandlungsunterstützung und Ausführung gelegt, indem der Vertrag selbst aktiv diese beiden Prozesse beeinflussen kann. Durch die integrierte, halbautomatische Komposition eines Vertrages ist das COSMOS-System in der Lage, in konsistenter Weise Information zur Ausführung der vereinbarten Leistungen im Sinne einer Workflow-Definition einzuschließen. Diese Information hilft, den Vertrag in der Abwicklungsphase zur Ausführung zu bringen.

### **3 Technologischer Hintergrund**

Die COSMOS-Architektur zielt darauf ab, Transaktionskosten sowohl durch *Automatisierung* als auch durch *Integration* zu reduzieren. Dies betrifft vor allem folgende Softwarekomponenten (vgl. Abb. 2):

#### **3.1 Softwarekomponenten der COSMOS-Architektur**

- *Online-Kataloge*. Informationen über Marktteilnehmer können heute über Suchmaschinen oder „Gelbe-Seiten“-Dienste erlangt werden. Beiden Ansätze mangelt es jedoch an Selektivität auf der Basis einer geeigneten Menge von Qualitätsattributen. Diese würden Anbietern die Möglichkeit gewähren, nicht nur Teilnehmerinformation, sondern auch Angebote zu registrieren. Nachfragern würden die Attributlisten als Hilfsmittel dienen, um präzisere Anfragen zu formulieren.
- *Broker* arbeiten im Auftrag eines Teilnehmers, um ein Konsortium aus potentiellen Vertragspartnern zu formieren. Sie benötigen sowohl Zugang zum Online-Katalog als auch zu den QoS-Spezifikationen der Nachfrager. Richtlinien zur Auswahl und Navigationsschnittstellen unterstützen Anwender des Brokers in flexibler Form. Als Kernkomponente des Brokers dient ein Trader, wie er im Rahmen der ODP- bzw. CORBA-Standardisierung spezifiziert wurde [ISO97, MJML95, OMG96]. Im Gegensatz zum CORBA-Trader ist der Broker jedoch in der Lage, Angebote zu mehr als einer Spezifikation (der „Import“-Anfrage des Traders) zu liefern. Resultat des

Brokers ist dabei ein *Vertragsvorschlag*, welcher zu jeder vertraglich definierten Rolle eine Partei und für jede Leistungsspezifikation eine von der entsprechenden Partei angebotene Leistung umfaßt. Diese Informationen werden aus dem Angebotsraum des Katalogs ermittelt.

- *Verhandlungsunterstützung*. Eine Verhandlung wird als das gemeinsame Editieren eines Vertrags als strukturiertes Dokumentes aufgefaßt. Jede Modifikation dieses Vertrages wird im Verhandlungskontext von der jeweils empfangenden Partei als ein Vertragsangebot aufgefaßt, welches wiederum als Gegenangebot oder Ablehnung zurückgesendet werden kann. Die Vertragsverhandlung kann unabhängig vom gemeinsamen Editieren durch ein Konferenzsystem - wie z.B. ein Telefon- oder Videokonferenzsystem - flankiert werden. Die Aufgabe der Verhandlungsunterstützung ist dabei erstens eine konsistente und integrierte Dokumentenbearbeitung, zweitens bietet sie die Möglichkeit, zusätzliche Dienste zur Steuerung der Verhandlung hinzuschalten [CePW98]. Dieses ist zum einen ein Workflow-System zur Verhandlungssteuerung, welches auf der Basis flexibler, gefärbter Petrinetze [Jens92] den Kommunikationsfluß zwischen den Parteien steuert, zum anderen bietet es für jeden Teilnehmer die Möglichkeit, ein individuelles Strategiemodul hinzuschalten, welches – im Rahmen der Definitionen von Verhandlungsrichtlinien – zur Durchführung der Verhandlung von einer Partei beauftragt werden kann [TGML98].
- *Unterstützung des Vertragsabschlusses*. Ein Dokument, welches kollaborativ editiert wurde, kann von den beteiligten Parteien gemeinsam elektronisch signiert werden [Schn95]. Bedingung ist hier die Existenz einer standardisierten externalisierten Repräsentation des Vertrages. Zu diesem Zweck im COSMOS-Projekt eine XML-Repräsentation für Verträge entwickelt worden [GoPr98, COSMOS98]. Wenn menschliche Teilnehmer an der Vertragsverhandlung beteiligt sind, ist es erforderlich die Funktionen einer natürlichen Unterschrift (Warnfunktion, Abschlußfunktion, Authentisierung, etc. [HaBi93]) auf das elektronische Medium zu übertragen. Daher folgt diese COSMOS-Komponente dem Motto „What you see is what you sign“: Eine in standardisierter Weise reproduzierbare grafische Repräsentation des Vertrages dient als Grundlage der Unterzeichnung.
- *Abwicklungsunterstützung*. Schließlich wird ein Workflow-System eingesetzt, um die Ausführung von Aktivitäten durch vertraglich spezifizierte Parteien in entsprechender Reihenfolge auszuführen. Dabei sind zwei Formen der Unterstützung möglich: bei *informellen Aktivitäten* (welche durch den menschlichen Benutzer durchgeführt werden) notifiziert das Workflow-System die betreffende Partei, daß eine Aktivität zu verrichten ist. Bei formellen Aktivitäten (die vom Workflow-System etwa durch Methodenaufruf ausgeführt werden können) ist der Austausch geeigneter Parameter- und Resultatobjekte erforderlich [MeLL97].

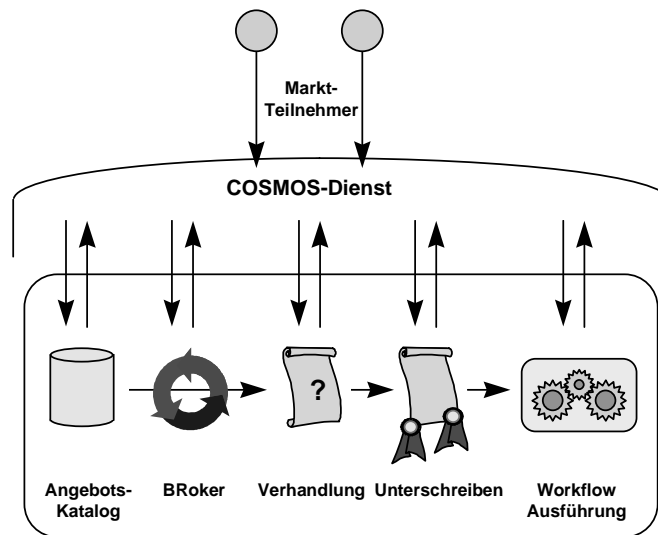


Abb. 2: Funktionen des Electronic Contracting-Systems

Jeder der erwähnten Dienste kann isoliert angeboten und eingesetzt werden. Die Dienste müssen nicht notwendigerweise vom gleichen Anbieter in Anspruch genommen werden. Der Vorteil einer integrierten Nutzung liegt jedoch in der bruchlosen Übergabe von Objekten zwischen den betreffenden Transaktionsphasen (vgl. Abb. 2): heute werden Online-Kataloge, Groupware- und Workflow-Systeme als Komponenten für die unterschiedlichen Transaktionsphasen eingesetzt, jedoch mangelt es für die Aufgabe des Electronic Contracting Service an der Integration dieser Systeme. Die COSMOS-Architektur spezifiziert aus diesem Grunde Schnittstellen und Funktionen, so daß eine logische Integration hin zum „one-stop-service“ erreichbar wird.

#### 4 Das COSMOS-Vertragsmodell

Die COSMOS-Architektur unterstützt alle Transaktionsphasen, daher ist die Nutzung eines einheitlichen Objektmodells über alle Phasen hinweg vorteilhaft. Aus Gründen der Übersichtlichkeit und des beschränkten Platzes wird im folgenden jedoch nur das Vertragsmodell skizziert. Eine ausführliche Behandlung aller Komponenten ist in [MGTM+98] zu finden. Die wesentlichen Merkmale dieses Objektmodells werden im *Vertragsmodell* zusammengefaßt. Es zielt darauf ab, nur jene Vertragsbestandteile zu formalisieren, die semantisch eindeutig abgrenzbar sind und damit eine effiziente Automatisierung erlauben. Abbildung 3 illustriert das entwickelte Objektmodell.

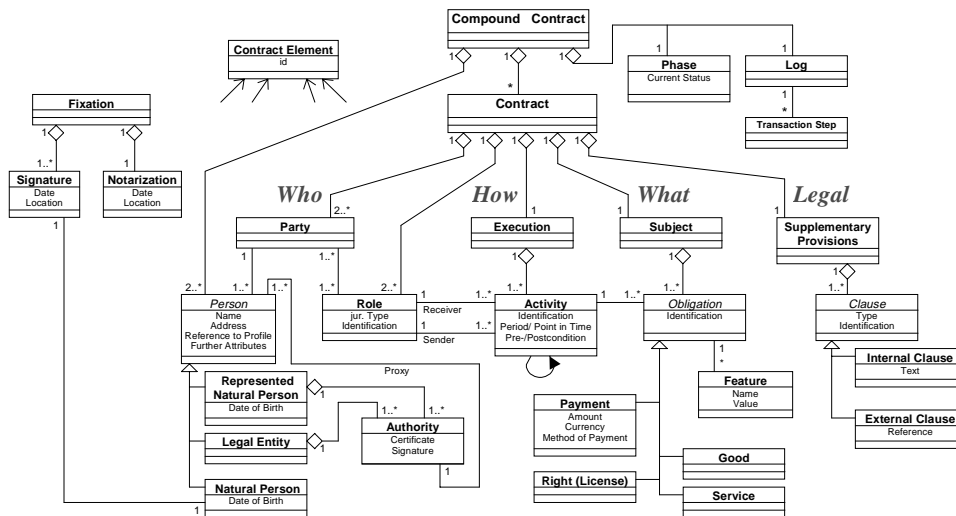


Abb. 3: Das COSMOS Vertragsmodell

Die Bestandteile des Vertragsmodells werden im folgenden nach ihren Schwerpunkten unterschieden:

### Das Vertragswerk

- Ein *Vertragswerk* (engl. "compound contract") ist ein Verbund mehrerer Einzelverträge. Dabei können diese jeweils in sich abgeschlossen sein oder untereinander Querbeziehungen besitzen. Das Vertragswerk setzt sich zusammen aus Verträgen, einer Liste von Unterschriften, sowie Meta-Informationen zum Zustand des Vertragswerks.
- *Vertrag*. Ein Vertrag ist ein Objekt, welches sich aus einer Menge von *Klauseln* zusammensetzt und anwendungsspezifisch um zusätzliche Klauseln erweitert werden kann. Ein Vertrag liegt im juristischen Sinne erst vor, wenn alle *Parteien* diesen signiert haben. Ein Vertrag definiert immer mindestens zwei *Leistungen*, die jeweils von einer der Parteien erbracht werden.
- *Klausel*. Ein *Vertrag* besteht strukturell aus Klauseln. Diese sind im wesentlichen den folgenden Bereichen zuzuordnen: Identifikation der Parteien („Wer“), des Vertragsgegenstands („Was“), der Durchführung („Wie“) sowie weiteren Bedingungen.
- Auf der Ebene des Vertragswerks werden *Metainformationen* verwaltet, die sich auf einzelne Verträge bzw. das gesamte Vertragswerk beziehen. Diese umfassen *Unterschriften* und *Statusinformationen*.

- *Unterschriften* werden von den *Parteien* für einen *Vertrag* geleistet. Befinden sich mehrere Verträge im Vertragswerk, so hat jede Partei alle Verträge zu unterschreiben, an denen sie als Partei beteiligt ist.
- *Beglaubigung des Notars*. Der *Notar* hat die Unterschriften der Parteien zu validieren und mit seiner eigenen Unterschrift den Zusammenhang dieser Unterschriften mit den Verträgen zu beglaubigen.
- *Statusinformation*. Ein Vertrag kann sich in den Zuständen „Schablone“, „Vorschlag“, „Angebot“, „Agreed“, „Unterschrieben“, „In Ausführung“ befinden. Diese Information kann dem Vertragsobjekt über einen Methodenaufruf entnommen werden. Sie sollte sich aus der Zusammensetzung der Vertragsstruktur ableiten lassen, so daß keine explizite Repräsentation des Zustandes (als Attribut) erforderlich ist.

### Parteien

- *Partei*. Organisatorisch wird ein *Vertrag* zwischen zwei oder mehreren Parteien geschlossen. Dabei qualifiziert der Begriff „Partei“ einen *Marktteilnehmer* als Vertragsteilnehmer. Er legt jedoch weder die *Rolle* fest, in der die Partei agiert, noch die Person(en), die sie repräsentiert.
- *Personen* treten (synonym für *Teilnehmer*) als Vertragsparteien auf. Dabei können Personen in mehrere Verträge involviert sein. Eine Person kann - gemäß dem Vertragsmodell - entweder eine juristische oder eine natürliche Person sein. In beiden Fällen sind hinreichend Informationen erforderlich zur Authentisierung der Person, zur Integration in Workflow-Prozesse sowie zur Spezifikation der individuellen Anforderungen an die COSMOS-Infrastruktur.
- *Rolle*. Eine *Partei* nimmt für einen individuellen *Vertrag* eine spezielle Rolle ein, z.B. „Verkäufer“, „Käufer“ oder „Finanzier“.

### Vertragsgegenstand

- Der *Vertragsgegenstand* setzt sich aus mehreren *Leistungen* zusammen, welche die Parteien untereinander austauschen. Eine Leistung besitzt Qualitätsattribute, die sich von Vertrag zu Vertrag unterschiedlich zusammensetzen können. Eine Leistung (engl. "obligation") repräsentiert grundsätzlich die Übertragung eines Rechts. Dieses Recht kann das Eigentum an einer *Ware*, an *Geld*, einer *Dienstleistung* oder eine *Lizenz* sein. Eine Leistung besitzt immer einen (Leistungs-)Erbringer und einen (Leistungs-)Empfänger. Dabei sind in diesem Bereich auch Abhängigkeiten zwischen Leistungen festzulegen. Diese umfassen im Rahmen der *Durchführung* kausale und temporale Abhängigkeiten zwischen jeweils zwei Leistungen: Bedingungen, unter denen eine Leistung zu erbringen ist, Informationen, die zwischen zwei Leistungen transferiert werden, sowie Informationen, die als Parameter bzw. Resultat einer Leistung auftreten. Leistungen sind im einzelnen:



- *Ware*: Hierbei wird anhand der *Leistungsattribute* eine Produktbeschreibung vorgenommen
- *Dienstleistung*: Die QoS-Parameter eines Dienstes werden hier festgelegt. Ferner kann eine Dienstleistung online – d.h. durch einen Server - erbracht werden. Im Falle der Online-Dienstleistung ist dazu eine URL auf eine *Adapter-Bean* erforderlich, die als Proxy für Methodenaufrufe beim Dienstleister verwendet wird (s.u.).
- *Lizenz*: Hierbei wird ein authentisiertes Datenobjekt vom Leistungbringer an den -empfänger übermittelt.
- *Geld*: Auch hier wird ein Datenobjekt übermittelt. Es kann sich um eine Zahlungsautorisierung oder um eine elektronische Münze handeln.

## 5 Implementation

Die Implementation des COSMOS-Systems lehnt sich an die CORBA BOCA (CORBA Business Object Component Architecture) an [OMG98]. Dabei dient der dort definierte *Adapter* als wichtigster Mechanismus zur Integration von Softwaresystemen der Vertragsparteien:

### 5.1 CORBA Business Objects als Integrationsgrundlage

Die CORBA BOCA wurde als eine der architekturellen Grundlagen für den COSMOS-Dienst gewählt [OMG98]. Hierbei liefern Business Objects eine standardisierte Infrastruktur zur Ausführung von Geschäftsanwendungen (als CORBA-Facility) sowie wiederverwendbare Common Business Objects, welche als elementare Softwarekomponenten für anwendungsnahe Softwareprojekte einsetzbar sind. Als Softwarekomponenten im Sinne der „Componentware“ [Grif98] sind Business Objects wiederverwendbare, mit Meta-Informationen versehene, inspizierbare und dynamisch integrierbare Softwaremodule. Die BOCA bietet zudem die Möglichkeit, *Frameworks* für unabhängige Anwendungskomponenten zu spezifizieren.

Unter diesen Voraussetzungen wird davon ausgegangen, daß die Grundlagen für einen offenen Komponentenmarkt erfüllt sind, auf dem die Teilnehmer gegenseitig Softwarekomponenten einbinden. Eine Kohärenz zwischen diesen Komponenten kann jedoch nur gewährleistet werden, wenn eine Standardisierung der Business Objects sowie aller über Organisationsgrenzen hinweg interoperabler Komponenten gewährleistet ist.

Die BOCA führt hierzu den Begriff der Business System Domains (BSD) ein, welche verteilte Objektsysteme charakterisieren, die innerhalb einer abgrenzbaren Organisation (Unternehmen, Abteilung) eingesetzt werden und nach den Richtlinien dieser Organisation konfiguriert und ausgeführt werden. Aus der technischen Perspektive des Electronic Contracting repräsentiert eine BSD eine Vertragspartei und damit die Dienstleistungen, die diese Partei anderen Marktteilnehmern zugänglich macht.

Der synergetische Effekt von Business-Object-Frameworks und deren praktischer Einsatz ist somit eine der wesentlichen Voraussetzungen für einen Electronic Contracting Service im Sinne des COSMOS-Projekts.

Als ein besonderes Konzept der BOCA ist schließlich der *Adapter* zu nennen. Er wird von einem Marktteilnehmer verwendet, um anderen einen logischen Zugangspunkt zu Diensten der eigenen BSD zu bieten. Dabei kann der Adapter als Komponente in die Domänen anderer BSDs herausgegeben werden.

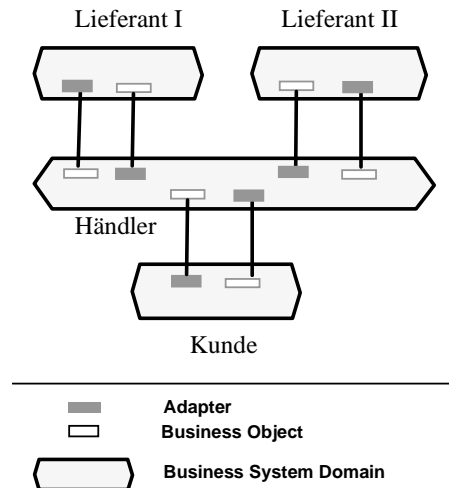


Abb. 4: Business System Domains und Adapter

Technisch kann der Adapter mit dem Client-Stub der CORBA-Architektur verglichen werden, der als Vertreterobjekt für ein Business-Objekt instanziiert werden kann. Im Gegensatz zum CORBA-Stub bildet der Adapter jedoch nicht notwendigerweise die API des Business Objects nur 1:1 ab. Er bietet vielmehr zusätzliche Metainformation über das Business Object und umfaßt lokale Logik z.B. zur Validation von Client-Parametern. Ein Adapter kann somit zur Laufzeit hinsichtlich seiner Eigenschaften inspiziert und zwischen verschiedenen BSDs ausgetauscht werden, um Softwarekomponenten der anderen BSD Zugang zur eigenen Umgebung zu verschaffen. Entsprechend kann durch diese komponentenbasierte Einbindung eine flexible Integration von Softwaresystemen erfolgen. Somit repräsentiert ein Adapter exakt den Teil eines proprietären Softwaresystems, der bewußt an die „Außenwelt“ herausgegeben wird (Abb. 4).

Entsprechend sollte damit der Electronic Contracting Service eine Editor-Komponente besitzen, welche Marktteilnehmern die Möglichkeit bietet, andere nach den von ihnen über den Adapter freigegebenen Informationen zu selektieren. Wird dieser Teilnehmer als Partei in den Vertragsentwurf eingebunden, können anhand der Metainformation Ereignisse und Methoden zwischen den Objekten zugeordnet werden. Diese Integration mehrerer Adapter verschiedener Marktteilnehmer definiert somit einen Workflow: Das

Ereignis der einen Partei wird abgebildet auf einen Methodenaufruf bei der anderen, z.B. um den Bestellvorgang bei Unterschreiten des Lagerbestand zu automatisieren. Der Vorteil dieser Technologie liegt im wesentlichen in der flexiblen Zuordnung von Adaptern.

Damit besteht der Verhandlungsprozeß aus der Konfiguration von Business Objects, der Definition von Beziehungen zwischen diesen sowie der Festlegung von Regeln für ihr Zusammenwirken. Diese Information wird als elektronischer Vertrag festgehalten.

## **5.2 Status der Implementation**

Das COSMOS-Projekt befaßte sich bisher mit dem Entwurf des Vertragsmodells und seiner XML-Repräsentation. Darüber hinaus wurden einige Vertragsschablonen entwickelt, um das Vertragsmodell zu validieren. Als Prototypen stehen Softwarekomponenten zum Editieren, Signieren und Ausführen von Verträgen zur Verfügung; sie zielen darauf ab, den vollständigen Ablauf von der Abgabe eines Angebots bis zur Ausführung des Vertrages zu unterstützen.

Entwicklungen des Katalogs, der Verhandlungsunterstützung sowie des Workflow-Systems der Prototyp-Phase zuzuordnen. Nachdem aufgrund der Prototypentwicklung hinreichende Erfahrungen gesammelt werden konnte, begann seit Dezember 1998 die Implementierung der vollständigen COSMOS-Architektur. Im wesentlichen erfolgt diese Entwicklung mit Hilfe des Java Development Kits 1.1.6 und Oracle 8i als DBMS.

Da sich auch die BOCA-Architektur selbst noch im Stadium der Spezifikation befindet, werden Business Objects mit der notwendigen Funktionalität zunächst im Rahmen des COSMOS-Projekts als erste Prototypen entwickelt.

## **6 Zusammenfassung**

„Electronic Contracting“ ist ein vielversprechendes Anwendungsfeld heutiger Internet-Technologien, welche den beteiligten Unternehmen bei geringem Aufwand ein hohes Optimierungspotential beim Verhandeln und Abschließen von Verträgen bieten. Zum anderen stellt dieser Anwendungsbereich hohe technologische und organisatorische Anforderungen an die COSMOS-Architektur, so daß in besonderem Maße Anwenderwissen und Praxisinformationen in den Entwurf einfließen müssen.

Dabei wird der Ansatz der CORBA Business Objects mit dem Ziel verfolgt, diverse Funktionen auf diese Weise in einer standardisierten Umgebung als vorgefertigte Komponenten nutzen, die sonst selbst zu entwickeln wären. Nichtsdestotrotz ist es ebenfalls denkbar, die COSMOS Architektur an andere Umgebungen anzupassen - wie z.B. das San Francisco Framework von IBM [IBM98].

Schließlich eignet sich die COSMOS-Plattform bisher als geeignete Grundlage, um interdisziplinär zwischen Juristen, Wirtschaftswissenschaftlern und Informatikern neue Ansätze im Bereich des Electronic Commerce zu entwickeln und validieren.

## 7 Literatur

- [CePW98] W. Cellary, W. Picard, W. Wiczerzycki: "Web-based Business-to-Business Negotiation Support". In (F. Griffel, T. Tu., W. Lamersdorf) *Electronic Commerce*, dpunkt, Heidelberg, 1998
- [COSMOS98] COSMOS Project Home Page: <http://www.ponton-hamburg.de/cosmos>
- [MGTM+98] M. Merz, F. Griffel, T. Tu, S. Müller-Wilken et al.: Supporting Electronic Commerce Transactions with Contracting Services. In: *International Journal on Cooperative Information Systems*, 4(7), Dezember 1998, S. 1-25
- [GoPr98] C. F. Goldfarb, P. Prescod: "The Xml Handbook", Prentice Hall 1998
- [Grif98] F. Griffel: "Componentware - Konzepte und Techniken eines Softwareparadigmas". dpunkt, 1998
- [HaBi93] V. Hammer, J. Bizer: "Beweiswert elektronisch signierter Dokumente". In: *Datenschutz und Datensicherung (DuD)* 12/93, S.689-699
- [IBM98] <http://www.ibm.com/Java/Sanfrancisco/technical.html>, 1998.
- [ISO97] ISO/IEC IS 13235-1, ITU/T Draft Rec X950-1, Part 1; ODP Trader Spec., 1997
- [Jens92] K. Jensen. *Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use*, Springer 1992.
- [MeLL97] M. Merz, B. Liberman, W. Lamersdorf: "Using Mobile Agents to Support Inter-organizational Workflow-Management". In: *International Journal on Applied Artificial Intelligence*, 11(6), September 1997, S. 551ff
- [MiAC96] Z. Milosevic, D. Arnold, L. O'Connor, *Inter-enterprise Contract Architecture For Open Distributed Systems: Security Requirements*, WET ICE'96 Workshop on Enterprise Security, Stanford, USA, June 1996.
- [Milo95] Z. Milosevic: „Enterprise Aspects Of Open Distributed Systems“, PhD. Thesis, Department of Computer Science, University of Queensland, 1995.
- [MJML95] K. Müller-Jones, M. Merz, and W. Lamersdorf, *Kooperationsanwendungen: Integrierte Vorgangskontrolle und Dienstvermittlung in offenen verteilten Systemen*. In F. Huber-Wäschle, H. Schauer, and P. Widmayer, eds., *GISI 95 - Herausforderungen eines globalen Informationsverbundes für die Informatik*, Zurich, pp. 518-525, Springer 1995.
- [OMG96] AT&T, DSTC, DEC, HP, ICL, Nortel, and Novell. *Trading Object Service*, OMG Document No.: orbos/96-05-06, Version 1.0, 1996.
- [OMG98] Object Management Group: "CORBA BOCA - Business Object Component Architecture", Spezifikation, OMG Dokument Nr. bom/98-01-07, 1998
- [Schm95] B. Schmid et al.: "Electronic Mall: Banking und Shopping in globalen Netzen". Teubner, Stuttgart 1995
- [Schn95] B. Schneier: "Applied Cryptography: Protocols, Algorithms, and Source Code in C", John Wiley & Sons, 1995
- [TGML98] T. Tu, F. Griffel, M. Merz, W. Lamersdorf: "A Plug-in Architecture Providing Dynamic Negotiation Capabilities for Mobile Agents". In: *Proc. 2. Intl. Workshop on Mobile Agents, MA'98, Stuttgart, 1998, Springer LNCS*, S. 222-236