

Vermittlung und Verwaltung von Diensten in offenen verteilten Systemen

Kay Müller, Kathrin Jones, Michael Merz

Universität Hamburg
Fachbereich Informatik – Datenbanken und Informationssysteme
Vogt-Kölln-Str. 30 — D-22527 Hamburg
e-mail: kmueller @ dbis1.informatik.uni-hamburg.de

Zusammenfassung In offenen verteilten Systemen wird zunehmend eine Vielzahl verschiedenartiger, bekannter und unbekannter Dienste angeboten. Um diese Dienste effektiv und effizient nutzen zu können, sind adäquate Unterstützungsmechanismen für die Dienstvermittlung, die Dienstverwaltung, den Dienstzugriff und die Dienstkontrolle erforderlich. Zur Realisierung dieser Aufgaben stellen wir die im COSM/TRADE-Projekt¹ entwickelte Systemarchitektur vor, wobei in diesem Artikel der Schwerpunkt der Betrachtung in der Vermittlung und Verwaltung sogenannter klassifizierter Dienste durch den TRADE-Trader liegt.

1 Einleitung

Die fortschreitende Entwicklung heutiger Kommunikationsnetze ermöglicht die räumliche Ausdehnung von verteilten Systemen auch über große Distanzen. Die Einführung und Verbreitung der neuartigen Netzwerktechnologien gewährleistet dabei nicht nur eine schnelle und zuverlässige Datenübertragung, sondern macht eine effektive und effiziente Nutzung von entfernten Diensten erst möglich. Dadurch entsteht hinsichtlich des allgemeinen Client/Server-Paradigmas [Svo85] erstmalig ein offener „Markt“ von Diensten, in dem Dienstbringer (Server) dedizierte Dienste über wohldefinierte Schnittstellen einer Vielzahl von externen Anwendungen als Dienstnehmer (Clients) zur Verfügung stellen. Aufgrund dieser Globalisierung des Dienstmarktes können Dienstnehmer aus einer großen Anzahl verschiedenartiger, bereits bekannter bzw. *klassifizierter* oder noch unbekannter bzw. *unklassifizierter* Dienste eine Auswahl treffen und diese lokal nutzen. Hierbei stehen dem Dienstnehmer jedoch mehrere Probleme gegenüber, die die Vermittlung und Verwaltung der verfügbaren Dienstangebote als auch den Zugriff auf die entsprechenden Dienstbringer betreffen. Um diesen Problemen zu begegnen sind generelle systemtechnische Mechanismen erforderlich, welche die Nutzung externer Dienste in offenen verteilten Systemen für einen Dienstnehmer (z. B. einen interaktiven Benutzer oder ein Anwendungsprogramm) angemessen unterstützen. Dabei müssen grundsätzlich zwei unterschiedliche Teilziele verfolgt werden, die in der Praxis oft miteinander im Wettstreit stehen: Einerseits

¹ Common Open Service Market / Service Trading and Coordination Environment

soll eine weitgehende *Standardisierung* und Klassifikation von verschiedenartigen „Diensttypen“ erfolgen, um einen möglichst hohen Grad an *Interoperabilität* und *Wiederverwendbarkeit* der angebotenen Dienste zu erreichen; andererseits liegt jedoch gerade in der *Individualität* unklassifizierter Dienstangebote für viele Dienstbetreiber ein wichtiger Anreiz, ihre Dienste in derartigen Märkten überhaupt kommerziell anzubieten. Zur Erfüllung beider Teilziele ergibt sich unmittelbar die Forderung nach der Entwicklung einer Systemarchitektur, welche sowohl den spezifischen Anforderungen der verschiedenartigen Dienste gerecht wird, als auch eine Integration der entwickelten Unterstützungsmechanismen für die Dienstvermittlung, die Dienstverwaltung, den Dienstzugriff und die Dienstkontrolle erlaubt.

Dieser Artikel gliedert sich wie folgt: Im folgenden Abschnitt wird anhand von zwei Beispielen die Untergliederung der verschiedenartigen Dienste in offenen Dienstmärkten in die zwei groben Kategorien der klassifizierten und unklassifizierten Dienste dargestellt. Anschließend wird im dritten Abschnitt die COSM/TRADE-Systemarchitektur vorgestellt, welche eine integrierte Unterstützung beliebiger Dienste, d. h. sowohl klassifizierter als auch unklassifizierter, bietet. Im vierten Abschnitt wird speziell auf die Dienstvermittlung und -verwaltung klassifizierter Dienstangebote mittels des *TRADE-Traders* eingegangen. Anschließend wird im fünften Abschnitt die zur Zeit durchgeführte Prototypimplementierung des *TRADE-Traders* beschrieben, welche auf Basis des „Distributed Computing Environment“ (DCE) [Fou92] und des Encina-Toolkits [IBM93] realisiert wird. Abschließend wird eine Zusammenfassung und ein Ausblick auf weitere aktuelle Forschungsaktivitäten im Rahmen des COSM/TRADE-Projektes gegeben.

2 Klassifizierte und unklassifizierte Dienste

Hinsichtlich der schon oben angedeuteten Verschiedenartigkeit der Dienste in verteilten Systemen werden diese grob in die Kategorien der *klassifizierten* bzw. *standardisierten* und *unklassifizierten* Dienste untergliedert. Diese Klassifikation dient als Grundlage für die im nachfolgenden Abschnitt dargestellte COSM/TRADE-Architektur. Im folgenden sollen diese beiden Dienstarten an je einem Beispiel veranschaulicht werden.

Ein Druckdienst als Beispiel für einen klassifizierten Dienst: In Unternehmen tritt oftmals die Situation auf, daß ein Benutzer mit Hilfe seines Textverarbeitungsprogramms ein Dokument auf einem Drucker mit spezifischen Eigenschaften bzw. *Attributen* ausdrucken möchte. Beispiele für derartige Eigenschaften sind der Ort des Druckers, die Papiergröße und die Druckqualität. Stehen mehrere in Frage kommende Druckdienstangebote verschiedener Druckdienstbringer zur Verfügung, so soll der Dienstnehmer (in diesem Fall das Anwendungsprogramm) bezüglich der Auswahl und der Vermittlung eines für ihn am besten geeigneten Dienstangebotes unterstützt werden. Diese Aufgabe wird in zukünftigen verteilten Systemen durch sogenannte *Trader* erbracht [ODP92], welche eine weitgehend automatische Auswahl und Vermittlung von

Dienstangeboten und der entsprechenden Dienstbringer aufgrund bestimmter Auswahlkriterien ermöglichen. Um eine derartige Vermittlung und Verwaltung von Dienstangeboten durchführen zu können, ist vorab eine *Standardisierung* bzw. *Klassifikation* von Diensten bzgl. ihrer Funktionalität und Semantik erforderlich. Zu diesem Zweck dient der Begriff des *Diensttyps* [Jon94, ODP92], welcher eine Formalisierung des Dienstbegriffs ermöglicht. Der Diensttyp legt den Diensttypnamen, die Dienstattributnamen und -typen und den Schnittstellentyp des Dienstes fest. Der Schnittstellentyp bestimmt hierbei die Menge der Operationstypen und spezifiziert u. a. die operationale Semantik des Dienstes. Abbildung 2 zeigt das Beispiel einer Diensttypbeschreibung des Diensttyps „PrintService“ für das hier angeführte Beispiel eines Druckdienstes. Nachdem der definierte Diensttyp dem Trader bekanntgegeben wurde, ist es speziellen Erbringern eines Druckdienstes möglich, sich mit ihren Dienstangeboten unter dem speziellen Diensttyp registrieren zu lassen. Entsprechend können die Dienstnehmer beim Trader geeignete Druckdienstangebote erfragen und sich bei entsprechendem Wunsch das am besten geeignete durch den Trader auswählen lassen. Die Auswahl erfolgt hierbei über die Diensttypkonformität [Car89] und die aktuellen Werte der Dienstattribute der Angebote. Neben statischen Attributen (z. B. die aktuelle Papiergröße) werden auch dynamische Informationen über die registrierten Dienstbringer berücksichtigt. Beim Druckdienst kann z. B. die aktuelle Warteschlangenlänge als dynamisches Attribut herangezogen werden. Grundsätzlich bietet der Trader verschiedene Strategien zur Auswahl geeigneter Dienstbringer an, die sich in der Qualität bzgl. der Wahl des „am besten geeigneten“ Dienstangebotes bzw. Dienstbringers unterscheiden [WT90].

Ein Satellitenvermietungsdienst als Beispiel für einen unklassifizierten Dienst: Über eine Satellitenvermietung können Satelliten für Datenübertragungen über weite Distanzen gebucht werden. Das Anbieten einer derartigen Dienstfunktionalität kann aus heutiger Sicht eine Innovation gegenüber den Diensten konkurrierender Vermietungsdienste darstellen, so daß ein besonderes Interesse besteht, möglichst früh und vor dem Auftreten anderer, vergleichbarer Dienste in Erscheinung zu treten und für potentielle Dienstnehmer nutzbar zu werden, um die damit verbundenen Wettbewerbsvorteile ausnutzen zu können. In der Regel wird ein derartiger Dienst nicht über einen standardisierten Diensttyp vermittelbar sein, da dieses die vorherige Bekanntgabe bzw. Definition eines allgemein zugänglichen Diensttyps voraussetzt, der die Funktionalität des neuartigen Dienstes vorab vollständig spezifiziert. Dieses wäre unter ökonomischen Gesichtspunkten jedoch eventuell sogar kontraproduktiv für den Dienstbetreiber, da in diesem Fall der Wettbewerbsvorteil der Innovation des neuartigen Dienstes stark eingeschränkt werden würde. Hieraus ergibt sich unmittelbar, daß die Vermittlung und die Verwaltung eines derartigen Dienstangebotes über einen klassischen Trader nicht durchführbar ist, da eine Zuordnung zu einem entsprechenden Diensttyp nicht möglich ist. Dieses bedeutet jedoch auch, daß der Zugriff auf einen unklassifizierten Dienst nur über spezielle *benutzerorientierte* Werkzeuge erfolgen kann, bei denen der menschliche Benutzer die Funktionalität bzw. Semantik des angebotenen Dienstes interaktiv erschließt

und als Entscheidungsgrundlage verwendet [MML94].

3 Die COSM/TRADE-Architektur

Im vorherigen Abschnitt wird deutlich, daß sich durch den zentralen Begriff des Diensttyps bzw. dessen Nichtanwendbarkeit signifikante Unterschiede bei der Dienstvermittlung, der Dienstverwaltung und des Dienstzugriffs der beiden unterschiedlichen Dienstarten ergeben. Entsprechend müssen die jeweiligen charakteristischen Eigenschaften der Dienste bei dem Entwurf systemtechnischer Unterstützungsmechanismen angemessen berücksichtigt werden. Das Ziel der COSM/TRADE-Architektur ist es nun, eine Architektur zu schaffen, welche eine integrierte Unterstützung von Diensten *beliebiger* Art in offenen Dienstmärkten ermöglicht. Hierfür wird durch die Architektur ein Lebenslaufmodell für Dienste realisiert: So kann zum Beispiel die Dienstbeschreibung des unklassifizierten Satellitenvermietungsdienstes als Vorbild für eine spätere Definition einer Diensttypbeschreibung des neuen Diensttyps „SatelliteRental“ dienen. Auf diese Weise kann ein nahtloser Übergang von einem unklassifizierten in einen klassifizierten Dienst erfolgen und eine adäquate, dienststartspezifische Systemunterstützung gewährleistet werden. Anlehnend an eine in [Ber93a] vorgeschlagene Begriffsbildung stellt sich die COSM/TRADE-Architektur wie folgt dar (Abbildung 1). Aufsetzend auf sogenannten *Middleware-Diensten*, wie sie zum Beispiel

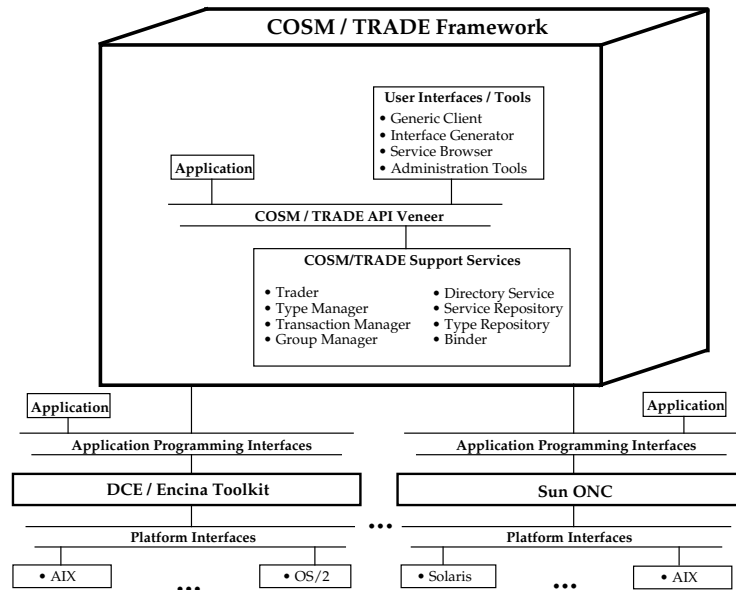


Abbildung 1. COSM/TRADE-Architektur

durch die DCE-Umgebung und das Encina-Toolkit bereitgestellt werden, wird durch die COSM/TRADE-Architektur eine Systemumgebung (*Framework*) realisiert, welche eine spezialisierte Anwendungsumgebung für die Dienstvermittlung, die Dienstverwaltung, den Dienstzugriff und die Dienstkontrolle für beliebige Dienste in verteilten Systemen implementiert. Diese besteht zum einen aus einer Anzahl von Systemdiensten (*Support Services*) wie zum Beispiel dem Trader zur Vermittlung und Verwaltung von klassifizierten Diensten; zum anderen werden auf der anwendungsnahen Ebene spezielle Werkzeuge (*Tools*) für die Benutzung und Programmierung der COSM/TRADE-Systemumgebung angeboten. Beispiele hierfür sind interaktive Werkzeuge wie der Generische Klient [ML93] zum Zugriff auf unklassifizierte Dienste oder eine grafische Management-schnittstelle für den Trader. Der Zugriff der Werkzeuge und Anwendungen auf die COSM/TRADE-Systemumgebung erfolgt hierbei über spezifische Programmierschnittstellen, welche die Komplexität des Zugriffs auf die COSM/TRADE-Systemdienste und die darunterliegenden Middleware-Dienste weitgehend verbergen.

Im folgenden wird speziell auf die Dienstvermittlung und -verwaltung klassifizierter Dienste durch die Trader-Komponente der COSM/TRADE-Architektur eingegangen. Für spezielle Fragen bzgl. der Unterstützung unklassifizierter Dienste sei auf [ML93, MML94] hingewiesen.

4 Vermittlung und Verwaltung klassifizierter Dienste: Der TRADE-Trader

Abbildung 2 stellt die modulare Architektur des TRADE-Traders vor, wobei der *Typmanager* und die *Dienstangebotsverwaltung* zwei Kernbausteine des Traders sind. Zusätzlich benötigt der Trader zur Realisierung seiner Aufgaben eine Reihe weiterer Komponenten, hierzu gehören ein *Namensdienst*, ein *Ablagedienst* (Repository) sowie ein *Autorisierungs-* und *Authentisierungsdienst*. Diese drei Komponenten können auch unabhängig vom Trader genutzt werden.

Der *Typmanager* ist für die Verwaltung des gesamten Typsystems des Traders verantwortlich, wobei die Konformitätsbeziehungen der Schnittstellentypen bzw. Dienstypen im Schnittstellen- und Diensttypgraph dargestellt werden. Der Typmanager besitzt eine Administrationsschnittstelle zum Einfügen von Dienst- und Schnittstellentypen, eine Browsing-Schnittstelle zum Blättern und Suchen von Dienst- und Schnittstellentypen und eine interne Trader-Schnittstelle, welche ausschließlich durch den Trader zum Auffinden konformer Dienstypen und zum Überprüfen der Konformität eines Dienstes zum Dienstyp genutzt wird. Die Aufgabe des Typmanagements durch den Trader wird durch die Nutzung eines externen Ablagedienstes unterstützt, um sämtliche Typdefinitionen persistent abzulegen. Diese Typinformationen können dann bei Bedarf effizient durch externe Werkzeuge, z. B. Werkzeuge für die verteilte Anwendungsprogrammierung, genutzt werden.

Die Verwaltung von Dienstangeboten (Einfügen, Löschen, Suchen) wird vom Trader über die Schnittstelle der **Dienstangebotsverwaltung** durchgeführt.

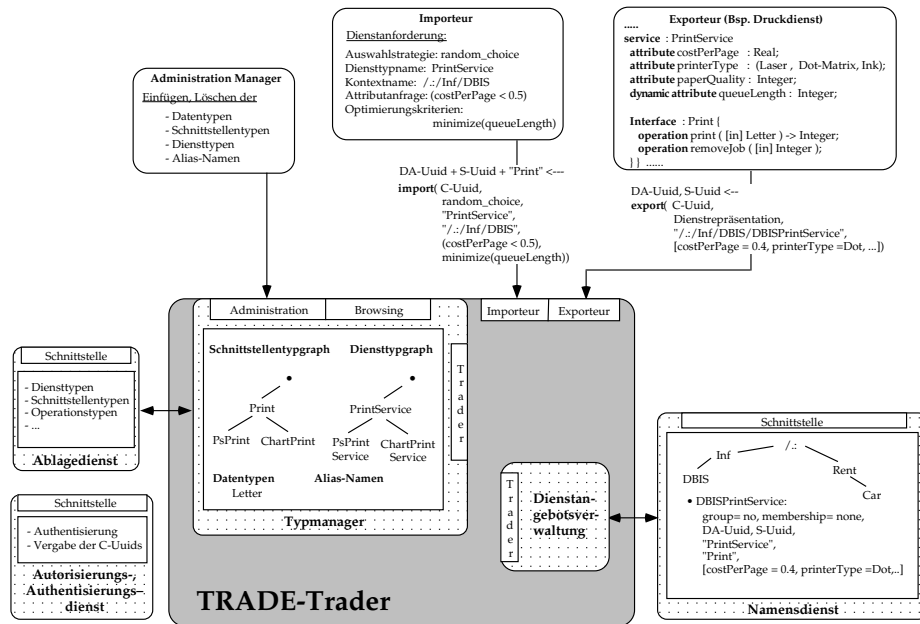


Abbildung 2. Architektur des TRADE-Traders

Diese verwaltet die Dienstangebote der Exporteure mit Hilfe eines Namensdienstes in einer hierarchisch aufgebauten Kontextstruktur. Jeder Kontext besitzt einen global eindeutigen Kontextnamen und verwaltet die Informationen der in diesen Kontext exportierten Dienstangebote. Für jedes Dienstangebot werden u. a. der Diensttypname, der Schnittstellentypname, die Schnittstellenkennung, sowie die Namen und Werte der statischen Attribute gespeichert. Das Einfügen und Löschen von Kontexten in die Kontextstruktur erfolgt durch den Administrator.

Mit Hilfe des Authentisierung- und Authentisierungsdienstes werden sämtliche Zugriffe auf die einzelnen Komponenten des Systems kontrolliert. Hierbei erfolgt je nach Grad der Sicherheitsanforderungen eine Überprüfung der Zugriffsberechtigung. Dieses erfolgt sowohl beim Zugriff der Importeure bzw. Exporteure auf den Trader als auch beim Zugriff der Importeure auf die Exporteure. Auch der Trader selbst muß beim Zugriff auf seine Unterstützungsdienste, z. B. den Namensdienst, überprüft werden.

Der Zugriff der Exporteure (Dienstbringer) und Importeure (Dienstnehmer) auf den TRADE-Trader erfolgt über die Schnittstellen für Exporteure und für Importeure. Bevor ein Exporteur einen Dienst exportieren kann, muß sowohl der Schnittstellentyp als auch der Dienstyp des Dienstes über die Administrationsschnittstelle des Typmanagers eingefügt worden sein. Auch der entsprechende Kontext muß bereits im Namensraum vorhanden sein. Erst dann können

Exporteure ihre Dienstangebote mittels der Export-Operation beim Trader registrieren. Nach dem Exportieren eines Dienstangebotes können die Dienstnehmer bzw. Importeure dieses über den Trader mittels der Import-Operation importieren. Die Auswahl eines geeigneten Dienstbringers kann hierbei anhand verschiedener Auswahlstrategien vorgenommen werden, z. B. durch Angabe einer zusammengesetzten Attributanfrage. Die Schnittstelle für Exporteure ermöglicht außerdem die Änderung der Werte der statischen Attribute eines Dienstangebotes und das Löschen eines Dienstangebotes.

5 Prototypimplementation des TRADE-Traders

Als Implementierungsplattform dient eine UNIX-Rechnerumgebung mit mehreren IBM RS/6000 Arbeitsplatzrechnern. Der Prototyp wird hauptsächlich in der Programmiersprache C realisiert. Eine wichtige Anforderung an die Implementierung der gesamten COSM/TRADE-Architektur ist das Aufsetzen auf weitgehend standardisierte Entwicklungsbausteine. Aus diesem Grund wird die aktuelle Prototypimplementation des TRADE-Traders weitgehend auf Basis des „Distributed Computing Environment“ (DCE) und des Encina-Toolkits durchgeführt, welche auf den IBM RS/6000 Arbeitsplatzrechnern installiert sind. Dadurch wird sowohl ein hoher Grad an Plattformunabhängigkeit der gesamten Architektur gewährleistet, als auch eine weitreichende Nutzung vorhandener Standardbasisdienste möglich. Wie schon in Abbildung 2 ersichtlich wird, lassen sich einige Teilkomponenten der Architektur des TRADE-Traders direkt mit Hilfe von DCE- und Encina-Basisdiensten realisieren. So werden die DCE-Namensdienste (Cell Directory Service bzw. Global Directory Service) für die Realisierung der Dienstangebotsverwaltung eingesetzt, wobei die X/Open Directory Service-Programmierschnittstelle [Fou93] benutzt wird. Mit Hilfe des DCE-Sicherheitsdienstes (Security Service) wird ein adäquater Sicherheitsmechanismus für den Zugriff der Dienstnehmer auf die Dienstbringer gewährleistet. Der eigentliche Zugriff auf die Dienstbringer erfolgt mittels des DCE-RPCs bzw. des Encina-TRPCs, falls ein transaktionaler Zugriff auf den Dienstbringer erforderlich ist.

6 Zusammenfassung und Ausblick

In diesem Beitrag wurde schwerpunktmäßig die Vermittlung und Verwaltung von *klassifizierten* Diensten mittels des TRADE-Traders betrachtet und dessen Architektur vorgestellt. Die einleitende Kategorisierung von Diensten hat verdeutlicht, daß klassifizierte und unklassifizierte Dienste jeweils verschiedenartige Unterstützungsmechanismen benötigen. Diese Unterstützungsmechanismen sollten jedoch nicht isoliert sein, sondern eine integrierte Behandlung der verschiedenartigen Dienste ermöglichen. Eine derartige *integrierte* Systemunterstützung für die Vermittlung und Verwaltung *beliebiger* Dienste ist das Hauptziel der vorgestellten COSM/TRADE-Architektur. Ein weiterer Schwerpunkt laufender Ar-

beiten ist neben der Entwicklung verschiedener verteilter Anwendungen auf Basis der COSM/TRADE-Systemumgebung insbesondere die Untersuchung und Entwicklung von Unterstützungsmechanismen zur *Ablaufkontrolle* komplexer Anwendungsvorgänge (z. B. innerhalb von Workflow Management-Anwendungen [Ber93b]), bei denen sich der Kontrollfluß nicht nur über einzelne Dienstbringer sondern über eine Vielzahl verteilter und miteinander kooperierender Dienstbringer erstreckt. Insbesondere die Integration des Trading-Konzeptes in eine derartige Kontrollumgebung wird hierbei untersucht.

Danksagung

Herrn Prof. Dr. W. Lamersdorf sei an dieser Stelle für seine hilfreichen Anregungen bei dem Entstehen dieses Artikels gedankt.

References

- [Ber93a] P. A. Bernstein. *Middleware – an architecture for distributed system services*. Technical report CRL 93/6, Digital Equipment Corporation, Cambridge Research Lab, 1993.
- [Ber93b] R. Berthold. *Workflow Management*. Teubner, 1993.
- [Car89] L. Cardelli. *Typeful programming*. Technical report, DEC SRC Research Report No. 45, 1989.
- [Fou92] Open Software Foundation. *Introduction to OSF DCE*. Prentice-Hall, Englewood Cliffs, New Jersey, 1992.
- [Fou93] Open Software Foundation. *OSF DCE Application Development Reference*. Prentice-Hall, Englewood Cliffs, New Jersey, 1993.
- [IBM93] IBM. *Encina for AIX/6000 Base Reference*. Number SC23-2464. IBM, 1993.
- [Jon94] K. Jones. *Vermittlung und Verwaltung von Diensten in offenen verteilten Systemen: Ein Objekt- und Architekturmodell*. Diplomarbeit, Fachbereich Informatik, Universität Hamburg, 1994.
- [ML93] M. Merz und W. Lamersdorf. *Cooperation support for an open service market*. In *Proceedings of the IFIP TC6/WG6.1 International Conference on Open Distributed Processing*. North-Holland, Elsevier Science Publishers B.V., 1993.
- [MML94] M. Merz, K. Müller und W. Lamersdorf. *Service trading and mediation in distributed computing environments*. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS '94)*. IEEE Computer Society Press, 1994. (to appear).
- [ODP92] ISO/IEC JTC1/SC21/WG7: Working Document on Topic 9.1 – ODP Trader. International Standardization Organization, November 1992.
- [Svo85] L. Svobodava. *Client/server model of distributed processing*. Technical report RZ 1350, IBM, Zürich, 1985.
- [WT90] A. Wolisz und V. Tschammer. *Service provider selection in an open services environment*. In *Proceedings of the 2nd IEEE Workshop on Future Trends on Distributed Computing in the 1990's*, S. 229–235, Cairo, Egypt, September 1990. IEEE.