

Informatik – Fachberichte

- Band 144: H. Oberquelle, Sprachkonzepte für benutzergerechte Systeme. XI, 315 Seiten. 1987.
- Band 145: K. Rothermel, Kommunikationskonzepte für verteilte transaktionsorientierte Systeme. XI, 224 Seiten. 1987.
- Band 146: W. Damm, Entwurf und Verifikation mikroprogrammierter Rechnerarchitekturen. VIII, 327 Seiten. 1987.
- Band 147: F. Belli, W. Görke (Hrsg.), Fehlertolerierende Rechensysteme / Fault-Tolerant Computing Systems. 3. Internationale GI/ITG/GMA-Fachtagung, Bremerhaven, September 1987. Proceedings. XI, 389 Seiten. 1987.
- Band 148: F. Puppe, Diagnostisches Problemlösen mit Expertensystemen. IX, 257 Seiten. 1987.
- Band 149: E. Paulus (Hrsg.), Mustererkennung 1987. 9. DAGM-Symposium, Braunschweig, Sept./Okt. 1987. Proceedings. XVII, 324 Seiten. 1987.
- Band 150: J. Halin (Hrsg.), Simulationstechnik. 4. Symposium, Zürich, September 1987. Proceedings. XIV, 690 Seiten. 1987.
- Band 151: E. Buchberger, J. Retti (Hrsg.), 3. Österreichische Artificial-Intelligence-Tagung. Wien, September 1987. Proceedings. VIII, 181 Seiten. 1987.
- Band 152: K. Morik (Ed.), GWA1-87. 11th German Workshop on Artificial Intelligence. Geseke, Sept./Okt. 1987. Proceedings. XI, 405 Seiten. 1987.
- Band 153: D. Meyer-Ebrecht (Hrsg.), ASST'87. 6. Aachener Symposium für Signaltheorie. Aachen, September 1987. Proceedings. XII, 390 Seiten. 1987.
- Band 154: U. Herzog, M. Paterok (Hrsg.), Messung, Modellierung und Bewertung von Rechensystemen. 4. GI/ITG-Fachtagung, Erlangen, Sept./Okt. 1987. Proceedings. XI, 388 Seiten. 1987.
- Band 155: W. Brauer, W. Wahlster (Hrsg.), Wissensbasierte Systeme. 2. Internationaler GI-Kongreß, München, Oktober 1987. XIV, 432 Seiten. 1987.
- Band 156: M. Paul (Hrsg.), GI – 17. Jahrestagung. Computerintelligenter Arbeitsplatz im Büro. München, Oktober 1987. Proceedings. XIII, 934 Seiten. 1987.
- Band 157: U. Mahn, Attributierte Grammatiken und Attributierungsalgorithmen. IX, 272 Seiten. 1988.
- Band 158: G. Cyranek, A. Kachru, H. Kaiser (Hrsg.), Informatik und „Dritte Welt“. X, 302 Seiten. 1988.
- Band 159: Th. Christaller, H.-W. Hein, M. M. Richter (Hrsg.), Künstliche Intelligenz. Frühjahrsschulen, Dassel, 1985 und 1986. VII, 342 Seiten. 1988.
- Band 160: H. Mäncher, Fehlertolerante dezentrale Prozeßautomatisierung. XVI, 243 Seiten. 1987.
- Band 161: P. Peinl, Synchronisation in zentralisierten Datenbanksystemen. XII, 227 Seiten. 1987.
- Band 162: H. Stoyan (Hrsg.), Begründungsverwaltung. Proceedings, 1986. VII, 153 Seiten. 1988.
- Band 163: H. Müller, Realistische Computergraphik. VII, 146 Seiten. 1988.
- Band 164: M. Eulenstein, Generierung portabler Compiler. X, 235 Seiten. 1988.
- Band 165: H.-U. Heiß, Überlast in Rechensystemen. IX, 176 Seiten. 1988.
- Band 166: K. Hörmann, Kollisionsfreie Bahnen für Industrieroboter. XII, 157 Seiten. 1988.
- Band 167: R. Lauber (Hrsg.), Prozeßrechensysteme '88. Stuttgart, März 1988. Proceedings. XIV, 799 Seiten. 1988.
- Band 168: U. Kastens, F. J. Rammig (Hrsg.), Architektur und Betrieb von Rechensystemen. 10. GI/ITG-Fachtagung, Paderborn, März 1988. Proceedings. IX, 405 Seiten. 1988.
- Band 169: G. Heyer, J. Krems, G. Görz (Hrsg.), Wissensarten und ihre Darstellung. VIII, 292 Seiten. 1988.
- Band 170: A. Jaeschke, B. Page (Hrsg.), Informatikanwendungen im Umweltbereich. 2. Symposium, Karlsruhe, 1987. Proceedings. X, 201 Seiten. 1988.
- Band 171: H. Lutterbach (Hrsg.), Non-Standard Datenbanken für Anwendungen der Graphischen Datenverarbeitung. GI-Fachgespräch, Dortmund, März 1988, Proceedings. VII, 183 Seiten. 1988.
- Band 172: G. Rahmstorf (Hrsg.), Wissensrepräsentation in Expertensystemen. Workshop, Herrenberg, März 1987. Proceedings. VII, 189 Seiten. 1988.
- Band 173: M. H. Schulz, Testmustergenerierung und Fehlersimulation in digitalen Schaltungen mit hoher Komplexität. IX, 165 Seiten. 1988.
- Band 174: A. Endrös, Rechtsprechung und Computer in den neunziger Jahren. XIX, 129 Seiten. 1988.
- Band 175: J. Hülsemann, Funktioneller Test der Auflösung von Zugriffskonflikten in Mehrrechnersystemen. X, 179 Seiten. 1988.
- Band 176: H. Trost (Hrsg.), 4. Österreichische Artificial-Intelligence-Tagung. Wien, August 1988. Proceedings. VIII, 207 Seiten. 1988.
- Band 177: L. Voelkel, J. Pliquett, Signaturanalyse. 223 Seiten. 1989.
- Band 178: H. Göttler, Graphgrammatiken in der Softwaretechnik. VIII, 244 Seiten. 1988.
- Band 179: W. Ameling (Hrsg.), Simulationstechnik. 5. Symposium. Aachen, September 1988. Proceedings. XIV, 538 Seiten. 1988.
- Band 180: H. Bunke, O. Kübler, P. Stucki (Hrsg.), Mustererkennung 1988. 10. DAGM-Symposium, Zürich, September 1988. Proceedings. XV, 361 Seiten. 1988.
- Band 181: W. Hoepfner (Hrsg.), Künstliche Intelligenz. GWA1-88, 12. Jahrestagung. Eingerfeld, September 1988. Proceedings. XII, 333 Seiten. 1988.
- Band 182: W. Barth (Hrsg.), Visualisierungstechniken und Algorithmen. Fachgespräch, Wien, September 1988. Proceedings. VIII, 247 Seiten. 1988.
- Band 183: A. Clauer, W. Purgathofer (Hrsg.), AUSTROGRAPHICS '88. Fachtagung, Wien, September 1988. Proceedings. VIII, 267 Seiten. 1988.
- Band 184: B. Gollan, W. Paul, A. Schmitt (Hrsg.), Innovative Informations-Infrastrukturen. I.I.I. – Forum, Saarbrücken, Oktober 1988. Proceedings. VIII, 291 Seiten. 1988.
- Band 185: B. Mitschang, Ein Molekül-Atom-Datenmodell für Non-Standard-Anwendungen. XI, 230 Seiten. 1988.
- Band 186: E. Rahm, Synchronisation in Mehrrechner-Datenbanksystemen. IX, 272 Seiten. 1988.
- Band 187: R. Valk (Hrsg.), GI – 18. Jahrestagung I. Vernetzte und komplexe Informatik-Systeme. Hamburg, Oktober 1988. Proceedings. XVI, 776 Seiten.
- Band 188: R. Valk (Hrsg.), GI – 18. Jahrestagung II. Vernetzte und komplexe Informatik-Systeme. Hamburg, Oktober 1988. Proceedings. XVI, 704 Seiten.
- Band 189: B. Wolfinger (Hrsg.), Vernetzte und komplexe Informatik-Systeme. Industrieprogramm zur 18. Jahrestagung der GI, Hamburg, Oktober 1988. Proceedings. X, 229 Seiten. 1988.
- Band 190: D. Maurer, Relevanzanalyse. VIII, 239 Seiten. 1988.
- Band 191: P. Levi, Planen für autonome Montageroboter. XIII, 259 Seiten. 1988.
-

Informatik-Fachberichte 241

Herausgeber: W. Brauer

im Auftrag der Gesellschaft für Informatik (GI)

Alexander Schill

Migrationsteuerung und Konfigurationsverwaltung für verteilte objektorientierte Anwendungen



Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo Hong Kong

Autor

Alexander Schill

Institut für Telematik, Universität Karlsruhe

Am Zirkel 2, D-7500 Karlsruhe 1

CR Subject Classifications (1987): C.2.4, D.2.10, D.3.3, D.4.4

ISBN-13: 978-3-540-52542-4 e-ISBN-13: 978-3-642-75685-6

DOI: 10.1007/ 978-3-642-75685-6

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, bei auch nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der Fassung vom 24. Juni 1985 zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

© Springer-Verlag Berlin Heidelberg 1990

Vorwort

Mit der zunehmenden Leistungsfähigkeit verteilter Rechnersysteme und dem wachsenden Bedarf nach verteilten Anwendungen steigen auch die Anforderungen an die Entwicklungsunterstützung für solche Anwendungen. Geeignete Methoden und Werkzeuge und vor allem geeignete Abstraktionen von zugrundeliegenden Systemdetails sind Voraussetzung für einen einfachen und effizienten Entwicklungsvorgang. In den letzten Jahren gewannen in diesem Bereich verteilte objektorientierte Ansätze nach dem Vorbild des Smalltalk-Systems zunehmend an Bedeutung. Dabei können Objekte als Zusammenfassung von Daten und Operationen auf verschiedene Rechnerknoten verteilt sein, mit Hilfe verteilungstransparenter Operationsaufrufe kommunizieren und durch Migrationen zur Laufzeit zwischen Rechnerknoten verlagert werden. Eine verteilte Anwendung setzt sich aus einer Menge verteilter Objekte zusammen, deren Anzahl und Lokation dynamisch variieren kann. Der verteilte objektorientierte Ansatz zeichnet sich gegenüber herkömmlichen Mechanismen für verteilte Anwendungen wie z.B. dem einfachen Nachrichtenaustausch oder dem Remote Procedure Call durch seine Uniformität und seine weitgehende Verteilungstransparenz aus.

Die vorliegende Arbeit führt zunächst in den verteilten objektorientierten Ansatz ein und vergleicht ihn mit anderen Mechanismen zur Erstellung verteilter Anwendungen. In ihrem Kernteil befaßt sich die Arbeit dann mit der Erweiterung dieses Ansatzes um neue Konzepte zur Steuerung von Objektmigrationen und zur Verwaltung einer verteilten Anwendungskonfiguration. Dabei werden integrierte Methoden und Werkzeuge zur statischen und dynamischen Platzierung der Objekte einer verteilten Anwendung auf einem verteilten Rechnersystem angeboten. Zusätzlich wird die Strukturierung der Anwendungskomponenten durch geeignete Abstraktionsmechanismen unterstützt.

Der Ansatz zur Migrationssteuerung hat die dynamische Zusammenführung kommunizierender Objekte an einer gemeinsamen Lokation zum Ziel. Motivation hierfür ist der signifikante Unterschied zwischen dem Aufwand für lokale und entfernte Interobjekt-Kommunikation. Die Migrationssteuerung basiert auf a-priori-Information über die erwartete Objektkommunikation während bestimmter Zeiträume sowie Laufzeitinformation über die aktuelle Objektverteilung. Ein verteiltes Laufzeitprotokoll realisiert auf dieser Basis geeignete Objektmigrationen. Das Verhalten der verteilten Anwendung und die getroffenen Entscheidungen werden zusätzlich von einem Monitorsystem analysiert, um eine teilweise Automatisierung von Migrationsentscheidungen zu ermöglichen.

Der Ansatz zur verteilten Konfigurationsverwaltung stellt dem Anwendungsentwickler eine neue objektorientierte Konfigurationssprache zur Verfügung. Damit wird die deklarative Beschreibung einer initialen Anwendungskonfiguration, bestehend aus typisierten Objekten mit hierarchischer Abstraktion, logischen Kommunikationsbeziehungen und Platzierungsvorgaben, ermöglicht. Zusätzlich werden dynamische Konfigurationsänderungen zur späteren Adaption einer Anwendungsstruktur angeboten.

Während die Konfigurationsverwaltung die explizite Grobstrukturierung und Grobplatzierung unterstützt, stellt die Migrationssteuerung Methoden und Werkzeuge zur teilautomatisierten Feinplatzierung der Anwendungsobjekte zur Verfügung. Insgesamt wird ein integrierter Ansatz geboten, der zwar den Anwendungsentwickler durch einen hohen Grad an Verteilungstransparenz in bezug auf die Platzierung entlastet, ihm aber dennoch wirksame Mittel zur expliziten Einflußnahme auf Platzierungs- und Strukturierungsentscheidungen bereitstellt.

Die vorliegende Arbeit entspricht inhaltlich meiner im Juli 1989 von der Fakultät für Informatik der Universität Karlsruhe genehmigten Dissertation. Die Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Telematik der Fakultät für Informatik an der Universität Karlsruhe.

Meinen besonderen Dank möchte ich Herrn Prof. Dr. G. Krüger aussprechen, der die Arbeit betreut hat. Er hat mit vielen fachlichen und praktischen Ratschlägen und mit engagierter Unterstützung bei organisatorischen Fragen ganz wesentlich zum Gelingen der Arbeit beigetragen. Durch das Gewähren entsprechender Freiräume im Rahmen der Institutstätigkeit ermöglichte er mir die intensive Beschäftigung mit dem Thema der Arbeit. Die gute technische Ausrüstung seines Instituts war für den praktischen Teil sehr wichtig. Durch die Förderung von Tagungs- und Informationsreisen ins Ausland konnte ich interessante internationale Kontakte knüpfen und wertvolle zusätzliche Erfahrungen sammeln.

Herrn Prof. Dr. S. Jähnichen möchte ich für die Übernahme des Korreferats und für die dabei durchgeführte gründliche Zweitbetreuung danken. Seine fachlichen Hinweise und Ratschläge waren von großer Bedeutung für die Arbeit.

Vor allem möchte ich auch den beteiligten Studentinnen und Studenten danken, die die Realisierung der beschriebenen Komponenten ermöglichten. Besonders den Damen und Herren C. Baader, M. Wagner, M. Wersch, G. Blakowski, S. Engler, A. Oyen, J. Höfler, A. Goos, T. Klenner, A. Geenen und N. Erhardt bin ich sehr dankbar. Die durch ihre Studien- und Diplomarbeiten erbrachten Leistungen übertrafen deutlich die üblichen Maßstäbe; dies spiegelt ihr großes Interesse und ihr hohes Engagement bei der Durchführung der Arbeiten wider.

Darüber hinaus danke ich allen Kolleginnen und Kollegen am Lehrstuhl für die angenehme Zusammenarbeit. Besonders Herrn Prof. Dr. M. Mühlhäuser, der mir nicht nur im Rahmen der Projektarbeit mit Rat und Tat zur Seite stand, bin ich zu großem Dank verpflichtet. Viele Anregungen ergaben sich auch aus der Arbeit in dem von ihm geleiteten Kooperationsprojekt *DOCASE* (Distribution and Objects in Computer-Aided Software Engineering) des Instituts für Telematik an der Universität Karlsruhe und des campusnahen Forschungszentrums CEC der Firma Digital Equipment GmbH in Karlsruhe.

Ich möchte der Projektgruppe und dabei auch den Kolleginnen und Kollegen vom CEC, insbesondere Frau H. Frank und Herrn L. Heuser sowie Herrn Dr. I. Varsek, meinen Dank für die produktive Zusammenarbeit aussprechen. Die über das Projekt aufgebauten internationalen Kontakte gaben wichtige Impulse, die unter anderem auch zur Veröffentlichung der zentralen Konzepte und Ergebnisse der Arbeit führten. Die in diesem Rahmen entstandenen wissenschaftlichen Tagungsbeiträge [SCH89I], [SCH89II], [SCH89III], [HSF88], [HSM88], [MSH88] und [SHM89] beleuchteten zusätzlich das weitere Umfeld der Arbeit, das sich aus der Integration in das Projekt *DOCASE* ergab.

Schließlich möchte ich besonders Herrn Prof. Dr. O. Drobniak danken, der die Arbeit im vorhergehenden Kooperationsprojekt *HECTOR* in der ersten Zeit meiner Institutstätigkeit betreut hat. In diesem Zeitraum konnte ich sehr wichtige praktische Erfahrungen sammeln.

Nicht zuletzt sei allen Freunden und Bekannten für viele gemeinsame Stunden gedankt, die oft ein guter Ausgleich für die manchmal allzu fachlich orientierten Tage und Nächte waren.

Inhaltsverzeichnis

1	Einleitung und Problemstellung	1
1.1	Motivation	1
1.2	Problemstellung	1
1.3	Gliederung der Arbeit	3
2	Grundlagen	4
2.1	Verteilte Anwendungen	4
2.1.1	Übersicht	4
2.1.2	Gegenstand der Verteilung	5
2.1.3	Zielsetzungen	6
2.1.4	Eigenschaften und Probleme	6
2.2	Ansätze zur verteilten Programmierung	8
2.2.1	Betriebssystemansatz	8
2.2.2	Nachrichtenaustausch mit Sprachintegration	10
2.2.3	Remote Procedure Call	10
2.2.4	Objektorientierte Kommunikationsmechanismen	11
2.2.5	Verteilte Programmiersprachen	12
2.2.6	Vergleich mit dem geforderten Modell	13
2.3	Verteilte objektorientierte Ansätze	14
2.3.1	Grundlagen des objektorientierten Ansatzes	14
2.3.2	Verteilte Erweiterungen	18
2.4	Objektmobilität	23
2.4.1	Grundbegriffe	23
2.4.2	Realisierungstechnische Probleme	23
2.4.3	Objektmobilität in einem existierenden System	25
2.4.4	Abgrenzung zur Prozeßmigration	27
2.5	Verteilte Konfigurationsverwaltung	29
2.5.1	Anforderungen	30
2.5.2	Existierende Ansätze	31
2.5.3	Bewertung	32
2.6	Modellbildung	32
2.6.1	Basismodell	32
2.6.2	Architekturmodell	35

3	Steuerung von Objektmigrationen	37
3.1	Grundlagen und Übersicht	37
3.1.1	Klassifikation von Migrationsverfahren	37
3.1.2	Anforderungen an das Verfahren	39
3.1.3	Übersicht und Einordnung in das Klassifikationsmodell	40
3.2	Basiskonstrukte	40
3.2.1	Kollokationsdefinitionen	41
3.2.2	Kollokationen	44
3.3	Semantische Eigenschaften	45
3.3.1	Laufzeitbehandlung von Kollokationen	45
3.3.2	Prioritätsstufen von Kollokationen	48
3.3.3	Bewertungsproblematik	49
3.3.4	Konfliktbehandlung für Kollokationen	54
3.3.5	Abbildung existierender Konstrukte	55
3.4	Kollokationsverwaltung	57
3.4.1	Anforderungen	57
3.4.2	Ein Verfahren zur Kollokationsverwaltung	57
3.4.3	Verfahrensvarianten	60
3.4.4	Verfahrensbewertung	62
3.5	Generierung von Kollokationsdefinitionen	66
3.5.1	Übersicht	66
3.5.2	Phase 1: Vorgabe und Erzeugung	68
3.5.3	Phase 2: Verteilungsunabhängige Analyse	70
3.5.4	Phase 3: Verteilungsabhängige Analyse	71
3.6	Zusammenfassung	71
4	Verwaltung der Anwendungskonfiguration	74
4.1	Grundlagen	74
4.1.1	Definitionen	74
4.1.2	Klassifikation	76
4.1.3	Anforderungen an die Konfigurationssprache	80
4.1.4	Anforderungen an das Laufzeitsystem	81
4.1.5	Modell der entwickelten Konfigurationsverwaltung	81
4.2	Konfigurationssprache	83
4.2.1	Systemkonfiguration	84
4.2.2	Anwendungskonfiguration	86
4.2.3	Änderungsnotation	93
4.3	Laufzeitebene	94
4.3.1	Erweiterungen des Basismodells	94
4.3.2	Ankopplung an die Verwaltungsebene	97
4.4	Konfigurationsverwaltung	98
4.4.1	Initiale Konfiguration	98
4.4.2	Konfigurationsänderungen	101
4.5	Zusammenfassung	104

5 Realisierung und Einsatz	106
5.1 Synthese der Konzepte	106
5.2 Gesamtarchitektur	107
5.3 Beschreibung der Teilkomponenten	109
5.3.1 Basissystem	109
5.3.2 Realisierung des Basismodells	112
5.3.3 Kollokationsverwaltung	114
5.3.4 Monitorkomponente	115
5.3.5 Konfigurationsbeschreibungskomponente	116
5.3.6 Konfigurationsverwaltungskomponente	117
5.4 Zusammenwirken der Teilkomponenten	118
5.5 Methodischer Ablauf der Anwendungsentwicklung	118
5.5.1 Verteilungsunabhängige Phase	120
5.5.2 Verteilungsabhängige Phase	121
5.6 Implementierung	121
6 Zusammenfassung und Ausblick	123
6.1 Ergebnisse der Arbeit	123
6.2 Weiterführende Arbeiten	124
Anhang	126
A Basissystem: Spracherweiterungen von C++	126
A.1 Grundlagen und Überblick	126
A.2 Beschreibung der Spracherweiterung	127
B Konfigurationssprache	130
B.1 Syntax der Konfigurationsbeschreibungsnotation	130
B.2 Syntax der Konfigurationsänderungsnotation	138
C Beispiel einer Konfigurationsbeschreibung	145
C.1 Übersicht	145
C.2 Konfigurationsbeschreibung	148
Literaturverzeichnis	160
Verzeichnis der Abbildungen	173
Verzeichnis der Tabellen	174