

Programm der JIT'98 Konferenz - Liste aller Vorträge

Diese Seite enthält nicht das Programm der 2.DJEK, welche ebenfalls Teil der JAVADAYS '98 ist.
Eine für den [Ausdruck](#) gedachte PDF-Fassung dieser Seite finden Sie [hier](#).

Verzeichnis der Vorträge:

[Keynote: Jini - der gute Geist aus dem Netzwerk](#)

[Invited Talk: Kaffee - ein Java für alle?](#)

[Great perspectives for developers - How Sun will professionally support Your work](#)

Sitzung: Sprachkritik:

[Kapselung und Methodenbindung: Javas Designprobleme und ihre Korrektur](#)

[Eine Evaluierung des Java JDK 1.2 Collections Framework aus Sicht der Softwaretechnik](#)

Sitzung: Sicherheits- und Systemfragen:

[Transitiver Schutz in Java durch Sicherheitsmetaobjekte](#)

[Java RMI, CORBA und Firewalls](#)

Sitzung: Middleware, CORBA und verteilte Systeme 1:

[Realisierung einer Client/Server-Anwendung mit CORBA und Java unter Berücksichtigung bestehender C++-Komponenten](#)

[Reflection in Java, CORBA und JacORB](#)

[Java, XML und Servlets zur Integration datenbankbasierter Applikationen im Web](#)

Sitzung: Eingebettete Systeme:

[Einsatz von Java-Komponenten in verteilten Embedded Systems](#)

[Java Komponenten für ein verteiltes Echtzeitsystem](#)

Sitzung: Spracherweiterung 1:

[JavaParty - portables, paralleles und verteiltes Programmieren in Java](#)

[JavaSet - eine Spracherweiterung von Java um persistente Mengen](#)

[JaWA: Java with Assertions](#)

Sitzung: Java in der Anwendung 1:

[SurfBorD - Systemunabhängig realisierte flexible Bedienoberfläche für relationale Datenbanken](#)

[Ein RMI-basierter Repository-Server zur Synchronisation der Software-Entwicklung in verteilten Unternehmen](#)

[Java-basierte Konfiguration und Steuerung des CORBA-verteilten CFD-Simulationssystems TENT](#)

Sitzung: Java in der Lehre:

[Unterstützung der Lehre durch Visualisierung von wissensbasierten Suchalgorithmen mit Java](#)

[Java in der Ausbildung](#)

Sitzung: Interpretation und Compilation von Java:

[Analysis of the Java Class File Format](#)

[Statische Analyse von Bibliotheken als Grundlage dynamischer Optimierung](#)

Sitzung: Softwareentwurf und Werkzeuge:

[Realisierung von verteilten Editoren in Java auf Basis eines aktiven Repositories](#)

[Common Logging Interface - Ein System zum Sammeln und Weiterverarbeiten von Debugnachrichten in verteilten Umgebungen](#)

[Flexibilität durch kombinierte Design Pattern](#)

Sitzung: Middleware, CORBA und verteilte Systeme 2:

[Entwicklung einer abstrakten Speicherkomponente für eine verteilte heterogene dynamische Infrastruktur in Java/CORBA](#)

[Flexible Vermittlung von skalierbaren Dienstobjekten in verteilten Systemen](#)

[ANT - Active Node Technology](#)

Sitzung: Java in der Anwendung 2:

[VConfig - Eine Java basierte Oberfläche zur Systemadministration](#)

[CORBA-basiertes Workflow-Management zur Bearbeitung von Kundenanfragen aus dem Internet](#)

Sitzung: Java in der Anwendung 3:

[Erfahrungen mit dem Einsatz von Java und CORBA in der Entwicklung einer leistungsstarken Customer Care-Anwendung für die Telekommunikationsindustrie](#)

[Interaktive Animationen und Visualisierungen - eine neue Qualität und "Spielwiese" für die Mechanik](#)

[OpenCard Application Framework](#)

Sitzung: Spracherweiterung 2:

[Poor Man's Genericity for Java](#)

[Java - formal fundiert](#)



Keynote: Jini - der gute Geist aus dem Netzwerk

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

Hellmuth Broda
(Chief Technologist Europe, Sun Microsystems GmbH)

09:45 - 10:45 am Donnerstag, 12. November 1998 (Track 3 (JIT'98))

- [Ansicht der Vortragsfolien \[Format ppt\] \(1002K\)](#) [\[Anderes Fenster nach vorne\]](#)

[Zurück zum Verzeichnis der Vorträge](#)



Invited Talk: Kaffee - ein Java für alle?

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

Peter Mehlitz

(TransVirtual Technologies Inc. (USA))

10:45 - 11:30 am Donnerstag, 12. November 1998 (Track 3 (JIT'98))

- [Ansicht der Vortragsfolien \[Format html\] \(1,6K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

This talk will conclude as follows:

Java is not just a language, it is a "middleware OS".

Kaffe is not "just" a VM, it is a framework for tailored Java systems.

Its desktop future: public sources (GPL), efficiency.

Its embedded systems future: portability, scalability.

[Zurück zum Verzeichnis der Vorträge](#)



Great perspectives for developers - How Sun will professionally support Your work

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

Brian Gentile

(Vice-President Sun Developer Relations)

11:30 - 12:15 am Donnerstag, 12. November 1998 (Track 3 (JIT'98))

- [Ansicht der Vortragsfolien \[Format pdf\] \(1,0M\)](#) [\[Anderes Fenster nach vorne\]](#)

[Zurück zum Verzeichnis der Vorträge](#)



Kapselung und Methodenbindung: Javas Designprobleme und ihre Korrektur

Peter Müller, Arnd Poetzsch-Heffter

(Fernuniversität Hagen)

[Referent: Peter Müller]

13:30 - 14:00 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: Sprachkritik)

- [Ansicht der Vortragsfolien \[Format pdf\] \(24K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

In Java bestehen enge Wechselwirkungen zwischen den Konstrukten für die Kapselung von Implementierungsteilen und den Regeln für die Bindung von Methoden. Anhand zweier Beispiele zeigen wir, daß die mangelnde Orthogonalität der beiden Sprachkonzepte zu Fehlern in der Sprachspezifikation und Implementierung von Java führte. Der Aufsatz diskutiert mögliche Lösungen und liefert eine korrigierte und konzeptionell vereinfachte Fassung des Methodenauswahlalgorithmus von Java. Darüber hinaus präsentiert er ein erweitertes Konzept zur Vergabe von Zugriffsrechten. Insgesamt wird damit die Orthogonalität der beteiligten Sprachkonzepte wesentlich gesteigert.

[Zurück zum Verzeichnis der Vorträge](#)



Eine Evaluierung des Java JDK 1.2 Collections Framework aus Sicht der Softwaretechnik

Mark Evered, Gisela Menger

(Universität Ulm)

[Referent: Mark Evered]

Abstrakt:

Sammlungen von Daten spielen eine wichtige Rolle bei fast jeder Art der Programmierung. Aus diesem Grunde bieten praktisch benutzte objektorientierte Sprachen Standard-Bibliotheken an, die Klassen für die Verwaltung von Objektsammlungen enthalten. Diese Bibliotheken sind besonders dafür kritisiert worden, daß sie softwaretechnische Prinzipien wie Geheimnisprinzip und Orthogonalität nicht hinreichend berücksichtigen. In diesem Beitrag evaluieren wir das neue 'Collections Framework' des 'Java Development Kit' 1.2, das durch die Definition der 'Core Collection Interfaces' einige Vorteile hinsichtlich der Verständlichkeit und Erweiterbarkeit solcher Bibliotheken bietet. Wir diskutieren sowohl Schwächen des Framework, die auf einer inkonsequenten Anwendung von Softwaretechnikprinzipien beruhen, als auch Probleme, die aus der Definition der Sprache Java selbst resultieren.

[Zurück zum Verzeichnis der Vorträge](#)



Transitiver Schutz in Java durch Sicherheitsmetaobjekte

Thomas Riechmann, Franz J. Hauck, Jürgen Kleinöder
(Universität Erlangen-Nürnberg)

[Referent: Thomas Riechmann]

14:30 - 15:00 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: Sicherheits- und Systemfragen)

- [Ansicht der Vortragsfolien \[Format pdf\] \(51K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Java hat sich wegen der einfachen, architekturunabhängigen und sicheren Verteilbarkeit von Programmen bereits als Plattform für verteilte, objektorientierte Applikationen, bei denen das Thema Sicherheit wichtig ist, etabliert. Das Sicherheitsmodell von Java basiert auf Objektorientierung: Ein Objekt kann nur angesprochen werden, wenn man eine Referenz auf dieses besitzt, und auch dann nur über Methoden, die in seiner Schnittstelle entsprechend deklariert sind (Capability-Implementierung). Zusätzlich können Klassen auch selbst Zugriffsschutz abhängig vom Aufrufer implementieren, um z.B. Datei-Objekte nur von bestimmten Klassen aus zugreifbar zu machen (Zugriffslisten-Implementierung). Bei Java ist die Sicherheitskonfiguration nicht von der Implementation der Applikationsklassen getrennt, sondern muß in die Applikationsklassen hineinimplementiert werden.

Das Hauptproblem bei Capabilities ist die Verbreitungskontrolle: Da bei objektorientierten Systemen oft Objektreferenzen übergeben werden, ist es schwer, die Kontrolle zu behalten, wer auf welche Referenz zugreifen kann. Zugriffslisten haben andererseits den Nachteil, daß es einer böswilligen Applikation möglich sein kann, fremden, privilegierten Subjekten (Programmteilen, Domains) Objektreferenzen unterzuschieben, an denen diese dann ungewollt Aufrufe mit ihren hohen Privilegien durchführen (Unix s-Bit Problem).

Wir stellen ein Sicherheitsmodell vor, das diese Probleme löst. Durch Einführung von Sicherheitsmetaobjekten trennen wir die Sicherheitsstrategie von der Implementierung der Applikationsklassen. Ein Sicherheitsmetaobjekt kann an eine Objektreferenz geheftet werden und schützt diese Referenz, indem es Zugriffe überwacht und entscheidet, welche Zugriffe zulässig sind (Implementation von konfigurierbaren Capabilities). Außerdem kann es Parameter- und Rückgabewerte überwachen und schützen, um die versehentliche Herausgabe von Referenzen zu verhindern bzw. herausgegebene Referenzen mit Zugriffsbeschränkungen zu versehen (Implementation von transitiven Capabilities). Trotz starker Interaktion und Objektreferenzaustausch zwischen Programmteilen kann die Sicherheitskonfiguration orthogonal zur Applikationsimplementation erfolgen und auf sehr hohem Abstraktionsniveau festgelegt werden.

[Zurück zum Verzeichnis der Vorträge](#)



Java RMI, CORBA und Firewalls

Rainer Falk
(TU München)

[Referent: Rainer Falk]

15:00 - 15:30 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: Sicherheits- und Systemfragen)

- [Ansicht der Vortragsfolien \[Format pdf\] \(115K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Wenn Verteilte Objekte über Firewallgrenzen hinweg angesprochen werden sollen, treten Probleme auf. In diesem Beitrag wird die Problematik erläutert und es werden Lösungsansätze vorgestellt und verglichen.

[Zurück zum Verzeichnis der Vorträge](#)



Realisierung einer Client/Server-Anwendung mit CORBA und Java unter Berücksichtigung bestehender C++-Komponenten

Klaus Beschorner
(iT media Consult GmbH, Stuttgart)

Wolfgang Rosenstiel
(Universität Tübingen)

[Referent: Klaus Beschorner]

16:30 - 17:00 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: *Middleware, CORBA und verteilte Systeme 1*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(736K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Dieses Papier stellt einige Vorgehensweisen zur Realisierung einer Client/Server-Anwendung mit CORBA und Java vor. Dazu gehört die Programmierung robuster und benutzerfreundlicher Java-Anwendungen in Form von Applets und Applikationen sowie die Wiederverwendung von bestehendem Code in Java- und CORBA-Objekten. Alle vorgestellten Ansätze sind erfolgreich im Rahmen von [Bes98] in einer prototypisch realisierten Internet/Intranet-Anwendung umgesetzt worden, die ebenfalls vorgestellt wird. Die gefundenen Vorgehensweisen sollen den Entwicklungsprozeß von anderen Anwendungen ähnlicher Natur vereinfachen.

[Zurück zum Verzeichnis der Vorträge](#)

Reflection in Java, CORBA und JacORB

Gerald Brose
(Freie Universität Berlin)

[Referent: Gerald Brose]

17:00 - 17:30 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: *Middleware, CORBA und verteilte Systeme 1*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(201K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Sowohl in Java wie in CORBA-Umgebungen können Programme auf Meta-Objekte zugreifen, die bestimmte Eigenschaften des Systems selbst repräsentieren. Wir beschreiben Ähnlichkeiten und Unterschiede solcher Reflection-Konzepte in Java und CORBA sowie eine Implementierung eines Interface Repositorys für die CORBA-Plattform JacORB, die mit Hilfe von Java-Reflection realisiert wurde.

Stichworte: Reflection, CORBA, Java, JacORB, Interface Repository.

[Zurück zum Verzeichnis der Vorträge](#)

Java, XML und Servlets zur Integration datenbankbasierter Applikationen im Web

Albrecht Schmidt, Günther Specht
(TU München)

[Referent: Albrecht Schmidt]

17:30 - 18:00 am Donnerstag, 12. November 1998 (Track 3 (JIT'98) in Sitzung: *Middleware, CORBA und verteilte Systeme 1*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(72K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Am Institut für Informatik der Technischen Universität München werden zwei bestehende Technologien, digitale Bibliothekssysteme und multimediale Datenbanken, in einem Web-basierten Kontext integriert, d.h. ein Benutzer soll letztendlich über einen der gängigen WWW-Browser auf die schon vorhandenen digitalen Bibliotheken als auch auf multimediale Informationen zugreifen können. Eine besondere Rolle spielt dabei die Verwendung von Java als Implementationssprachen und XML als (zunächst internem) Datenformat zum Zusammentragen der Information aus den verschiedenen Datenbanken.

[Zurück zum Verzeichnis der Vorträge](#)

Keynote: Jini - der gute Geist aus dem Netzwerk

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

Hellmuth Broda
(Chief Technologist Europe, Sun Microsystems GmbH)

09:45 - 10:45 am Donnerstag, 12. November 1998 (Track 4 (JIT'98))

- [Ansicht der Vortragsfolien \[Format ppt\] \(1002K\)](#) [\[Anderes Fenster nach vorne\]](#)

[Zurück zum Verzeichnis der Vorträge](#)



Invited Talk: Kaffee - ein Java für alle?

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

[Peter Mehlitz](#)

(TransVirtual Technologies Inc. (USA))

10:45 - 11:30 am Donnerstag, 12. November 1998 (Track 4 (JIT'98))

- [Ansicht der Vortragsfolien \[Format pdf\] \(491K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

This talk will conclude as follows:

Java is not just a language, it is a "middleware OS".

Kaffe is not "just" a VM, it is a framework for tailored Java systems.

Its desktop future: public sources (GPL), efficiency.

Its embedded systems future: portability, scalability.

[Zurück zum Verzeichnis der Vorträge](#)



Great perspectives for developers - How Sun will professionally support Your work

[Gemeinsam mit den Tracks 1 und 2 der DJEK]

Brian Gentile

(Vice-President Sun Developer Relations)

11:30 - 12:15 am Donnerstag, 12. November 1998 (Track 4 (JIT'98))

- [Ansicht der Vortragsfolien \[Format pdf\] \(1,0M\)](#) [\[Anderes Fenster nach vorne\]](#)

[Zurück zum Verzeichnis der Vorträge](#)



Einsatz von Java-Komponenten in verteilten Embedded Systems

[Uwe Rastofer](#), [Ulrich Gall](#), [Frank Schinkmann](#)

(Universität Erlangen-Nürnberg)

[Bernd Hindel](#)

(3SOFT GmbH)

[Referent: Uwe Rastofer]

12:25 - 12:55 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: Eingebettete Systeme)

- [Ansicht der Vortragsfolien \[Format pdf\] \(483K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Die zunehmende Vernetzung von Rechenanlagen dringt immer weiter auch in den Bereich von Embedded Systems vor. Um eine Interaktion zwischen verschiedenen vernetzten Geräten zu ermöglichen, sind allerdings klare Softwareschnittstellen erforderlich.

Die vorliegende Arbeit beschreibt, wie Softwarekomponentenarchitekturen, wie zum Beispiel Java-Beans, verwendet werden können, um sowohl Interaktion zwischen Geräten als auch die Erstellung von Benutzeroberflächen für das Bedienen und Beobachten aus der Ferne zu ermöglichen. Dabei können solche Benutzeroberflächen sowie die Interaktionssemantik zwischen Geräten visuell mit JavaBeans-Entwicklungsumgebungen implementiert werden.

Es werden zwei Ansätze für die Verwendung von JavaBeans vorgestellt. Beim Stellvertreter-basierten Modell werden Geräte lokal durch in Java implementierte Proxy-Komponenten repräsentiert, die über ein beliebiges Protokoll mit der proprietären Software im Gerät kommunizieren. Beim verteilten Ansatz findet die Interaktion in einer homogenen Java-Umgebung über Java RMI statt.



Java Komponenten für ein verteiltes Echtzeitsystem

Jan Richling, Janek Schwarz, Andreas Polze
(Humboldt-Universität zu Berlin)

[Referent: Jan Richling]

12:55 - 13:25 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Eingebettete Systeme*)

Abstrakt:

Die Programmiersprache Java mit ihrem architekturunabhängigem Abarbeitungsansatz bietet viele Vorteile, wie Sicherheit in verteilten Umgebungen, Wiederverwendbarkeit von Code und Portabilität. Diese Charakteristika führen zum Einsatz von Java in vielen neuartigen Umgebungen.

Obwohl Java viele Vorteile bietet, ist der Einsatz der Sprache in Fällen kritisch, in denen Echtzeitverhalten gefordert ist. Die existierenden Java-Ausführungsumgebungen und Sprachspezifikationen reichen für das Gewähren von Echtzeitgarantien nicht aus. Andererseits existieren auch in einem Echtzeitsystem Komponenten, an deren zeitliches Verhalten keine strengen Forderungen gestellt werden und für deren Implementation sich Java anbietet.

Unsere Forschung richtet sich auf Mechanismen zur Verknüpfung von Java-Komponenten mit (existierenden) Echtzeitsystemen. In diesem Artikel beschreiben wir die Verwendung des "Composite Objects"-Ansatzes zur zeitlichen Entkopplung und Interoperabilität zwischen Java-Komponenten und dem Echtzeitsystem. Wir diskutieren die Erweiterung eines bestehenden Echtzeitsystems um offene Schnittstellen.

Die "Unstoppable Robots" stellen eine verteilte, responsive (echtzeitfähige, fehlertolerante) Demo-Applikation auf Basis des Betriebssystems Mach dar. Wir demonstrieren die Erweiterung der Applikation um inJava geschriebene Komponenten. Diese Komponenten, eine graphische Benutzerschnittstelle und ein Controller, unterliegen unterschiedlich strengen Echtzeitanforderungen. Wir evaluieren die Leistung der erweiterten Demo-Applikation und zeigen die Brauchbarkeit unseres Ansatzes.

[Zurück zum Verzeichnis der Vorträge](#)



JavaParty - portables, paralleles und verteiltes Programmieren in Java

Michael Philippsen, Matthias Zenger, Matthias Jacob
(Universität Karlsruhe)

[Referent: Michael Philippsen]

14:30 - 15:00 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Spracherweiterung 1*)

Abstrakt:

Während Java Threads (Aktivitätsstränge) als geeignetes Sprachmittel für die Programmierung von SMPs (Parallelrechnern mit gemeinsamem Speicher) anbietet, fehlen elegante und ausreichende Sprachmittel für die Programmierung von Parallelrechnern mit verteiltem Speicher (DMPs), also auch für Cluster von Arbeitsplatzrechnern. Die in der Java-Distribution angebotene explizite Socket-Kommunikation und der Aufruf entfernter Methoden (RMI) erfordern bei der Portierung eines für eine SMP-Maschine entwickelten mehrsträngigen Programms auf eine DMP-Maschine erhebliche Programmänderungen und -erweiterungen.

JavaParty behebt diesen Mißstand, ermöglicht ein Java-artiges Programmieren auch von DMPs und Clustern von Arbeitsplatzrechnern und verallgemeinert die Idee des plattformunabhängigen Codes auch für Parallelrechner unterschiedlicher Architekturen. Die Erweiterung beruht auf einem neuen, die Klassendeklaration ergänzenden, Modifikator 'remote' mit dem potentiell entfernt zu realisierende Objekte für den JavaParty-Übersetzer gekennzeichnet werden. Aus dem Attribut wird dann portabler Java-Code sowohl für SMPs als auch DMPs erzeugt, ohne den Programmierer der Komplexität von expliziter Socket-Kommunikation bzw. RMI auszusetzen.

[Zurück zum Verzeichnis der Vorträge](#)



JavaSet - eine Spracherweiterung von Java um persistente Mengen

Markus Schordan, Harald Kosch
(Universität Klagenfurt (A))

[Referent: Markus Schordan]

15:00 - 15:30 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Spracherweiterung 1*)

Abstrakt:

Wir stellen in diesem Beitrag eine Spracherweiterung von Java vor, die es ermöglicht, elegant persistente und transiente Mengen von Objekten zu manipulieren und deklarative Mengenabfragen zu formulieren. Durch diese Spracherweiterung ist der Zugriff auf den

persistenten Datenspeicher für den Programmierer transparent. Die Übersetzung von JavaSet beinhaltet eine effiziente Objektalgebra Optimierung von Mengenoperationen und deren Abbildung auf einen Ausführungsplan.

[Zurück zum Verzeichnis der Vorträge](#)



JaWA: Java with Assertions

Clemens Fischer, Dieter Meemken
(Universität Oldenburg)

[Referent: Clemens Fischer, Dieter Meemken]

15:30 - 16:00 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Spracherweiterung 1*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(107K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Methoden zur Entwicklung von korrekter Software und deren Einführung in die Praxis stellen heute immer noch ein schwieriges Problem dar. Bertrand Meyer [Mey97a] hat unter dem Stichwort 'Programmieren mit Vertrag' einen Vorschlag gemacht und durch Eiffel implementiert, wie man formale Spezifikationsanteile mit Programmcode mischen und zur Laufzeit überprüfen kann. In diesem Paper geben wir einen Überblick über die Sprache JaWA, mit der dieses Konzept auf Java übertragen wurde.

JaWA Programme bestehen aus herkömmlichem Java-Code, der Zusicherungen (Invarianten, Vor- und Nachbedingungen) in Form von Kommentaren enthält. Der JaWA-Präcompiler übersetzt JaWA in Java. Er erzeugt zusätzliche Bedingungen, mit denen die spezifizierten Eigenschaften zur Laufzeit überprüft werden können.

Dabei wird die Java Ausnahmebehandlung ausgenutzt und um Eiffel ähnliche 'rescue' und 'retry' Anweisungen ergänzt. Es werden konfigurierbare Fehlermeldungen oder Warnungen mit den korrekten Zeilennummern erzeugt.

[Zurück zum Verzeichnis der Vorträge](#)



SurfBorD - Systemunabhängig realisierte flexible Bedienoberfläche für relationale Datenbanken

Dirk Fischer, Heike Utermann
(Technische Universität München)

[Referent: Heike Utermann]

17:00 - 17:30 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Java in der Anwendung 1*)

Abstrakt:

Der Wechsel auf ein anderes DBMS (Datenbank-Management-System) oder das Erweitern einer Datenbank ist in der Regel mit viel Nachholarbeit verbunden, da auch die verschiedenen Anwendungen angepaßt werden müssen, die auf die Datenbank zugreifen. Mit SurfBorD wurde eine Anwendung in Java entwickelt, die es ermöglicht, beliebige Datenbanken abzufragen oder zu bearbeiten. Dabei ist es gleichgültig, von welchem Hersteller das DBMS stammt und auf welcher Plattform SurfBorD eingesetzt wird. Durch dynamisch ladbare Komponenten kann SurfBorD beliebig um neue Analyse- und Manipulationswerkzeuge erweitert werden.

[Zurück zum Verzeichnis der Vorträge](#)



Ein RMI-basierter Repository-Server zur Synchronisation der Software-Entwicklung in verteilten Unternehmen

E. Ulrich Kriegel, Dirk Kurzmann
(Fraunhofer Institut für Software- und Systemtechnik Berlin)

[Referent: E. Ulrich Kriegel]

17:30 - 18:00 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: *Java in der Anwendung 1*)

Abstrakt:

Im Rahmen eines Verbundprojektes zur Verbesserung der Softwareproduktion in kleinen und mittelständischen Unternehmen wurde am Fraunhofer ISST ein Repository-Server als Aufsatz zu bestehenden Softwareentwicklungsumgebungen entwickelt. Über diesen Server können mit Hilfe spezieller Werkzeuge Entwicklungsdokumente zwischen verteilten Entwicklern ausgetauscht werden. Der vorliegende Bericht beschreibt den Entwurf und die Implementation des Repository-Servers sowie der Werkzeuge, die auf ihn zugreifen.

[Zurück zum Verzeichnis der Vorträge](#)



Java-basierte Konfiguration und Steuerung des CORBA-verteilten CFD-Simulationssystems TENT

Thomas Breitfeld

(GMD Forschungszentrum Sankt Augustin)

[Referent: Thomas Breitfeld]

18:00 - 18:30 am Donnerstag, 12. November 1998 (Track 4 (JIT'98) in Sitzung: Java in der Anwendung 1)

- [Ansicht der Vortragsfolien \[Format gif\] \(449K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Dieser Artikel beschreibt die Benutzung von Java in einem CFD-Simulationssystem, welches die gegenwärtig bestehenden Probleme bezüglich der Kopplung und Konfiguration der in einem solchen System notwendigen Komponenten überwindet. Dieses System namens TENT dient der Unterstützung des Entwurfs und Designs von Flugzeugen und deren Teilen.

TENT basiert auf CORBA und es benutzt Java, um die Steuerung und Konfiguration der an einer Simulation beteiligten Komponenten zu realisieren. Neben einer Einführung in die Problematik dieser Simulationssysteme wird im einzelnen erläutert, wie Java für die Kopplung und Überwachung der Komponenten und für ein generisches Konfigurations-GUI benutzt wird. Es wird gezeigt, daß sich Java hervorragend für diese Aufgaben eignet und Vorteile gegenüber anderen Implementierungssprachen an diesen Stellen aufweist. Damit wird die vorteilhafte Benutzung von Java in einem industriellen und praxisrelevanten Anwendungsszenario unter Beweis gestellt.

[Zurück zum Verzeichnis der Vorträge](#)



Unterstützung der Lehre durch Visualisierung von wissensbasierten Suchalgorithmen mit Java

Jörg Denzinger, Bernd Löchner, Sebastian Scheffler

(Universität Kaiserslautern)

[Referent: Bernd Löchner]

09:00 - 09:30 am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: Java in der Lehre)

Abstrakt:

Dieser Bericht enthält unsere Erfahrungen mit Java bei der Entwicklung und Implementierung eines wissensbasierten Suchsystems. Der Schwerpunkt lag auf der Visualisierung der Vorgänge und Entscheidungen im System, um es Studenten zur Nachbereitung der entsprechenden Vorlesung zur Verfügung zu stellen. Das System wurde im Rahmen eines Praktikums implementiert, und Java hat sich sowohl bei der Entwicklung als auch im Hinblick auf die Visualisierung der Vorgänge als sinnvolle Wahl als Implementierungssprache erwiesen. Überraschend war auch die gute Effizienz des Systems.

[Zurück zum Verzeichnis der Vorträge](#)



Java in der Ausbildung

Robert Tolksdorf

(Technische Universität Berlin)

Wilhelm Weisweber

(Technische Universität Berlin)

[Referent: Robert Tolksdorf]

09:30 - 10:00 am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: Java in der Lehre)

- [Ansicht der Vortragsfolien \[Format pdf\] \(718K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Java spielt in der Informatik Aus-, Fort- und Weiterbildung eine immer stärkere Rolle. Dieses Papier faßt die positiven Erfahrungen der Autoren mit der Entscheidung für den Einsatz von Java in der Lehre und deren Durchführung zusammen. Die Hauptvorteile der Sprache für die Ausbildung liegen in der Modularisierbarkeit von Lerninhalten, der Skalierbarkeit der Sprache über Anforderungsniveaus und ihrer Relevanz für die Berufspraxis. Nachteile wie schlechte Lehrbuchsituation oder unintuitive Entwicklungsumgebungen wiegen nicht so schwer, daß sie einer Entscheidung für Java in der Ausbildung entgegenstünden.

[Zurück zum Verzeichnis der Vorträge](#)



Analysis of the Java Class File Format

Denis N. Antonioli, Markus Pilz
(Universität Zürich (CH))

[Referent: Denis N. Antonioli]

10:10 - 10:40 am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: *Interpretation und Compilation von Java*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(63K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

The goal of this work is to study the properties of the Java class file format, and particularly to answer these questions:

1. It is often claimed that programs written in Java are compiled to a compact and efficient format. What are the sizes of typical class files?
2. The class file is structured around several parts of variable length. How do they add to the size of the file?
3. The Java Virtual Machine defines a rich instruction set of over 200 instructions. Are all these instructions used? How often?

The paper is thus structured as follows: Section 1.0 describes the format of the class file. Section 2.0 describes the experiment and section 3.0 discusses three important results. Finally, section 4.0 summarizes the findings and concludes the article.

[Zurück zum Verzeichnis der Vorträge](#)



Statische Analyse von Bibliotheken als Grundlage dynamischer Optimierung

Michael Thies, Uwe Kastens
(Universität-GH Paderborn)

[Referent: Michael Thies]

10:40 - 11:10 am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: *Interpretation und Compilation von Java*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(208K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Dieser Beitrag schlägt einen neuartigen Ansatz zur optimierten Ausführung von Java-Bytecode vor, der die dynamische Optimierung eines Java-Programms durch statische Programmanalyse vorbereitet. Die Analyseinformation wird unabhängig von der Programmausführung bezogen auf alle Klassendateien einer Softwarebibliothek ermittelt und gespeichert. Zur Laufzeit unterstützt die an den Bibliotheksschnittstellen komponierte Information insbesondere Optimierungen, die zusätzlich dynamische Programmeigenschaften ausnutzen.

[Zurück zum Verzeichnis der Vorträge](#)



Realisierung von verteilten Editoren in Java auf Basis eines aktiven Repositories

Udo Kelter, Marc Monecke, Dirk Platz
(Universität Siegen)

[Referent: Marc Monecke]

11:30 - 12:00 am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: *Softwareentwurf und Werkzeuge*)

Abstrakt:

Dieses Papier behandelt die Konstruktion verteilter Editier-Umgebungen an Beispiel von Software-Entwicklungsumgebungen. Die vorgeschlagene Architektur basiert auf einem aktiven Repository, das Änderungen in den Daten netzwerkweit und sichtenübergreifend propagiert und diverse andere Dienste (Transaktionen, Zugriffskontrollen) anbietet. Die textuellen und/oder graphischen Benutzungsschnittstellen sind in Java geschrieben und daher (z.B. als Applets) leicht verteilbar. Sie kommunizieren über einen Java-API-Server indirekt mit dem Repository; dessen Aufgabe ist es, die Java-Anwendungen an das Repository anzubinden. Wir beschreiben diverse Details dieser Architektur, insb. das Prozeßkonzept, das die parallele Ausführung von Java-Anwendungen ermöglicht, die netzwerkweite Notifizierung sowie die relevanten Java-Klassen für den Zugriff zum Repository.

[Zurück zum Verzeichnis der Vorträge](#)



Common Logging Interface - Ein System zum Sammeln und Weiterverarbeiten von Debugnachrichten in verteilten Umgebungen

Raimar Falke, Michael Peter, Achim Gratz, Rainer G. Spallek

- [Ansicht der Vortragsfolien \[Format pdf\] \(47K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Mit CLI wird ein in Java implementiertes System vorgestellt, welches mit Hilfe eines minimalen API das zeitnahe Beobachten von mehrfädigen und verteilten Anwendungen auf der Basis von Debugnachrichten gestattet. Die Anwendung muß dazu nicht unter einem in-process Debugger laufen. Anwendung und Beobachter sind soweit als möglich entkoppelt. Die Selektion der Debugnachrichten wird mittels mehrstufiger Filterung anhand verschiedener Kriterien zur Laufzeit vorgenommen. Die Weiterleitung der Nachrichten erfolgt durch eine Transport-Schicht, die je nach vorliegender Umgebung angepaßt und ausgetauscht werden kann.

[Zurück zum Verzeichnis der Vorträge](#)



Flexibilität durch kombinierte Design Pattern

Carsten Weise

(BRICS, Universität Aalborg (DK))

[Referent: Carsten Weise]

- [Ansicht der Vortragsfolien \[Format pdf\] \(256K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

In graphischen Benutzerschnittstellen (aber auch in vielen anderen Anwendungen) tritt häufig das Problem auf, daß dasselbe Objekt auf verschiedene Arten dargestellt werden soll. Dies führt dazu, daß konzeptuell gleiche Klassen in verschiedenen Versionen implementiert werden. Die restliche Anwendung sollte davon unbeeindruckt bleiben. Anhand einer einfachen Fallstudie wird vorgeführt, wie durch geeignete Kombination von Design Pattern diese Problematik gelöst werden kann.

[Zurück zum Verzeichnis der Vorträge](#)



Entwicklung einer abstrakten Speicherkomponente für eine verteilte heterogene dynamische Infrastruktur in Java/CORBA

Torsten Fink, Michael M. Gutzmann, Torsten Wolf, Werner Erhard
(Universität Jena)

[Referent: Torsten Fink]

- [Ansicht der Vortragsfolien \[Format pdf\] \(52K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Da die Entwicklung von Applikationen für verteilte heterogene dynamische Systeme komplexe Probleme birgt, existieren vielfältige Werkzeuge, die eine Infrastruktur zur Unterstützung des Entwicklers bilden. In dieser Arbeit wird eine Speicherkomponente als Bestandteil einer umfassenden Infrastruktur vorgestellt, die eine ortstransparente Datenhaltung mit hoher Abstraktion ermöglicht. Zur Steigerung der Zugriffsgeschwindigkeit können lokale Kopien transparent erzeugt werden. Um einen schnellen Datentransport in heterogenen Netzwerken zu gewährleisten, wird zur Laufzeit aus einer Menge unterstützter Übertragungsprotokolle ein jeweils geeignetes gewählt.

Diese Speicherkomponente wurde unter Einsatz einer Kombination von Java und CORBA implementiert. Messungen an einem exemplarischen Testsystem offenbaren den auftretenden Organisationsaufwand.

[Zurück zum Verzeichnis der Vorträge](#)



Flexible Vermittlung von skalierbaren Dienstobjekten in verteilten Systemen

Arnd Grosse, Stefan Dolk, Rainer Ruggaber

(Universität Karlsruhe)

[Referent: Rainer Ruggaber]

- [Ansicht der Vortragsfolien \[Format pdf\] \(139K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Die zunehmende arbeitsteilige Integration zwischen weiträumig verteilten Anwendungen bei gleichzeitiger Partitionierung zuvor monolithischer Anwendungen in feingranularere (Dienst-)Objekte, erfordert eine systemtechnische Unterstützung, um zur Laufzeit flexibel das jeweils benötigte Dienstobjekt zu ermitteln und darauf zuzugreifen. Diese Flexibilität wird in verteilten Systemen durch eine Vermittlungsschicht auf Basis verteilter Trading-Dienste bereitgestellt. Der Nachteil dieses Ansatzes liegt in seiner Beschränkung auf die Vermittlung von Objektreferenzen. In dem folgenden Artikel wird ein in Java realisierter, erweiterter Trading-Ansatz vorgestellt, der neben der Referenz zudem den Objektzustand und die Implementierung vermittelt und damit eine flexible Erstellung verteilter Anwendungen unterstützt.

[Zurück zum Verzeichnis der Vorträge](#)



ANT - Active Node Technology

Bernhard Zwantschko, Christian Gütl
(Technische Universität Graz (A))

ausg. - cancel am Freitag, 13. November 1998 (Track 3 (JIT'98) in Sitzung: *Middleware, CORBA und verteilte Systeme 2*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(159K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Die vorliegende Arbeit stellt in kurzen Umrissen die wesentlichsten Merkmale der Active Node Technology (ANT) vor. ANT dient der transparenten Integration verschiedenster Netzwerkkomponenten und Services durch Generieren eines virtuellen Adreßraumes. ANT inkludiert dabei ein Naming- und Directory Service, Property Support sowie asynchrone und synchrone Kommunikation. Weiters werden neueste Designaspekte durch die Implementierung dynamisch generierter Services sowie der strikten Trennung von Daten und Strukturen berücksichtigt.

Im zweiten Teil werden erste Erfahrungen mit ANT in diversen Projekten vorgestellt und ein Ausblick auf zukünftige Entwicklungen gegeben.

[Zurück zum Verzeichnis der Vorträge](#)



VConfig - Eine Java basierte Oberfläche zur Systemadministration

Peter Kalthoff, Raimund Klute, Rüdiger Schuster
(Siemens Nixdorf AG, Paderborn)

[Referent: Rüdiger Schuster]

10:10 - 10:40 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: *Java in der Anwendung 2*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(855K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Unix Systeme im High-End Bereich erreichen eine Größenordnung, die weit in das Gebiet herkömmlicher Mainframes hineinreicht. Heutige Managementsysteme müssen natürlich mit einer modernen, objektorientierten, graphischen Oberfläche ausgestattet sein, die dem Systemadministrator alle notwendigen Informationen auf einen Blick zugänglich macht. Administrationstätigkeiten müssen einfach und ohne die Kenntnis komplexer interner Strukturen durchführbar sein. In dem vorgestellten Projekt wurde eine Lösung basierend auf einer in Java realisierten Benutzerschnittstelle für das oben skizzierte Szenario entworfen. Mit dem realisierten Programm können sowohl Konfigurations- als auch Überwachungstätigkeiten ausgeführt werden.

[Zurück zum Verzeichnis der Vorträge](#)



CORBA-basiertes Workflow-Management zur Bearbeitung von Kundenanfragen aus dem Internet

David Kaluza, Felix Meyer, Matthias Reisecker
(Daimler-Benz Forschung und Technologie)

[Referent: Felix Meyer, Matthias Reisecker]

10:40 - 11:10 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: *Java in der Anwendung 2*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(671K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Electronic Commerce verändert die Kommunikation zwischen Unternehmen und Kunden. Der Web Server eines Unternehmens löst den klassischen Ladentisch ab. Bei Kundenanfragen, die über "Customer Self Service" hinausgehen, ist eine Einbindung des Kunden in interne Arbeitsabläufe des Unternehmens erforderlich [Bishop et al. 97]. Dieser Beitrag zeigt mit dem "Customer Interaction Management System" einen Ansatz zur strukturierten Bearbeitung von Kundenanfragen mit dem Ziel einer schnellen und kompetenten Beantwortung. Erreicht wird dies durch eine leistungsfähige Unterstützung der kooperativen Arbeitsprozesse, die mit der Findung der Experten und der Beantwortung der Anfragen verbunden sind. Das vorgestellte System basiert dabei auf einer CORBA Architektur, der sich sowohl der externe Kunde als auch die internen Experten bei der Beantwortung der Anfrage bedienen. Die offene Gestaltung des Systems erlaubt zum einen eine durchgängige Benutzeroberfläche für Kunden und Experten und zum anderen eine unkomplizierte Erweiterung des Systems und Nutzung der gewonnenen Daten durch dritte Systeme.

[Zurück zum Verzeichnis der Vorträge](#)



Erfahrungen mit dem Einsatz von Java und CORBA in der Entwicklung einer leistungsstarken Customer Care-Anwendung für die Telekommunikationsindustrie

Jochen Kappel, Michael Meadows
(LHS)

[Referent: Jochen Kappel]

11:30 - 12:00 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: Java in der Anwendung 3)

- [Ansicht der Vortragsfolien \[Format pdf\] \(2,6M\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Dieser Beitrag beschreibt die Erfahrungen mit dem Einsatz von Java und Corba in einer verteilten Komponentenarchitektur zur Entwicklung einer Kundenverwaltungs- und Abrechnungs-Anwendung für den Telekommunikationsbereich. Es werden verschiedenste Aspekte des Projektes angesprochen, der Fokus liegt jedoch auf der Performanz, da die entwickelte Anwendung für Installationen mit einigen Millionen Kunden und einer hohen Anzahl an Transaktionen ausgelegt wurde. Nach einer kurzen Beschreibung der Anwendungsarchitektur werden die Ergebnisse einer Reihe von Performanz-Benchmarks und eine Auslegung dieser Ergebnisse vorgestellt.

[Zurück zum Verzeichnis der Vorträge](#)



Interaktive Animationen und Visualisierungen - eine neue Qualität und "Spielwiese" für die Mechanik

Thorsten Hampel, Reinhard Keil-Slawik
(Universität-GH Paderborn)

[Referent: Thorsten Hampel]

12:00 - 12:30 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: Java in der Anwendung 3)

- [Ansicht der Vortragsfolien \[Format pdf\] \(7,2M\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Komplexe Versuchsbauten in Bereichen der Mechanik sind oftmals schwer zu verstehen und stellen somit eine Hürde für Mechanikstudenten im Hauptstudium dar. Dafür ist in vielen Fällen nicht die Thematik an sich, sondern fehlende Visualisierungen des Versuchsablauf der Kern des Problems.

Im Rahmen der PG ChiLis wird versucht oben angesprochene Lücken zu schließen, indem reale Versuche in autonome Applets entwickelt werden, so daß Mechanikstudenten sich auf spielerische Weise der Thematik erschließen können. Konsequenterweise wird Wert auf eine durchgängig hohe Motivation und Realitätsnähe gelegt. Mit Hilfe der neuen 3D-API von Java gelingt es, die Versuchsbauten dreidimensional nachzubilden, so daß Ansichten aus beliebiger Perspektive möglich sind. Eingebaute Animationen visualisieren die Prozeßdurchführung. Auch lassen sich reale Versuchsergebnisse durch Einstellung assoziierter Parameter, was optional durch Dialogelemente oder interaktiv im 3D-Fenster erfolgen kann, simulieren.

[Zurück zum Verzeichnis der Vorträge](#)



OpenCard Application Framework

Thomas Schaeck, Thomas Stober
(IBM Deutschland)

[Referent: Thomas Stober]

12:30 - 13:00 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: Java in der Anwendung 3)

- [Ansicht der Vortragsfolien \[Format pdf\] \(818K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

In diesem Artikel definieren wir eine Architektur für Smartcard-Anwendungen und stellen das OpenCard Application Framework vor, das als Basis für die Entwicklung von Anwendungen in Java gemäß dieser Architektur dient. Der erste Prototyp dieses Frameworks basiert auf dem OpenCard Framework, das im Begriff ist, zum Standard-Interface für Kartenanwendungen in Java zu werden.

Die Architektur ermöglicht durch Einsatz der Bean-Technologie visuelle Programmierung und fördert eine klare Aufteilung der Anwendungen in Benutzerschnittstelle, Anwendungslogik und kartenbezogenen Teil. Wir erreichen dadurch die Austauschbarkeit von kartenbezogenen Komponenten, so daß Anwendungen mit verschiedenen Typen von Smartcards arbeiten können, von verbreiteten dateiorientierten Smartcards bis hin zu Java Cards.

[Zurück zum Verzeichnis der Vorträge](#)



Poor Man's Genericity for Java

Boris Bokowski, Markus Dahm
(Freie Universität Berlin)

[Referent: Boris Bokowski]

14:20 - 14:50 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: *Spracherweiterung 2*)

Abstrakt:

A number of proposals have been made as to how Java can be changed to support parameterized types. We present a new proposal that does not try to provide more powerful constructs or cleaner semantics, but instead tries to minimize the changes that need to be made to existing Java compilers. In particular, we found that changing only one method in Sun's Java compiler already results in a reasonable implementation of parameterized types, which we call "Poor Man's Genericity". We have implemented our solution based on simple byte-code transformations both at compile-time and at load-time. The paper explains how our solution works, and compares it to other proposals. We also describe how the draw-backs of our approach can be overcome by making additional, but minimal changes to an existing Java compiler.

[Zurück zum Verzeichnis der Vorträge](#)



Java - formal fundiert

David von Oheimb, Cornelia Pusch
(Technische Universität München)

[Referent: David von Oheimb]

14:50 - 15:20 am Freitag, 13. November 1998 (Track 4 (JIT'98) in Sitzung: *Spracherweiterung 2*)

- [Ansicht der Vortragsfolien \[Format pdf\] \(330K\)](#) [\[Anderes Fenster nach vorne\]](#)

Abstrakt:

Dieser Artikel gibt eine Übersicht über das Projekt Bali, das die formale Behandlung möglichst vieler Aspekte von Java zum Ziel hat. Die bisherigen Arbeiten umfassen eine formale Semantik großer Teile der Java-Quellsprache und des Bytecodes, jeweils zusammen mit einem Beweis der Typsicherheit. Als Spezifikations- und Verifikationswerkzeug dient Isabelle/HOL. Wir berichten über die Ziele, die grobe Vorgehensweise, sowie die bisherigen Ergebnisse und Erfahrungen in diesem Projekt.

[Zurück zum Verzeichnis der Vorträge](#)

© 1998 JIT'98 Steeringkomitee im Auftrag der Gesellschaft für Informatik e.V.



Hinweis: Viele der Dokumente sind im PDF-Format. Der Acrobat Viewer kann bei Bedarf [hier](#) kostenlos heruntergeladen werden.

[JIT Home](#)



[English](#)



[JAVADAYS '99](#)



[About & Links...](#)



[Programmübersicht 98](#)
[Liste aller Vorträge 98](#)
[XML 98](#)
[DJEK-Vorträge 98](#)
[Alte JIT'98 Homepage](#)



[Programm 99...](#)



[Willkommen & Anmeldung...](#)



[Aktuelles...](#)



[Organisation...](#)



[Archiv...](#)

