

Neuigkeiten für Softwareentwickler
kritischer Anwendungen

Editorial

Liebe Leserin, lieber Leser,

der Frühling ist schön spürbar hier, so dass es höchste Zeit wird, für eine neue Ausgabe der *aicas news*, die Sie gerade in Ihren Händen halten.

Mit der Realtime Specification for Java etabliert sich nun ein Standard für den Einsatz von Java Technologie in Echtzeitanwendungen. Wir sind stolz darauf, hier schon jetzt ein Lösung bieten zu können.

In dieser Ausgabe gehen wir speziell auf die Optimierung der Performance von Java Anwendungen ein. Neue integrierte und grafische Werkzeuge für den Einsatz der Revisionskontrolle RCE und die Integration von Jamaica in die leistungsfähige Entwicklungsumgebung Eclipse bringen eine erheblichen Erleichterung bei der Anwendung dieser Anwendungen.

Viel Spass mit dieser neuen Ausgabe der *aicas news* wünscht Ihnen

Ihr

Fridtjof Siebert.

JamaicaVM 2.0

Java Technologie für zeit- und sicherheitskritische eingebettete Anwendungen.

Die neueste Version 2.0 der echtzeitfähigen Java-Implementierung JamaicaVM wurde als beta Release 1.9 auf der Embedded World 2003 in Nürnberg erstmals präsentiert. JamaicaVM 2.0 bietet dabei die erste industrielle Implementierung der Echtzeitschnittstelle, wie sie in der Real-Time Specification for Java definiert ist. Diese Schnittstelle bietet für Echtzeitsysteme nötige Funktionen wie Echtzeit-scheduling, Prioritätsvererbung und asynchrone Ereignisse.

aicas JamaicaVM bietet unerreichte Echtzeitfähigkeit, hohe Laufzeitperformance und geringen Ressourcenbedarf. JamaicaVM bedient sich dabei neue-

ster Technologien für die echtzeitfähige und hocheffiziente Implementierung moderner Programmiersprachen. Somit wird der Einsatz flexibler und effizienter Softwareentwicklungsmethoden basierend auf der Java Programmiersprache auch in kritischen Anwendungen ermöglicht.

JamaicaVM unterstützt derzeit die Echtzeitbetriebssysteme Vx-Works, QNX, TimeSys Linux/RT, ThreadX/NetOS und euros, sowie die Hostsysteme Solaris und Linux. Eine Version für Windows befindet sich derzeit in Entwicklung. (es)

Revisionsverwaltung

aicas VisualRCE erleichtert Teamarbeit.

Bei der Revisionsverwaltung aicas VisualRCE handelt es sich um eine Neuentwicklung des populären RCS/ CVS, bei der bewährte Konzepte beibehalten

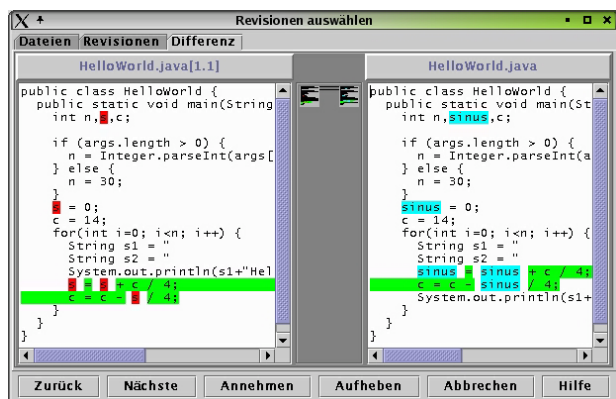
und um neue, wesentlich verbesserte Vergleichs- und Verschmelzungsalgorithmen ergänzt wurden.

Mit VisualRCE ist es möglich, komplexe Änderungen wie Verschiebungen von (Programm-) Textblöcken zuverlässig zu erkennen und richtig zuzuordnen, selbst

wenn diese verändert wurden. Dabei wurde besonderes Augenmerk auf die Verwaltung von Java- und C-Programmen gelegt. aicas VisualRCE erhöht damit die Effizienz von Programmerteams ganz erheblich.

Neben der einfach zu bedienenden grafischen Benutzeroberfläche VisualRCE sind die zugrundeliegenden Algorithmen auch als Bibliotheken (BDE) erhältlich. Dadurch können Ihre Anwendungen bequem eine bewährte und effiziente Revisionsverwaltung integrieren.

VisualRCE ist erhältlich für Linux, Windows und zahlreiche Unix Derivate. (jjh)



VisualRCE: Darstellung von Differenzen

Optimale Performance für kritische Anwendungen

Für die Verwendung von Java-Technologie in eingebetteten und echtzeit Anwendungen ist eine vorhersagbar effiziente Ausführung essentiell. Dieser Artikel beschreibt die hierfür nötigen Techniken.

Das Konzept der Java Virtuellen Maschine favorisiert die Verwendung eines Interpreters für die Ausführung von Java-Anwendungen. Dies ist jedoch zu ineffizient für viele Echtzeitanwendungen, so dass Compilationstechniken angewendet werden müssen.

Klassische Java-Implementierungen benutzen sogenannte Just-In-Time-Compilation, um oft ausgeführte Methoden zu beschleunigen. Diese Technik verursacht jedoch unvorhersagbare Ausführungszeiten für die betroffenen Methoden und ist in zeitkritischen Systemen nicht anwendbar. Darüber hinaus wird für den Compiler auf dem Zielsystem zusätzlicher

RAM- und ROM-Speicher benötigt.

Eine Alternative ist ein Ladezeit-compiler. Dabei wird der Code aller Klassen beim Laden der jeweiligen Klasse compiliert. Der gesamte ausgeführte Code ist compiliert und vorhersagbar schnell. Die Nachteile eines solchen Systems liegen im ROM- und RAM-Speicher, der für den Compiler auf dem Zielsystem benötigt wird, in der großen Zeitverzögerung beim Klassenladen und im hohen Speicherbedarf für den compilierten Code.

Der Zeit voraus: Ahead of Time

Für Anwendungen, die nur wenig Code dynamisch laden, ist sogenannte Ahead-Of-Time-Compilation ein ideales Mittel, um die Ausführungsgeschwindigkeit zu erhöhen. Performance-kritischer Code kann dabei beim Bauen des Systems compiliert werden und hochoptimierter Maschinencode ersetzt die interpretierten Bytecode-Instruktionen.

Die Verwendung von Ahead-Of-Time-Compilation vermeidet den Rechenzeit- und Speicherbedarf für den Compiler auf dem Zielsystem und die zeitliche Unvorhersagbarkeit bei der Compilation. Allerdings muss der Compiler das Mischen von compiliertem Code und interpretiertem Bytecode erlauben, um die Ausführung von Code in dynamisch nachgeladenen Klassen zu ermöglichen. Dieser nachgeladene Code profitiert dann nicht direkt von der Beschleunigung durch die Compilation, aber indirekt von den vorcompilierten Standardklassen auf dem System.

Maschinencode ist größer als Bytecode

Ein wichtiger Aspekt bei der Compilation ist die Größe des compilierten Codes. Maschinencode ist im Allgemeinen erheblich größer als Java-Bytecode-Instruktionen. Darüberhinaus verbringen die meisten Anwendungen einen Großteil der Ausführungszeit in nur einem kleinen Bruchteil Ihres Co-

des. Es macht daher Sinn, nur die Teile einer Applikation zu compilieren, in denen ein erheblicher Teil der gesamten Ausführungszeit verbraucht wird. Der Rest der Anwendung sollte im kompakteren, aber langsameren Bytecode bleiben.

Profiling findet die „Hot Spots“

Werkzeuge wie ein Profiler können dabei helfen, die wichtigsten Teile einer Anwendung automatisch zu finden und zu compilieren. So haben wir als Beispiel den "Embedded Caffeine Benchmark" mit unterschiedlichen Mengen an compilierten Methoden ausgeführt. Das Ergebnis zeigte, dass es ausreicht, nur 10% der Methoden zu compilieren, um praktisch die gleiche Geschwindigkeit wie beim Compilieren aller Methoden zu erreichen. Allerdings ist die Codegröße beim Compilieren aller Methoden etwa 100kByte größer.

Ausblick

Mehr Technologiegrundlagen zu folgenden Themen stellen wir Ihnen in den nächsten Ausgaben der *aicas news* vor:

Echtzeittechnik

Java Technologie für Echtzeitprogrammierung in Java: Die Real-Time Specification for Java.

Dynamisches Klassenladen

Die dynamischen Fähigkeiten in eingebetteten Anwendungen richtig nutzen.

Rückblick

In früheren Ausgabe *aicas news* stellten wir folgende Themen vor

Reduzierung des Speicherbedarfs (Herbst 2002)

Für die Verwendung von Java-Technologie in eingebetteten Anwendungen in Massenprodukten muss der nötige Speicher minimal sein. Eine Reihe an Techniken wurden entwickelt, um den Speicherbedarf zu verringern.

(fs)

Neuigkeiten

Neuer Finanzdirektor bei aicas

Die aicas GmbH beruft zum 9. Mai 2003 ihren früheren Finanzbeirat Peter Marggrander zum Director of Finance und Prokuristen. Damit würdigt das Unternehmen seine konstruktive und zukunftsorientierte Arbeit als Beirat, die zum positiven Jahresergebnis 2002 beigetragen hat. Vor seiner Tätigkeit bei aicas war Herr marggrander langjähriges Vorstandsmitglied der Volksbank Karlsruhe.

„Ich freue mich, dass wir Herrn Marggrander noch stärker in unser Unternehmen einbinden können“
—Götz Fluck, Geschäftsführer, aicas GmbH

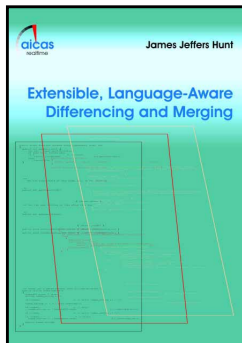


Peter Marggrander

Techniken zum Differenzieren und Verschmelzen

Dieses Buch beschreibt die Grundlage für eine leistungsfähige Versionsverwaltung sind.

Parallele Entwicklung im Team ist zur Standardtechnik in der Softwareentwicklung geworden. Obwohl praktisch jedes Revisions- und Konfigurationsverwaltungssystem einen Mechanismus für die Verschmelzung unterschiedlicher Modifikationen bietet, sind die Ergebnisse meist unbefriedigend. Dieses Buch beschreibt neue Differenz- und Verschmelzungsalgorithmen, die sprachsensitiv erheblich bessere Ergebnisse liefern. Damit schafft es die Grundlage für leistungsfähige Revisionsverwaltung und effiziente Entwicklung im Team. (aw)



James Jeffers Hunt: *Extensible, Language-Aware Differencing and Merging*, aicas GmbH, 45,- Euro, ISBN 3-8311-4444-3

Kurse und Vorträge

Java Crashkurs, Karlsruhe

12.-16. Mai 2003 und
23.-27. Juni 2003.

Grundlagen von Java. Java erfordert als objektorientierte Sprache grundlegendes Verständnis des objektorientierten Softwaremodells. Anhand ausführlicher Beispiele und Übungen wird die objektorientierte Denkweise erläutert und geübt. Kontakt: judt@fzi.de

Java und Sicherheit, Karlsruhe

28.-29. April 2003.

Aus den Themenschwerpunkten: Passive Sicherheit in Java, Sicherheitsmechanismen der Java-API, Java Authentication and Authorization Service (JAAS), Signierte Applets und Anwendungen, Java-Kryptographie-Erweiterung. Kontakt: judt@fzi.de

Java Virtual Maschine Intern

28. Mai 2003.

JVM Implementierung, Funktion des Bytecode-Interpreters, unterschiedliche Compilationstechnologien (just-in-time vs. ahead-of-time) und mögliche Optimierungen (dynamisches Binden, Interfaceaufrufe, Inlining, Call-Graph Analyse, etc.), Automatische Speicherverwaltung (Garbage Collection, Finalisation), Exceptions, Dynamisches Laden, JNI-Implementierung. Kontakt: judt@fzi.de

Realtime & Embedded Java

19.-23. Mai 2003.

Java-Entwicklung für eingebettete Systemen. Notwendige Änderungen und Anpassungen der virtuellen Maschine für die Unterstützung von Anwendungen mit Echtzeitanforderungen auf kleinen Geräten mit begrenzten Ressourcen.

Inhalte: Realtime Specification for Java(RTSJ) und Java 2 Plattform, Micro Edition (J2ME) Kontakt: judt@fzi.de

Java in Wireless und Embedded Umgebungen

21.-25. Juli 2003.

Aus dem Inhalt dieser 5-tägigen Schulung: Editionen, Konfigurationen und Profile; Anforderungen an Embedded Plattformen, Knackpunkte: Java-Umgebung und Java-Anwendung; Wireless- und Embedded-Szenarien; Echtzeit-OS, VM und Tools; Java und Echtzeit; Java und Performance; Java und Codegröße.

Dieser Kurs wird zusammen mit rpCube und Microconsult angeboten.

Kontakt: info@aicas.com

Weitere Java-Kursangebote im Internet unter <http://www.swt.fzi.de/>

Java - back to the roots

Java Technologie für Handheld-Devices.

„Java, eine Lösung für Anwendungen auf Handheld Devices und in der Customer Elektronik mit geringen Hardware-Ressourcen?“ Dies war mit das originäre Ziel bei der Entwicklung des einfachen, kleinen, plattformunabhängigen Java Systems.

Die so genannte Kilobyte Virtual Machine (KVM) der Java 2 Plattform ist ausgerichtet auf 16/32-bit RISC/CISC Microprozessoren mit einem minimalen Speicherangebot von 160 KB für das gesamte Java System. Damit lassen sich die Vorzüge einer Java-Implementierung auch für kleine eingebet-

tete Systeme nutzen.

Moderne eingebettete Systeme müssen neben der Limitierung ihrer Anforderungen an Ressourcen oft dem Anspruch genügen, echtzeitfähig und deterministisch zu agieren. Solche Systeme werden zunehmend in sicherheitskritischen Anwendungen eingesetzt, bei denen Leben und Gesundheit von Menschen von fehlerfreier Funktion der Systeme abhängt. Die Anforderung für Echtzeitsysteme steht im Kontrast zur Ressourcenbeschränkung eingebetteter Systeme. Die Realtime Java Spezifikation (RTSJ) beschreibt die Mo-

difikation der Java Virtual Maschine (JVM) und des Sprachumfangs. Damit lassen sich sicherheitskritische, echtzeitfähige Systeme mit modernen Softwaretechniken entwickeln.

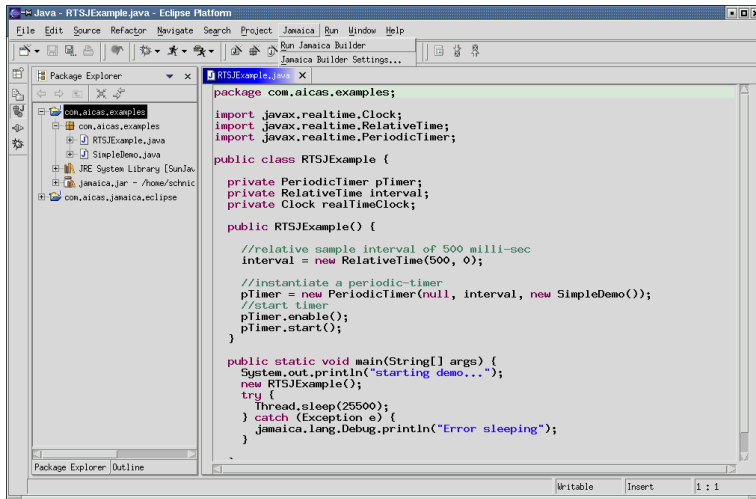
Über den Autor:

Marc Schanne ist Mitarbeiter des Authorized Academic Java Center (A/AJC) in der Forschungsgruppe Softwaretechnik am Forschungszentrum Informatik (FZI). Das A/AJC bietet Schulungen, Projektberatung und Prototyp-Implementierung speziell für die Entwicklung eingebetteter Systeme an. (ms)

aicas JamaicaVM mit Eclipse-Unterstützung

Die leistungsfähige grafische Entwicklungsumgebung Eclipse steht nun auch für Java Echtzeitanwendungen mit Jamaica zur Verfügung.

Für die Open-Source Entwicklungsumgebung Eclipse, auf der unter anderem auch IBM Websphere Studio basiert, ist jetzt ein Plugin verfügbar, das den Jamaicanen Klicks eine Konfiguration für einen Build Prozess zusammengestellt werden. Bei Unklarheit der Bedeutung einer Option ist ein Hilfetext verfügbar, der eine detail-



aicas JamaicaVM mit Eclipse IDE

ca Builder in eine IDE (Integrated Development Environment) integriert.

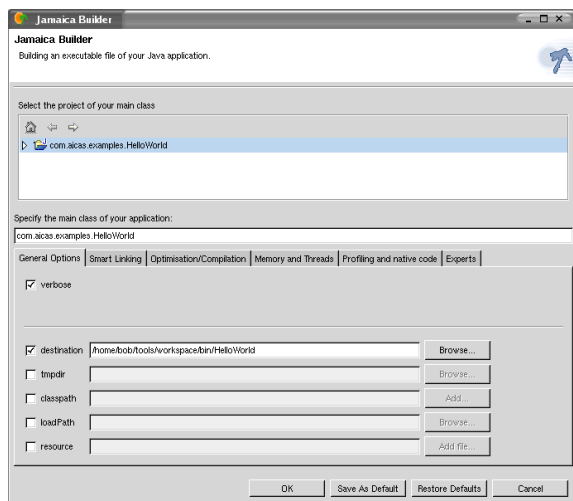
Eclipse bietet eine universelle Plattform zum Entwickeln von Software, sei dies in Java, C++ oder einer beliebig anderen Sprache. Mittels Plugins kann die Eclipse-Plattform individuell erweitert werden.

Das Jamaica Plugin ermöglicht in einem Dialog das Setzen aller Builder Optionen. Damit kann mit we-

lierte Erklärung liefert. Alle gemachten Eingaben werden vor dem Aufruf des Builders auf ihre Gültigkeit und Abhängigkeit von anderen Optionen geprüft. Fehlerhafte Angaben, wie sie in in einer Kommandozeile sehr schnell vorkommen können, werden vermieden. Die gemachten Konfigurationen werden projektbezogen gespeichert. Hat man alle gewünschten Einstellungen gesetzt, kann der Builder direkt aus Eclipse heraus gestartet werden. Auf

Wunsch wird im Anschluss die gebaute Applikation gleich ausgeführt.

Mit der Integration vom Builder in eine grafische Entwicklungsumgebung kann der Programmierer die Vorteile einer etablierten IDE mit der mächtigen Fähigkeit, Java Programme für Echtzeitsysteme zu entwickeln, kombinieren. (rs)



aicas JamaicaVM mit Eclipse IDE

Veranstaltungen

Java Forum Stuttgart

3. Juli. 35 Vorträge in 5 parallelen Tracks: Technologie, Erfahrungsberichte und Produktpräsentationen.
www.jfs2003.de

LinuxTag, Karlsruhe

10.-13. Juli. Der LinuxTag ist die größte Messe zum Thema Linux und Open Source in Europa. Dieses Jahr findet er erneut in Karlsruhe statt.
www.linuxtag.org

Net.ObjectDays 2003, Erfurt

22.-25. September. Die große Konferenz über Softwareentwicklung heute und morgen.
www.netobjectdays.org

Embedded in Munich 2003

11.-14. November. Die 1. internationale Fachmesse für Embedded Solutions. Sie findet parallel zu den sich jährlich abwechselnden Weltmessen electronica und Productronica statt. Besuchen Sie den Stand von aicas.

SPS/IPC/Drives, Nürnberg

25.-27. November. Die Messe für elektrische Automatisierungstechnik. Sie umfasst alle Komponenten bis hin zum System und bietet damit integrierte Automatisierungslösungen.

Die aicas GmbH wird auch dieses Jahr auf der SPS vertreten sein.

Kontakt

Redakteure:

Dr. James J. Hunt (jjh), Dr. Torsten Rupp (tr), Dr. Fridtjof Siebert (fs), Marc Schanne (ms), Roman Schneider (rs), Emelia Shahbazian (es), Andy Walter (aw)

aicas GmbH

Hoepfner Burg
Haid-und-Neu-Str. 18
76131 Karlsruhe
Germany

tel +49.721.663.968-0
fax +49.721.663.968-99
email info@aicas.com
web www.aicas.com