SCORE
Integration Suite
DELTA
software
technology

SCORE Integration Suite und Enterprise JavaBeans

SCORE Integration Suite basiert auf dem methodischen Ansatz der Komponententechnik und dem Konzept der Isolation Layers. Ein Einsatzschwerpunkt hierfür ist die Integration bestehender Anwendungssysteme in neue Komponentenarchitekturen, wie Enterprise Java Beans. Dazu werden vorhandene Anwendungen unter Verwendung von Generatoren in Komponenten umgewandelt, ohne dass die Anwendungen selbst modifiziert werden müssten. Diese Komponenten können auch in Umgebungen eingesetzt werden, die keine echte Komponentenarchitektur kennen wie z.B. OLTP-Systeme.

© 2003 Delta Software Technology GmbH

INHALT

SCORE Integration Suite 3
Enterprise JavaBeans und die Java Connector Architecture 3
Adaptive Frameworks 5
Frames 6
Generierte EJBs und Resource Adapter 7
Compound Operations 7
Interface-Versionierung 8
Persistenz und Session Management 8
Transaktionen 9
Weitere Features 9
Datenkonversion 9
ANGIE-Generatortechnik 9
Entwicklungsumgebung 10
Systeme und Plattformen 11
Laufzeitumgebung 11
Entwicklungsumgebung 11
Literatur 11

Delta Software Technology GmbH behält sich alle Rechte für Änderungen und Verbesserungen am Produkt vor.

Alle Rechte an dieser Dokumentation sind vorbehalten. Ohne schriftliche Genehmigung durch Delta Software Technology GmbH darf dieses Dokument weder als Ganzes noch in Teilen kopiert, photokopiert, reproduziert, übersetzt oder in irgend eine elektronische Form überführt werden.

Delta und SCORE sind eingetragene Warenzeichen der Delta Software Technology GmbH. Andere Warenzeichen sind registrierte Warenzeichen der jeweiligen Eigentümer.

SCORE INTEGRATION SUITE

SCORE Integration Suite basiert auf dem methodischen Ansatz der Komponententechnik und dem Konzept der *Isolation Layers*. Ein Einsatzschwerpunkt hierfür ist die Integration bestehender Anwendungssysteme in neue Komponentenarchitekturen, wie Enterprise Java Beans. Dazu werden vorhandene Anwendungen unter Verwendung von Generatoren in Komponenten umgewandelt, ohne dass die Anwendungen selbst modifiziert werden müssten. Diese Komponenten können auch in Umgebungen eingesetzt werden, die keine echte Komponentenarchitektur kennen wie z.B. OLTP-Systeme.

SCORE Integration Suite setzt diese Konzepte um mit generierten, anpassungsfähigen Frameworks (*Generated Adaptive Frameworks*). Dabei werden keine vorgefertigten Adapter verwendet sondern Frameworks, die maschinell, d.h. durch Generatoren, an die jeweiligen Anwendungen und Umgebungen adaptiert werden.

Die mit *SCORE Integration Suite* realisierten Integrationslösungen gewährleisten – im Gegensatz zu speziellen Adaptern oder selbst geschriebenen Anbindungen – nicht nur Portabilität, sondern auch Unabhängigkeit des Integrationsergebnisses von verwendeten Betriebssystemen, TP-Monitoren und Middleware-Produkten. Mit dem Produkt *SCORE Integration Suite* wird von Delta Software Technology eine optimale Integration in objekt-orientierte Komponentenumgebungen angeboten.

ENTERPRISE JAVABEANS UND DIE JAVA CONNECTOR ARCHITECTURE

Enterprise JavaBeans (kurz "EJB") bilden eine Komponentenarchitektur für verteilte, serverseitige und transaktionsorientierte Komponenten. Auf einem Server angesiedelt, stellen sie Clients ihre Dienste zur Verfügung. Enterprise-Beans werden ausschließlich in Java entwickelt.

Es gibt zwei unterschiedliche Ausprägungen der EJBs:

- Entity-Beans-
- und Session-Beans.

Entity-Beans repräsentieren Business-Objekte, deren Daten sich in dauerhaften Speichern (Datenbanken) befinden, Session-Beans modellieren Geschäftsvorfälle.

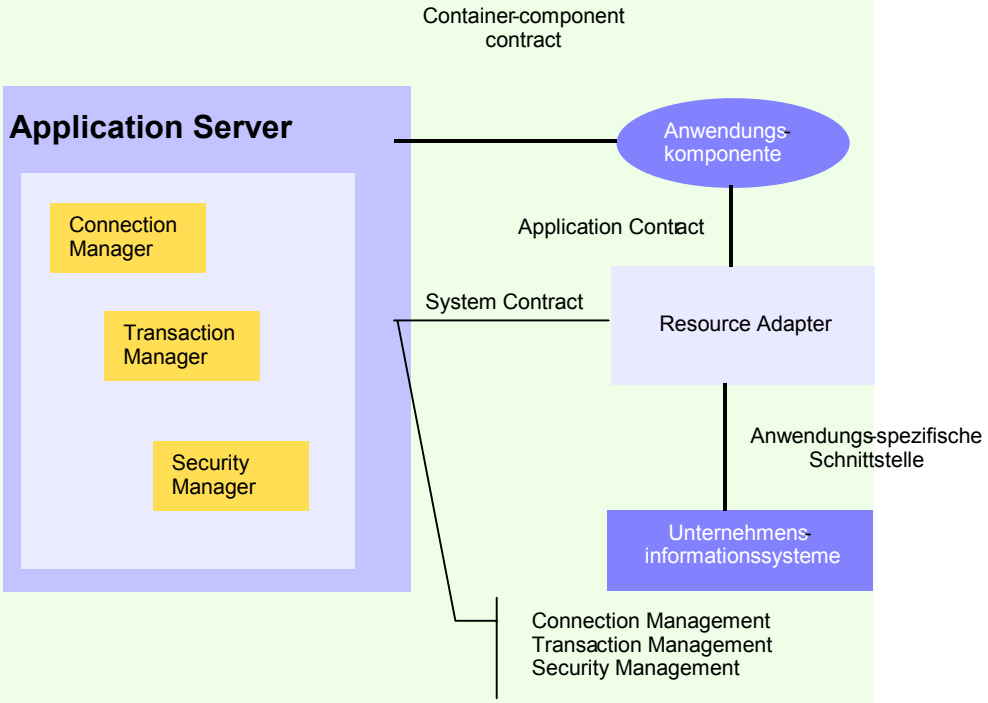
SCORE Integration Suite generiert ausschließlich Session-Beans.

Zur Ausführung benötigen EJBs einen sog. EJB-Container, der die Laufzeitumgebung zur Verfügung stellt. Dieser Container ist u.a. zuständig für die Ansprechbarkeit durch Client-Programme und das Management des Lebenszyklus. Der Container enthält zusätzliche Dienste, die über Standardschnittstellen den Beans zur Verfügung stehen. Dazu gehört insbesondere der Zugriff auf einen Transaktionsservice über JTA (**J**ava **T**ransaction **A**rchitecture).

Ein solcher EJB-Container ist normalerweise Bestandteil eines Application Servers (z.B. BEA WebLogic, IBM WebSphere, Borland AppServer), der i.d.R. noch zusätzliche Laufzeitumgebungen z.B. für Servlets oder Java Server Pages bereithält.

Enterprise JavaBeans ist Bestandteil der *Java-2-Plattform, Enterprise Edition* (kurz "J2EE"). Aktuell ist derzeit die Version 1.3. Die Spezifikation der EJB-Architektur und der Schnittstellen, die von Enterprise-Beans und den Containern zu implementieren sind, sind von der Firma Sun Microsystems in der "Enterprise JavaBean Specification" festgelegt worden. Aktuell ist Version 1.1. Version 2.0 liegt als Draft vor; hierfür gibt es bereits erste Implementierungen.

Die EJB-Spezifikation erlaubt keinen direkten Zugriff von EJBs auf nicht-Java-Anwendungen. Es gibt jedoch z.T. in den verschiedenen Application Server proprietäre Lösungen, die den Zugriff auf Legacy-Anwendungen bzw. Informationssysteme ermöglichen, die nicht in Java entwickelt wurden. Diese Lösungen sind jedoch nicht portabel. Um eine allgemein verfügbare Lösung anzubieten, wurde von Sun Microsystems zusammen mit verschiedenen anderen Herstellern (z.B. IBM, BEA, Inprise, SAP AG, ...) die *J2EE Connector Architecture Specification* (kurz "JCA") entwickelt.



Übersicht über die Java Connector Architecture

Die JCA definiert eine Standardarchitektur für die Verbindung der J2EE-Plattform mit heterogenen Unternehmensinformationssystemen. Dazu beschreibt sie Standard Schnittstellen für die Entwicklung sogenannter Resource Adapter, die die Verbindung zwischen dem Application Server (und somit den EJBs) und den vorhandenen Anwendungen herstellen. Version 1.0 der Spezifikation befindet sich derzeit im Status eines "Proposed Final Draft" (siehe [3]).

In einem Application Server können mehrere Resource Adapter parallel verwendet werden. Dabei bleibt es dem Entwickler überlassen, ob ein Resource Adapter je Informationssystem verwendet werden soll oder eher spezifische Adapter für die einzelnen Teilanwendungen des Informationssystems.

Die Schnittstelle des Resource Adapters zu den Anwendungskomponenten (**C**ommon **C**lient **I**nterface) ist ebenfalls in der Spezifikation [3] festgelegt und ist unabhängig von den Systemen, zu denen der Resource Adapter die Verbindung herstellen soll.

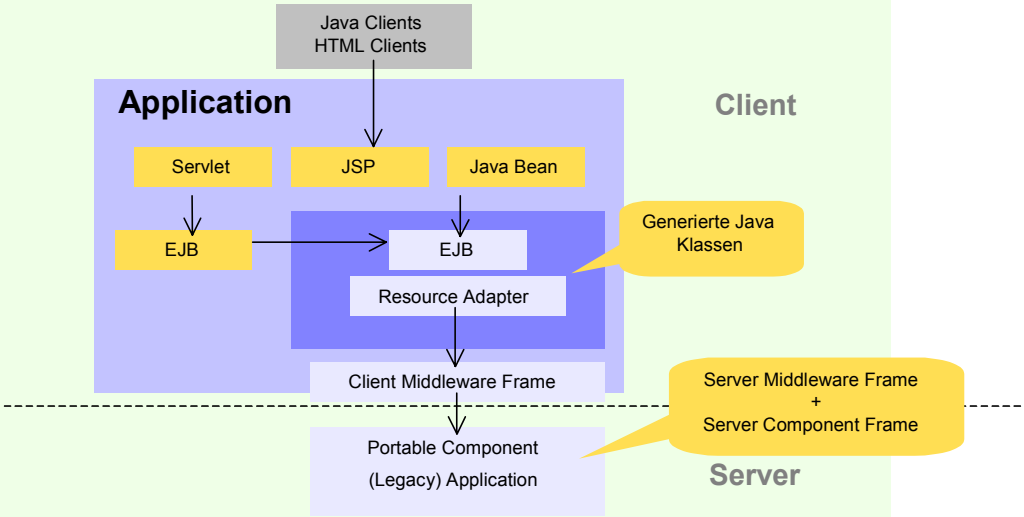
Die vom CCI definierte Schnittstelle ist relativ einfach und funktions-orientiert aufgebaut. Für einen Einsatz in objekt-orientierten Umgebungen empfiehlt sich die Verwendung passender Zugriffsklassen.

ADAPTIVE FRAMEWORKS

Im Rahmen der *SCORE Integration Suite* werden die Definitionen der Komponenten und ihrer Schnittstellen zentral in einem Component Repository abgelegt. Dabei findet die Definition auf einer logischen und deklarativen Ebene statt. Wartung und Pflege dieser Spezifikationen übernimmt ein intelligentes, interaktives Frontend: Der *Composition Manager*. Definiert werden sowohl die externen Sichten bzw. Interfaces einer Komponente – das sind die Schnittstellen mit ihren Operationen – als auch die interne Sicht, d.h. die Schnittstelle zum Funktionskern mitsamt den Abbildungsregeln (Mapping) zwischen externer und interner Sicht.

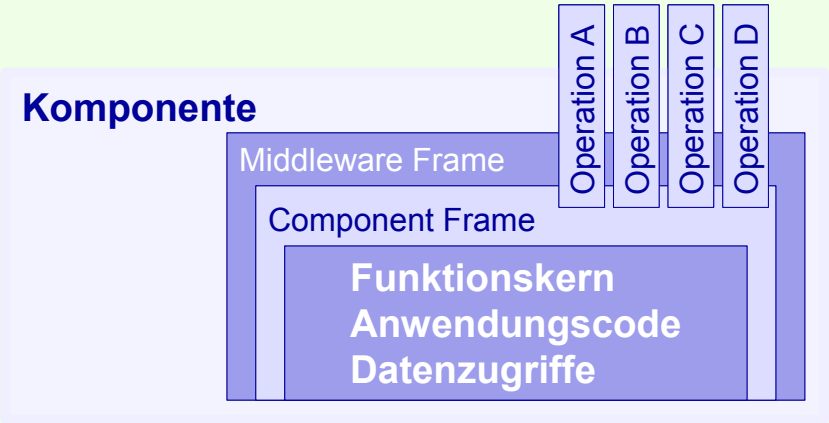
Aus der Komponentendefinition werden verschiedene Module generiert, die die Zwischenschicht zwischen Client-Komponenten, Server-Komponenten und einer Middleware bilden. Sie werden im folgenden als Frames bezeichnet. Die Frames bilden zusammen mit der Server-Anwendung eine integrierte Server-Komponente: Die Frames implementieren die Anbindung an die verwendete Middleware, sowie die Abbildungsregeln zwischen den Interfaces und der internen Funktionsschnittstelle der zu integrierenden Anwendung.

Um die so geschaffenen Komponenten in einer EJB-Umgebung verwenden zu können, werden von *SCORE Integration Suite* zusätzlich komplette Session-Beans (i.d.R. eine Bean pro Server-Komponente) generiert, die über ebenfalls generierte Resource Adapter (optional) auf die Server-Komponenten zugreifen können. Durch diese Struktur werden die erstellten Komponenten so in die EJB-Umgebung eingebunden, als wären sie selbst Enterprise-Beans. Sie können wie „normale“ Enterprise-Beans verwendet werden.

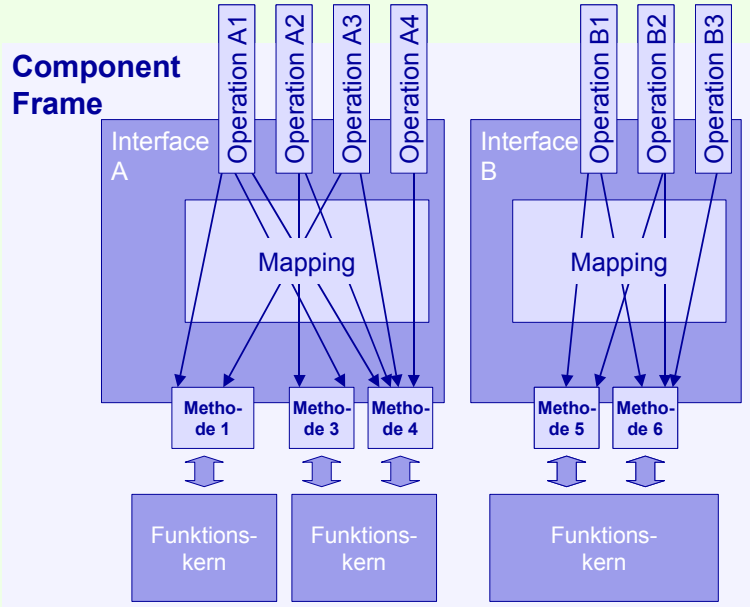


FRAMES

Die generierten Frames bestehen Server-seitig aus Component Frames und Middleware Frames. In den Component Frames sind die Interfaces mit der Abbildung (Mapping) der Operationen auf den Funktionskern implementiert. Die Middleware Frames enthalten entsprechend dem Konzept der Isolation Layers alle Aufrufe der Middleware und stellen damit die Verbindung zwischen dem Application Server und den generierten EJBs und dem Component Frame her.



Das in den Frames enthaltene Coding wird abhängig von der verwendeten Middleware und der gegebenen Systemplattform (Betriebssystem, TP-Monitor) generiert.



Der Server Component Frame ist in einer Sprache entsprechend der Server-Applikation realisiert, beispielsweise COBOL 85 oder C/C++. Er führt die Aufrufe der Server-Applikation aus und sorgt damit auf einer logischen Ebene für die Integration der Server-Anwendung.

Die zu integrierende Anwendung wird durch die generierten Component Frames zu einer portablen Komponente.

GENERIERTE EJBS UND RESOURCE ADAPTER

Für die Java-Seite generiert SCORE Integration Suite eine komplette Session-Bean - je nach Notwendigkeit entweder stateful oder stateless. Diese Enterprise-Bean tritt auf als Proxy (Stellvertreter) der Server-Komponente. Durch die Generierung *kompletter* Enterprise-Beans inklusive aller Deployment-Informationen können diese einfach in Java-Entwicklungssystemen verwendet werden.

Die Generierung der Resource Adapter ist optional. Wenn bereits ein (Standard-) Resource Adapter für eine bestimmte Middleware bzw. TP-Monitor oder Unternehmensinformationssystem vorhanden ist, so kann dieser verwendet werden. Wenn kein passender Adapter existiert oder wenn dieser nicht die benötigte Funktionalität bietet, können individuelle Resource Adapter generiert werden. Auch der Wechsel von generischen zu speziell mit *SCORE Integration Suite* generierten und wieder zurück ist jederzeit möglich, ohne dass Clients oder Server modifiziert werden müssen. Lediglich die Enterprise-Bean, die als Proxy dient, muss in diesem Fall neu generiert werden.

Vom Generator werden alle benötigten Module für die Erstellung und das Deployment der erzeugten EJBs und Resource Adapter erzeugt. Dies sind im Einzelnen die folgenden Module:

- pro Server-Komponente eine EJB mit allen notwendigen Java-Klassen als Proxy,
- Java-Klassen für den Resource Adapter (optional),
- Deployment Descriptoren für die generierte Enterprise-Bean und den Resource Adapter,
- Konfigurationsdateien für SCOUT² um die erzeugten Klassen kompilieren und zu Archiven zusammenfassen zu können.

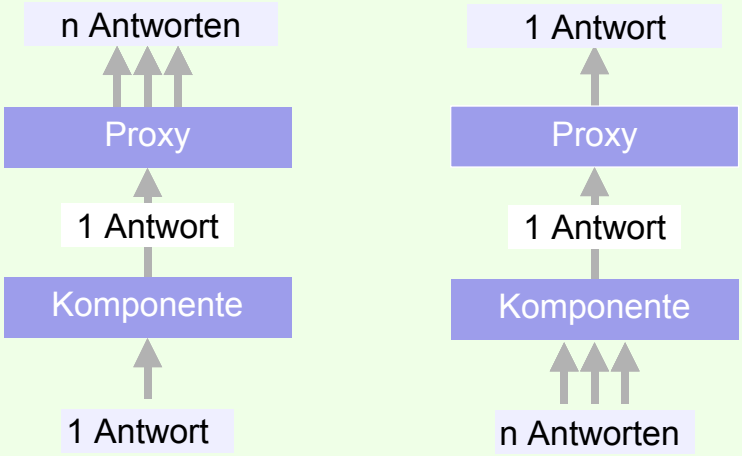
COMPOUND OPERATIONS

SCORE Integration Suite unterstützt eine Performance-Optimierung durch die Möglichkeit, mehrere Operationsaufrufe zu einem Server-Aufruf zusammenzufassen (Compound Operations, auch: Packaging).

Solche Compound Operations werden bei der Komponentendefinition mit dem *Composition Manager* spezifiziert, sind jedoch vollkommen getrennt von der Modellierung der Anwendungslogik. Die Trennung von technischer Optimierung und Sachlogik ermöglicht eine Anpassung der Optimierung unter Beibehaltung der logischen Anwendungsstruktur.

Das für die Compound Operations notwendige Coding wird generiert. Außer der deklarativen Spezifikation muss sich der Anwendungsprogrammierer um nichts kümmern.

Im Detail gibt es zwei Arten Compound Operations: Ein Request-Packaging fasst mehrere Operationsaufrufe zu einem zusammen; ein Response-Packaging mehrere Rückmeldungen zu einer Meldung.



Die Compound Operations werden vor dem Aufruf des Servers automatisch in der generierten Enterprise-Bean gebildet, um die Anzahl der Server-Aufrufe zu minimieren.

INTERFACE-VERSIONIERUNG

Eine besondere Aufgabe in der Anwendungsintegration stellt die Versionierung von Schnittstellen dar. Vor allem, wenn eine größere Anzahl von Clients nur zeitversetzt umgestellt werden kann.

Dieses Versionierungsproblem kann z.B. verursacht werden durch neue Versionen der internen Schnittstelle der Server-Anwendung. In diesem Fall bleiben die Änderungen auf die interne Schnittstelle beschränkt. Dann reicht es aus, die Komponentendefinition anzupassen und den Server Component Frame neu zu generieren.

Wird eine neue Version durch Änderungen in den Interfaces verursacht, löst *SCORE Integration Suite* dieses Versionsproblem durch ein Versionierungskonzept auf der Ebene von Operationen: Die betreffende Komponente stellt dann gleichzeitig verschiedene Versionen einzelner Operationen zur Verfügung.

Der Anwendungsprogrammierer muss sich um die existierenden Versionen nicht kümmern, da dieser Mechanismus von den Generatoren für die Client- und Server-Seite implementiert wird und an der Programmierschnittstelle unsichtbar ist.

PERSISTENZ UND SESSION MANAGEMENT

Mitunter ist es notwendig, bestimmte Komponenten oder Bestandteile von Komponenten über mehrere Aufrufe hinweg aufzubewahren. Insbesondere gibt es den Bedarf, auf bestimmte Datenbereiche auch über Transaktionsgrenzen hinweg zugreifen zu können. Hierfür stellt *SCORE Integration Suite* ein Workspace Management bereit: Einzelne Daten werden dabei mit dem Attribut "stateful" versehen, damit sie im Workspace aufbewahrt werden.

Die für die Speicherung und späteren Abruf notwendigen Aufrufe werden in die server-seitigen Frames generiert; für den Anwendungsprogrammierer stellt sich nur die Aufgabe der Spezifikation dieser Funktion.

Weiterhin unterstützt *SCORE Integration Suite* logische Kooperationsverbindungen (logical Sessions) zwischen Client- und Server-Komponenten. Damit lassen sich zusammen mit dem Workspace Management auch über Transaktionsgrenzen hinweg Bearbeitungsstände speichern. Auch hier reicht es aus, die für den Client gültige Kooperationsverbindung zu identifizieren. Der Anwendungsprogrammierer braucht sich nicht um die Details der Implementierung zu kümmern. *SCORE Integration Suite* bietet die Möglichkeit, globale und server-spezifische Workspaces zu definieren und diese optional mit Sessions zu verknüpfen.

TRANSAKTIONEN

Die EJB-Architektur realisiert Transaktionen mit Hilfe eines Transaktionsservice. Dieser Service wird vom Application Server angeboten.

Die EJB-Spezifikation sieht 2 Arten der Transaktionssteuerung vor: Implizite (Container-managed) und explizite (Bean-managed) Transaktionen. Generell wird die Verwendung impliziter Transaktionen empfohlen, da dadurch die Wiederverwendbarkeit erhöht wird.

SCORE Integration Suite unterstützt beide Arten der Transaktionssteuerung.

Werden die Resource Adapter generiert, so wird sichergestellt, dass die Server-Komponenten – falls gewünscht – von einen Transaktionsabschluss benachrichtigt werden noch bevor die Transaktion selbst abgeschlossen wird.

WEITERE FEATURES

DATENKONVERSION

Nicht alle serverseitig bekannten Datenformate werden durch Java-Datentypen abgedeckt. *SCORE Integration Suite* löst dieses Problem durch automatische Konvertierungen. Diese Konvertierungen bleiben dem Anwendungsentwickler verborgen; er sieht die Daten immer in dem Format, in dem er sie benötigt.

ANGIE-GENERATORTECHNIK

Die Generatoren der *SCORE Integration Suite* basieren auf der modernen *ANGIE*-Generatortechnik. Codefragmente werden dabei wie Objekte behandelt, die in einem Repository abgelegt werden können. Damit können sie in mehreren Generierungsstadien bearbeitet und ausgewertet werden. Ferner unterstützt die in *ANGIE* verwendete Frame-Technologie die Definition von Entwurfsmustern (Design Patterns).

Bei *ANGIE* handelt es sich um eine leistungsfähige und robuste Technologie, die besonders im Hinblick auf langfristige Wartbarkeit und Portabilität entwickelt wurde; sie stellt damit eine solide technische Basis des Produkts *SCORE Integration Suite* dar.

ENTWICKLUNGSUMGEBUNG

Über entsprechende Add-Ins werden die Tools, Teilprodukte und Generatoren der *SCORE Integration Suite* in die Development Middleware *SCOUT²* eingebunden, einer Kollektion von Entwicklungs- und Administrations-Werkzeugen, deren Funktionalität die klassischer Workbench-Tools weit übersteigt.

Bei SCOUT² handelt es sich um eine offene Integrationsumgebung, die sowohl Anbindungen beliebiger Tools erlaubt als auch selbst in bestehende Umgebungen eingebunden werden kann.

SYSTEME UND PLATTFORMEN

LAUFZEITUMGEBUNG

SCORE Integration Suite unterstützt Application Server, die auf J2EE 1.3 oder höher basieren und die Java Connector Architecture unterstützen, z.B.

- Borland AppServer ab Version 4.5
- BEA Weblogic Server ab Version 6.0
- IBM WebSphere ab Version 5.0

ENTWICKLUNGSUMGEBUNG

Für die *SCORE Integration Suite* müssen die folgenden Systemanforderungen erfüllt sein:

Hardware

- IBM-kompatibler PC
- Pentium II, min. 300 MHz
- 128 MB Hauptspeicher (minimal)

Betriebssysteme

- Microsoft Windows NT 4.0 / 2000 / XP

Software

- SCORE Integration Suite
- SCOUT²

LITERATUR

[1] Sun Microsystems: Enterprise Java Bean (EJB) specification, version 2.0
`http://java.sun.com/products/ejb`

[2] Sun Microsystems: Java 2 Platform Enterprise Edition (J2EE), versions 1.3
`http://java.sun.com/j2ee`

[3] Sun Microsystems: J2EE Connector Architecture Specification Version 1.0
`htpp://java.sun.com/j2ee/connector`

MA 14'011.03

www.d-s-t-g.com