

Umsetzung des OrViA-Frameworks mit ARIS

Sebastian Stein

IDS Scheer AG
Altenkesseler Str. 17, 66115 Saarbrücken, Germany
E-Mail: sebastian.stein@ids-scheer.com

Kurzfassung

Im Rahmen des Forschungsvorhabens OrViA wird ein Framework zur Integration verschiedener Anwendungssysteme mittels Orchestrierungsmodellen entwickelt. Die Hauptbestandteile sind dabei eine strukturierte Anforderungsanalyse, eine Transformation der Anforderungen in ein Orchestrierungsmodell und die Nutzung von Validierung an verschiedenen Punkten. Dieser Beitrag zeigt, wie das OrViA-Framework bei der Geschäftsprozessmodellierung und Implementierung unterstützen kann.

1. Einleitung

Unternehmen setzen eine Vielzahl unterschiedlicher Anwendungssysteme ein. Damit diese verschiedenen Systeme miteinander gekoppelt werden können, ist eine technische Integration notwendig. Ein aktueller Lösungsansatz ist die so genannte serviceorientierte Architektur (SOA) (vgl. z. B. [1]). Dabei steht nicht mehr das einzelne Anwendungssystem im Vordergrund, sondern die von ihm angebotenen Dienste (Services). Diese Dienste werden über Schnittstellen anderen Systemen zur Verfügung gestellt. Da die Dienste über Netzwerke abrufbar sind, bezeichnet man die Dienste als Web Services. Für die technische Beschreibung der Schnittstellen hat sich die Web Service Description Language (WSDL) [2] etabliert. Neben den Schnittstellen mit den zu-

gehörigen Operationen werden auch die Formate der Ein- und Ausgabenachrichten definiert. Weiterhin enthält eine WSDL-Definition Informationen, wie der Web Service erreicht werden kann und welche Kommunikationsprotokolle unterstützt werden.

Neben den technischen Schnittstellen muss weiterhin der zeitliche Ablauf der Integration genau beschrieben werden. Für diesen Zweck hat sich die Business Process Execution Language (BPEL) [3] etabliert. In BPEL wird die Integration immer aus Sicht eines Partners beschrieben, der die verschiedenen Web Services nutzen will. Die Beschreibung der Integration erfolgt als Prozess. Für die Definition des Prozesses stehen alle typischen Konstrukte der prozeduralen Programmierung wie Sequenz, Bedingung und Schleife zur Verfügung.

Unter Verwendung von WSDL und BPEL ist eine technische Integration verschiedener Anwendungssysteme möglich. Wagner et al. [4] weisen aber darauf hin, dass eine rein technische Integration nicht ausreichend ist. Vielmehr muss die technische Integration aus den Anforderungen der Geschäftsprozesse abgeleitet werden. Während die technischen Standards von verschiedenen namhaften Anbietern wie SAP, IBM und Oracle unterstützt werden, gibt es keinen einheitlichen Ansatz für die gesamtheitliche Bewältigung dieser Problematik. Kühne et al. [5] schlagen deshalb das OrViA-Framework vor. Dieses Framework besteht aus den drei Hauptbestandteilen strukturierte Anforderungsanalyse, Transformation und Validierung. Da ARIS eine etablierte Standardsoftware für Geschäftsprozessmodellierung ist und viele Unternehmen bereits über in ARIS modellierte

Geschäftsprozesse verfügen, soll in diesem Beitrag gezeigt werden, wie das OrViA-Framework mit ARIS umgesetzt werden könnte. Dadurch wird ein Migrationspfad für bestehende Unternehmensmodelle hin zu einer SOA aufgezeigt.

Dieser Beitrag ist entsprechend den drei Hauptbestandteilen des OrViA-Frameworks gegliedert. Weiterhin wird die Frage diskutiert, wie ein passender Service während der Modellierung gefunden werden kann. Am Ende folgen eine Zusammenfassung sowie die Diskussion von weiteren Forschungsfragen.

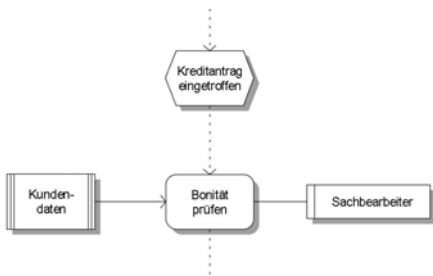


Abbildung 1: Prozessmodell mit manuellem Prozessschritt.

2. Strukturierte Anforderungsanalyse mit ARIS

Die Basis des OrViA-Frameworks ist die strukturierte Anforderungsanalyse [5]. In dieser werden die funktionalen und nicht-funktionalen Anforderungen an die Integration und die übergeordneten Geschäftsprozesse erhoben. Dabei ist es entscheidend, dass die Erhebung formalisiert erfolgt, da das Ergebnis der strukturierten Anforderungsanalyse später als Grundlage für Validierung und Transformation dient.

Die Anforderungsanalyse erfolgt auf der Ebene der Geschäftsprozesse. Eine bewährte Methode zur Modellierung von Geschäftsprozessen ist die Architektur integrierter Informationssysteme (ARIS) nach Scheer [6]. ARIS ermöglicht die Darstellung eines gesamten Unternehmensmodells und geht dabei über die reine Darstellung der dynamischen Prozesse hinaus. So können z. B. Organisationsmodelle, Landschaften von Anwendungssystemen und

die Struktur von Datenobjekten modelliert werden. Die Wiederverwendung der modellierten Objekte in verschiedenen Sichten auf das Gesamtmodell ist ein wichtiges Konzept von ARIS. Die Darstellung der dynamischen Abläufe in einem Unternehmen erfolgt in ARIS meist mit ereignisgesteuerten Prozessketten (EPK). Abbildung 1 zeigt einen entsprechenden Prozessausschnitt. In dem gezeigten Beispiel prüft ein Sachbearbeiter nach Eintreffen eines Kreditantrages die Bonität des Kunden anhand der Kundendaten. Es handelt sich bei diesem Prozessschritt um eine komplexe Aufgabe, die durchaus aus vielen Einzelschritten bestehen kann. Bonitätsprüfung wird heute bereits von verschiedenen Firmen als Dienstleistung angeboten. Es ist deshalb vorstellbar, diesen Prozessschritt durch einen externen Dienstleister als Web Service durchführen zu lassen. Daraus ergeben sich zwei Forderungen:

1. Der Web Service muss im Modellierungswerkzeug dargestellt werden.
2. Im Prozessmodell muss anstatt auf den Sachbearbeiter auf den entsprechenden Service verwiesen werden.

Punkt 1 kann erfüllt werden, wenn die im 1. Abschnitt diskutierten Servicebeschreibungen in Form von WSDL in ARIS importiert werden können. Dafür ist es notwendig, eine Abbildung der WSDL-Konstrukte in ARIS zu finden. So kann der Service in seiner Gesamtheit z. B. als UML Component gesehen werden, die über mehrere UML Interfaces (WSDL Port Types) mit dazugehörigen Operationen verfügt. Weiterhin muss der Aufbau der vom Web Service unterstützten Nachrichten in ARIS darstellbar sein. Dies kann z. B. über einen XML Schema (XSD) Import ermöglicht werden.

Um Punkt 2 erfüllen zu können, muss es im Prozessmodell möglich sein, Services darzustellen und mit Prozessschritten zu verknüpfen. Die Verwendung z. B. der beim WSDL Import erstellten UML Component im Fachmodell ist keine gangbare Lösung, da sonst technische Modellierung mit fachlicher Modellierung vermischt würde. Deshalb muss für den Service zusätzlich ein Objekt angelegt werden, das den Service aus fachlicher Sicht repräsentiert. Dieses fachliche Objekt wird dann in einem Zugriffsdiagramm mit der UML Component

verbunden. Dieses Vorgehen hat den Vorteil, dass eine klare Trennung zwischen fachlicher und technischer Modellierung möglich wird. Auch können Services im Geschäftsprozess leichter ausgetauscht werden. Weiterhin werden Fachanwender nicht mit technischen Details konfrontiert, was zu einer vereinfachten Bedienung führt. Abbildung 2 zeigt das geänderte Prozessmodell. Es ist zu erkennen, dass die Bonitätsprüfung eine automatisierte Funktion¹ ist. Die Automatisierung wird über den Service einer Rating Agentur ermöglicht.

Neben diesen hier im Detail diskutierten Modellierungskonventionen gibt es eine Reihe weiterer. Da z. B. BPEL kein Konstrukt für manuelle Tätigkeiten enthält, können nur solche EPKs nach BPEL transformiert werden, die nur noch automatisierbare Prozessschritte enthalten.

Praxiserfahrungen der IDS Scheer AG zeigen, dass die wenigsten fachlichen Prozessmodelle (EPK) den formalen Konventionen genügen. Häufig werden z. B. die Startereignisse nicht modelliert, auch wenn dies zwingend notwendig ist. Bevor eine Transformation der fachlichen Prozessmodelle in BPEL erfolgen kann, muss deshalb die Konsistenz der Fachmodelle mittels Validierung gesichert werden.

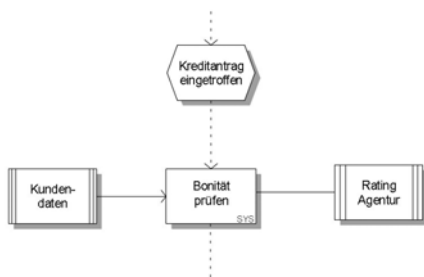


Abbildung 2: Prozessmodell mit automatisiertem Prozessschritt.

3. Validierung mit ARIS

Die Validierung ist ein weiterer wichtiger Bestandteil des OrViA-Frameworks. Nach Küh-

¹ eine Systemfunktion, gekennzeichnet durch das zusätzliche „SYS“ im Symbol

ne et al. [5] wird Validierung an verschiedenen Stellen im Framework genutzt, z. B. vor der Transformation eines fachlichen Prozessmodells in seine technische Repräsentation.

Solch ein Vorgehen ist in ARIS ebenfalls denkbar. So könnte z. B. während der Modellierung ständig die Einhaltung der Modellierungskonventionen überprüft werden. Dies könnte aber von den Fachmodellierern als eine zu große Belastung empfunden werden und es besteht die Vermutung, dass viele Anwender deshalb die Validierung deaktivieren würden. Deshalb erscheint es sinnvoller, eine Validierung erst auf Wunsch des Anwenders sowie zwingend vor Start der Transformation durchzuführen.

Die Validierung könnte dabei aus einer Vielzahl verschiedener Regeln bestehen. Viele dieser Regeln gelten dabei unabhängig davon, ob die Modellierung im Rahmen einer SOA-Strategie durchgeführt wird. So dürfen in EPKs Funktionen jeweils nur eine ein- und ausgehende Kante haben. Eine SOA-spezifische Regel hingegen ist, dass im Prozess lediglich AND und XOR Operatoren verwendet werden dürfen, aber kein OR Operator. Hintergrund ist, dass es für OR keine semantische Entsprechung in BPEL gibt.

Mit der Validierung wird letztendlich sichergestellt, dass die während der strukturierten Anforderungsanalyse erhobenen Modelle formal korrekt sind.

Das OrViA-Framework schlägt weiterhin eine Validierung des Transformationsergebnisses vor. Es muss davon ausgegangen werden, dass diese Validierung in ARIS nicht zur Verfügung stehen wird, da der Transformationsalgorithmus vor der Implementierung genau spezifiziert und nach der Implementierung getestet wird. Damit ist eine Validierung an dieser Stelle nicht notwendig.

Eine Validierung des durch den Nutzer geänderten Orchestrierungsmodells ist hingegen sinnvoll. Da BPEL eine umfangreiche Sprache ist, kann davon ausgegangen werden, dass der Anwender während der Modellierung Fehler macht. Auch hier kann wiederum die Validierung während der Modellierung oder vor dem Export in die Ausführungsumgebung stattfinden. Da allerdings die meisten Ausführungsumgebungen sehr ausgereifte Validierungsmechanismen bieten, ist es durchaus denkbar,

im Rahmen von ARIS auf solch eine Validierung zu verzichten.

```

...
<partnerLinks>
  <partnerLink
    name="RatingService"
    partnerLinkType="bon:RatingServicePLT"
    myRole="ratingRequester"
    partnerRole="ratingServiceProvider"
  />
...
</partnerLinks>
...
<process>
...
  <invoke>
    partnerLink="RatingService"
    portType="bon:GetCreditRatingPT"
    operation="CheckIndividual"
    inputVariable="IndividualRecord"
    outputVariable="CreditRating"
  />
...
</process>

```

Abbildung 3: Aufruf Rating Service in BPEL Syntax.

4. Transformation mit ARIS

Der dritte und letzte Hauptbestandteil des OrViA-Frameworks ist die Transformation fachlicher Prozessmodelle in ihre technische Repräsentation. Die technische Repräsentation wird dabei auch als Orchestrierungsmodell bezeichnet.

Ein häufig nachgefragtes Orchestrierungsmodell ist BPEL. Könnte eine zumindest teilautomatisierte Transformation von EPK nach BPEL geschaffen werden, würde dies entscheidend zur Schließung der Lücke zwischen Geschäftsprozessmodellierung und technischer Umsetzung beitragen. Ein erster Ansatz für solch eine Transformation bestände in der Überführung der fachlichen Prozessstruktur nach BPEL. Enthält eine EPK z. B. einen Zyklus, muss dieser in BPEL durch ein entsprechendes Schleifenkonstrukt ersetzt werden. Enthält die EPK z. B. eine Aufspaltung des Prozessfluss in parallel ablaufende Stränge², so muss in BPEL das Flow Konstrukt verwendet werden. Durch Web Services zu automatisierende Prozess-

² Dies kann mit einem AND Operator modelliert werden.

schritte müssen in BPEL durch das Invoke Konstrukt ersetzt werden.

Es muss das Ziel sein, neben der Prozessstruktur weitere Informationen aus dem Fachmodell in das Orchestrierungsmodell zu überführen. Da in der EPK bereits der Service modelliert ist, kann diese Information bei der Erzeugung des Invoke Konstrukts verwendet werden. Sind in der EPK bereits Ein- und Ausgabedaten modelliert, können diese in BPEL als Ein- und Ausgabevariablen übernommen werden. Abbildung 3 zeigt die BPEL-Repräsentation des in Abbildung 2 gezeigten Prozessausschnitts.

Eine Transformation von EPK nach BPEL hat aber auch Grenzen, weil a) beide Sprachen unterschiedliche Ausrichtungen und damit Konzepte haben und b) weil generell in EPK noch zusätzliche Dinge dargestellt werden können, wie z. B. manuelle Aktivitäten. Für eine vertiefende Diskussion sei auf Stein et al. [7] verwiesen.

Es ist anzunehmen, dass nach der Transformation noch weitere technische Details zum Orchestrierungsmodell hinzugefügt werden müssen. So bietet ein Service z. B. meist eine Vielzahl von Operationen. Da im EPK lediglich der Gesamtservice einem Prozessschritt zugeordnet wird, muss in BPEL noch eine konkrete Operation des Service ausgewählt werden. Anschließend muss die BPEL und WSDL Datei des Orchestrierungsmodells exportiert werden, um sie in einer entsprechenden Ablaufumgebung zur Ausführung zu bringen.

Neben der Orchestrierung bereits existierender Services muss auch eine Möglichkeit gefunden werden, wie zumindest die Anforderungen an einen neuen Service aus dem Fachmodell abgeleitet werden können. Anhand dieser Spezifikation kann mit Methoden des Software Engineering der fehlende Service entwickelt werden. Da ARIS im Bereich Software Engineering die UML Methode unterstützt, kann zumindest das Grundgerüst des Web Service bereits in ARIS erzeugt werden.

Die Ausführung des technischen Prozesses erfolgt nicht mehr in ARIS. Viele Hersteller von Ablaufumgebungen bieten neben der Unterstützung des BPEL und WSDL Standards proprietäre Erweiterungen. Da ARIS ein plattformunabhängiges Werkzeug ist, können solche Erwei-

terungen nicht direkt unterstützt werden. Es ist allerdings denkbar, dass sowohl die Transformation als auch der BPEL/WSDL Export durch individuelle Funktionen erweitert werden können, damit Kunden für eine Unterstützung dieser spezifischen Erweiterungen sorgen können.

5. Serviceidentifikation

In der Praxis verfügt ein Unternehmen nicht nur über eine handvoll Services, sondern unter Umständen über mehrere Hundert. Werden all diese Services in ARIS importiert, ist es ohne weitere Hilfsmittel fast unmöglich, während der Modellierung einen passenden Service zu wählen. Um dieses Problem zu lösen, gibt es prinzipiell 2 mögliche Ansätze, die miteinander kombinierbar sein sollten.

- Der Kunde führt eine Datenmodellierung durch. Dabei erstellt er eine Menge von fachlichen Datenobjekten, die typische Geschäftsdaten wie z. B. Adresse, Kunde, Bestellanforderung, Auftragsposition und Stückliste repräsentieren. In einem so genannten Datenglossar werden diese fachlichen Datenobjekte mit den vom Service verwendeten Datenbeschreibungen verknüpft. Der Fachmodellierer gibt in der EPK zu jedem Prozessschritt die fachlichen Ein- und Ausgabedaten an. Über das Datenglossar und die Angabe von Ein- und Ausgabedaten kann nun automatisch eine Identifizierung von passenden Services durchgeführt werden.
- Der Kunde führt eine funktionale Klassifizierung der Services durch. Dazu verbindet er den Service mit so genannten Informationssystemfunktionen. Dies wird in Abbildung 4 exemplarisch für den Rating Service gezeigt. Bei der Klassifizierung ist entscheidend, dass Informationssystemfunktionen bei verschiedenen Services wieder verwendet werden, damit Funktionscluster entstehen. So kann z. B. ein Service zur Bonitätsprüfung sowohl im Bereich Rechnungswesen, Kundenmanagement als auch Risikomanagement eingeordnet werden. In der Praxis sollte man eine Hierarchie von Funktionsclustern aufbauen, um eine bessere Strukturierung der

ServiceLandschaft zu erreichen. Der Fachmodellierer gibt im EPK zu jedem Prozessschritt die benötigten Funktionscluster an. Über die Clusterbildung kann nun ein passender Service identifiziert werden.

Es ist denkbar, dass zur Unterstützung des Fachmodellierers Programmdialoge entwickelt werden, die bei der Serviceidentifikation unterstützen. Dazu müsste ein intelligenter Algorithmus sowohl den aktuellen Kontext (die EPK) als auch die allgemeinen Informationen zu den Services (Datenglossar und Funktionscluster) auswerten können.

Ein weiteres Hilfsmittel kann z. B. darin bestehen, dass man sich auf einfache Art und Weise einen Überblick über die Verwendung eines Service verschaffen kann. Dazu können Analysewerkzeuge entwickelt werden, die z. B. zeigen, in welchen anderen Fachprozessen der Service ebenfalls verwendet wird.

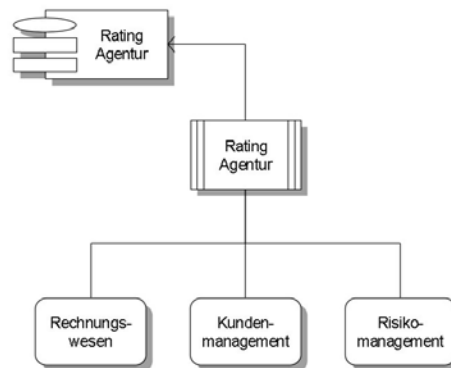


Abbildung 4: Funktionale Klassifizierung eines Service.

6. Zusammenfassung und Ausblick

Dieser Beitrag hat gezeigt, wie ein Verantwortlicher für Geschäftsprozessmodellierung das OrViA-Framework mit ARIS nutzen kann. Dabei liegt der Schwerpunkt nicht auf der technischen SOA Umsetzung, sondern in der Ableitung dieser aus den Geschäftsprozessen. Dies ist eine logische Konsequenz aus der Forderung, dass die Geschäftsprozesse die technologische

gische Umsetzung bestimmen sollen und nicht umgekehrt.

Durch den Fokus auf eine serviceorientierte Architektur wird die Aufmerksamkeit weg von den einzelnen Anwendungssystemen hin zu den gebotenen Diensten gelenkt. Diese Zerlegung der Anwendungssysteme in wieder verwendbare Services kann dabei völlig neue Möglichkeiten zur Kombination der verschiedenen Funktionalitäten bieten. Auch können z. B. mittelfristig redundante Anwendungssysteme durch Dienste anderer aktiv genutzter Anwendungssysteme ersetzt werden, was zu einer Kostenreduktion und damit zum Ausbau der Wettbewerbsfähigkeit führt.

Auch wenn der hier vorgestellte Ansatz alle Teile des OrViA-Frameworks zumindest exemplarisch aufgreift, ergeben sich viele weitere Forschungsfragen. Da in der Praxis viele formal falsche Prozessmodelle vorhanden sind, muss z. B. ein Weg gefunden werden, diese ohne großen manuellen Aufwand in formal richtige Prozessmodelle umzuwandeln. Weiterhin wurde in diesem Beitrag lediglich eine Transformation von EPK nach BPEL diskutiert. Es muss analysiert werden, wie Transformationen zwischen anderen Modelltypen wie z. B. BPMN oder XPD L aussehen könnten und ob sich die Ansätze vereinheitlichen lassen. Ein weiteres wichtiges Thema ist die Integration von Geschäftsregeln (Business Rules) in das Gesamtverfahren. Aus Anwendersicht sollte es möglich sein, die Geschäftsregeln bereits im fachlichen Prozessmodell anzugeben. Es muss deshalb untersucht werden, wie dies geschehen kann, welchen Einfluss dies auf die Transformation hat und welche Unterstützung durch Validierung geboten werden kann. Die Diskussion zu Serviceidentifikation im 5. Abschnitt hat gezeigt, dass einfach zu bedienende Mechanismen notwendig sind, den richtigen Service zu finden. Es muss deshalb untersucht werden, ob z. B. über ähnliche Ansätze wie in einer Enterprise Architecture weitere Erleichterungen möglich sind. Auch muss geprüft werden, ob eventuell Algorithmen der künstlichen Intelligenz und Techniken des Semantic Web [8] in der Geschäftsprozessmodellierung sinnvoll anwendbar sind.

Literatur

- [1] *Juric, Matjaz B.; Mathew, Benny; Sarang, Poornachandra*: Business Process Execution Language for Web Services. GB: PACKT Publishing, Birmingham 2006.
- [2] *Christensen, E.; Curbera, F.; Meredith, G.; Weerawarana, S.*: Web Service Description Language (WSDL) 1.1 / W3 Consortium. 2001.
<http://www.w3.org/TR/wsdl>, Abruf am 2006-08-14.
- [3] *Wohead, P.; Aalst, Wil M. P. van der.; Dumas, M.; Hofstede, A. H. M.*: Pattern-Based Analysis of BPEL4WS. Queensland University of Technology. Brisbane, Australien 2002.
- [4] *Wagner, Julia; La Greca, Calogero; Leyking, Katrina*: Serviceorientierte Architektur als Basis für flexibles Geschäftsprozessmanagement. In: *Fähnrich, Klaus-Peter; Thränert, Maik; Wetzel, Peter (Hrsg.)*: Umsetzung von kooperativen Geschäftsprozessen auf eine internetbasierte IT-Struktur. Leipzig 2005, S. 77–96.
- [5] *Kühne, Stefan; Thränert, Maik; Speck, Andreas*: Towards a methodology for orchestration and validation of cooperative e-business components. In: *Rutherford, Matthew J. (Hrsg.)*: 7th GPCE Young Researcher Workshop. 2005, S. 29–34.
- [6] *Scheer, August-Wilhelm*: ARIS - Vom Geschäftsprozeß zum Anwendungssystem. Springer, Berlin 2002.
- [7] *Stein, Sebastian; Kühne, Stefan; Wagner, Julia*: Das Forschungsprojekt OrViA – Orchestrierung und Validierung integrierter Anwendungssysteme. In: *Fähnrich, Klaus-Peter; Kühne, Stefan; Speck, Andreas; Wagner, Julia (Hrsg.)*: Integration betrieblicher Informationssysteme, Problemanalysen und Lösungsansätze des Model-Driven Integration Engineering. Eigenverlag Leipziger Informatik-Verbund (LIV), Leipzig 2006, S. 3–12.
- [8] *Hepp, Martin; Leymann, Frank; Domingue, John; Wahler, Alexander; Fensel,*

Dieter: Semantic Business Process Management: A Vision Towards Using Semantic Web Services for Business Process Management. In: IEEE International Conference on e-Business Engineering (ICEBE 2005). Beijing 2005, S. 535–540.

„ARIS“, „IDS“ und das Symbol „Y“ sind eingetragene Marken der IDS Scheer AG, Saarbrücken. SAP NetWeaver ist eine Marke der SAP AG, Walldorf. Alle anderen Marken sind Eigentum ihrer jeweiligen Inhaber. Alle Rechte vorbehalten. Der Inhalt dieses Dokuments unterliegt dem Urheberrecht. Veränderungen, Kürzungen, Erweiterungen und Ergänzungen bedürfen der vorherigen schriftlichen Einwilligung durch IDS Scheer AG, Saarbrücken. Jede Vervielfältigung ist nur gestattet unter der Bedingung, dass dieser Urheberrechtsvermerk beim Vervielfältigen auf dem Dokument selbst erhalten bleibt. Jede Veröffentlichung oder jede Übersetzung bedarf der vorherigen schriftlichen Einwilligung durch IDS Scheer AG, Saarbrücken.

© Copyright IDS Scheer AG, Saarbrücken, 2006

Bestandsnummer ASOA0706-D-FS