

# Szenarien für die Abbildung globaler Datenflüsse

Markus Bon  
Universität Kaiserslautern  
Postfach 3049, 67653 Kaiserslautern  
bon@informatik.uni-kl.de

In großen Unternehmen wie DaimlerChrysler existieren eine Vielzahl verschiedener, oftmals sehr komplexer Prozesse. Bei der Entwicklung neuer Fahrzeuge sind unzählige Personen beteiligt, angefangen bei den Ingenieuren, die jeweils ihren Beitrag zum Gesamtfahrzeug liefern, über Werkzeugbauer, Personalverantwortliche oder auch Vorgesetzte, die das Zusammenspiel koordinieren.

Hinzu kommt, dass die Durchführung der Entwicklung nicht allein im Hause DaimlerChrysler durchgeführt wird. Aufgrund der immer kürzeren Entwicklungszeiten wird vielmehr immer öfter auf das Fachwissen spezialisierter Firmen zugegriffen, die ihren Beitrag zum Gesamtprodukt liefern.

In diesem Zusammenhang wird schnell klar, dass der Einsatz von Workflow-Management zur Verwaltung dieser doch sehr komplexen Prozesse großes Optimierungspotential bietet. Im Zuge des Projektes COW (Cross-Organizational-Workflows [KRS01] wurde bereits untersucht, inwieweit sich inselübergreifende Prozesse beschreiben lassen, und wie ein globaler Kontrollfluss umgesetzt werden kann. Ein weiterer hochinteressanter Aspekt ist der inselübergreifende Datenfluss [BHR01, BRH02]. Um sinnvoll zusammenarbeiten zu können, müssen natürlich auch Daten zwischen den beteiligten Gruppen fließen. Zwischen den an der Konstruktion beteiligten Ingenieuren müssen beispielsweise CAD-Geometrien ausgetauscht werden, um die Passgenauigkeit der Einzelteile zu garantieren. Für einen DMU-Check (den virtuellen Zusammenbau des Systems) müssen Gittermodelle aller Einzelteile zu einem Gesamtsystem zusammengefügt werden. Die Liste lässt sich beliebig fortführen.

Diesen inselübergreifenden Datenfluss wollen wir (soweit dies möglich ist) automatisieren. Zu diesem Zweck wird langfristig eine Middleware entstehen, mit deren Hilfe sich diese Automatisierung umsetzen lässt. Dazu müssen auch auf der Modellierungsebene entsprechende Konstrukte angeboten werden, mit deren Hilfe sich die Datenflussabhängigkeit (DfA) zwischen den Inseln hinreichend beschreiben lässt. Abbildung 1 verdeutlicht das Zusammenspiel der verschiedenen Ebenen.

In diesem Bericht werden wir genauer untersuchen, welche verschiedenen Varianten von inselübergreifendem Datenfluss auftreten können. Wir werden dazu verschiedene Kriterien identifizieren, beispielsweise die Art der Datenspeicherung oder die gewählte Übertragungsart. Anschließend werden wir verschiedene Szenarien detaillierter vorstellen.

## 1 Klassifikation der Datenflussabhängigkeiten

In diesem Abschnitt wollen wir zunächst versuchen, einen Überblick über die Szenarien zu geben, die bei einer DfA relevant werden können. Dazu werden wir zunächst die beteiligten Komponenten betrachten, um anschließend verschiedene Kriterien zu identifizieren, mit deren Hilfe sich die DfAs klassifizieren lassen.

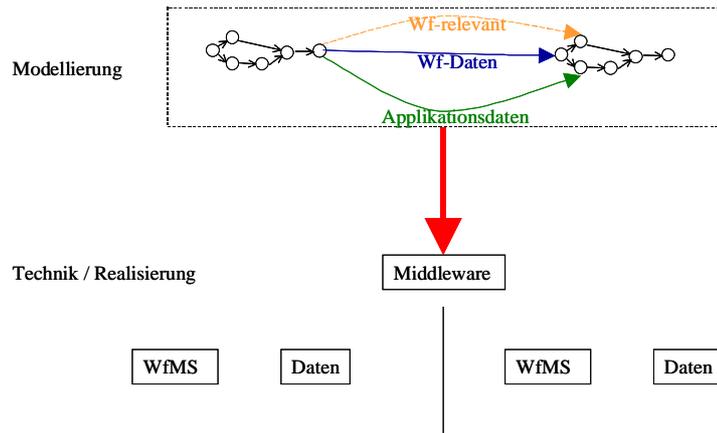


Abbildung 1 : Trennung zwischen Modellierungsebene und technischer Umsetzung

### 1.1 Beteiligte Komponenten

Zur Vereinfachung gehen wir bei den folgenden Betrachtungen davon aus, dass nur zwei Insel-Systeme miteinander über eine DfA verknüpft werden sollen. Auf beiden Inseln befindet sich jeweils ein Workflow-Management-System (WfMS) zur automatisierten Unterstützung lokaler Prozesse. In Abbildung 2 sind diese System mit WfMS<sub>1</sub> und WfMS<sub>2</sub> bezeichnet. Diese Systeme bearbeiten insbesondere auch diejenigen Workflows, die an unseren globalen DfAs beteiligt sein sollen.

Weiterhin befinden sich auf beiden Inseln Applikationen, mit deren Hilfe der Prozessrelevante Tätigkeiten durchgeführt werden sollen. Diese sind mit A<sub>1</sub> und A<sub>2</sub> gekennzeichnet. Die zu verarbeitenden Daten werden in den meisten Fällen persistent gespeichert. Daher befindet sich auf jeder der Inseln auch ein Datenhaltungssystem (DS<sub>1</sub> und DS<sub>2</sub>). Dabei kann es sich beispielsweise um Datenbankverwaltungssysteme (DBVS) oder Produktdaten-Managementsysteme (PDMS) handeln. Letztere gewinnen immer mehr an Bedeutung, da sie eine gute Unterstützung für den Produktlebenszyklus bieten [VDI99]. Die letzte Komponente ist die Workflow-Integrations-Middleware. Diese wird eingesetzt, um definierte DfAs zu überwachen, aufzulösen und für das Bereitstellen der Kooperationsdaten in der gewünschten Form zu sorgen. Anhand dieses in Abbildung 2 zu sehenden Grundaufbaus werden wir später die verschiedenen Szenarien detailliert betrachten.

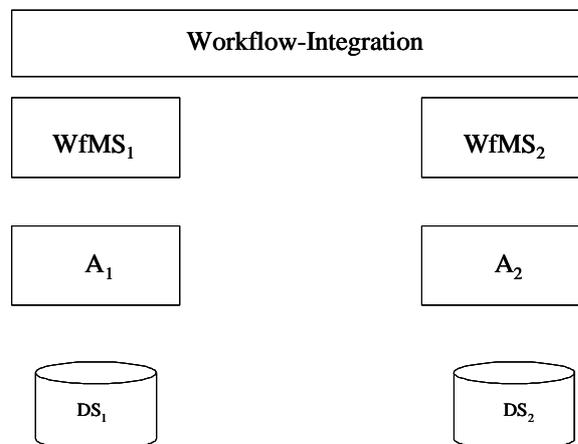


Abbildung 2 : Übersicht über die an einer globalen DFA beteiligten Komponenten

	Speicherung Quellseite	Zugriff	Übertragungsart	Speicherung Zielseite	Zugriff
1	persistent	r	materialisiert	persistent	w
2					ci
3				transient	-
4			referenziert	persistent	w
5					ci
6				transient	-
7		co	materialisiert	persistent	w
8					ci
9				transient	-
10			referenziert	persistent	w
11					ci
12				transient	-
13		direkt	-	-	-
14	transient	-	materialisiert	persistent	w
15					ci
16				transient	-

Tabelle 1 : Übersicht über die möglichen Szenarien bei inselübergreifendem Datenfluss

## 1.2 Klassifikationskriterien

DfAs sind in den verschiedensten Varianten denkbar. Um eine sinnvolle Übersicht über die auftretenden Fälle zu bekommen, ist es daher notwendig, sich geeignete Klassifikationskriterien zu definieren. Wir haben uns für folgende Kriterien entschieden:

- *Speicherung auf der Quellseite*  
Die Kooperationsdaten können entweder *persistent* im insoleigenen Datenhaltungssystem gespeichert sein, oder aber *transient* direkt nach ihrer Erzeugung durch die Quellaktivität weitergeleitet werden.
- *Zugriffsart auf der Quellseite*  
Hier unterscheiden wir drei verschiedenen Arten des Zugriffs: nur lesend (*r*), Check-Out (*co*) und *direkt*. Letzteres bedeutet, dass der Zugriff außerhalb der Systemkontrolle stattfindet.
- *Art der Übertragung der Kooperationsdaten*  
Wir unterscheiden *materialisierte* und *referenzierte* Übertragung der Kooperationsdaten. Bei der materialisierten Übertragung werden die tatsächlichen Kooperationsdaten zur Zielinsel transferiert. Referenziert bedeutet hingegen, dass nur kleine Datenmengen übertragen werden, die aber die Identifikation und den Zugriff auf die eigentlichen Kooperationsdaten ermöglichen.
- *Art der Speicherung auf der Zielseite*  
Wie auf der Quellseite gibt es auch hier die Möglichkeit der persistenten Speicherung oder der transienten Verarbeitung.
- *Zugriff auf der Zielseite*  
Bei persistenter Speicherung der Kooperationsdaten gibt es die Möglichkeiten, die Daten in das Zielsystem zu schreiben (*w*) oder über einen Check-In einzubringen. Bei transienter Verarbeitung spielt dieser Punkt keine Rolle.

Nicht alle Kombinationen sind dabei sinnvoll. Es macht beispielsweise wenig Sinn, bei transienter Weiterleitung der Kooperationsdaten auf der Quellseite die Zugriffsformen *lesend* oder *Check-Out* zu untersuchen. In Tabelle 1 ist eine Sammlung der unserer Ansicht nach relevanten Fälle zusammengestellt. Diese werden wir in Abschnitt 1.4 detailliert vorstellen.

### 1.3 Modellierungselemente

Um unsere Datenflussabhängigkeiten zu modellieren, benötigen wir Elemente, die unsere verschiedenen Kantentypen ausreichend beschreiben. In Abbildung 3 ist eine Sammlung dieser Elemente zu sehen. Die Darstellung des Knotens beschreibt, in welcher Form die Daten gespeichert sind, bzw. ob mit einem Check-Out-Mechanismus gearbeitet werden soll. Ein Pfeil mit einer einfachen Spitze beschreibt ein Übertragen der Daten, das die Originaldaten auf dem Quellsystem unverändert lässt. Eine doppelte Spitze hingegen bedeutet, dass die Daten das Quellsystem tatsächlich verlassen, bzw. als „nicht aktuell“ deklariert werden. Ein gestrichelter Pfeil symbolisiert eine DfA, deren Datenfluss nicht unter der Kontrolle des Systems liegt, aber nichtsdestotrotz im Modell auftauchen soll. Die Kooperationsdatenmenge wird durch eine entsprechende Kantenanschrift symbolisiert (hier „D“). Soll nicht materialisiert sondern referenziert übertragen werden, so wird diese Kantenanschrift in Klammern gesetzt.

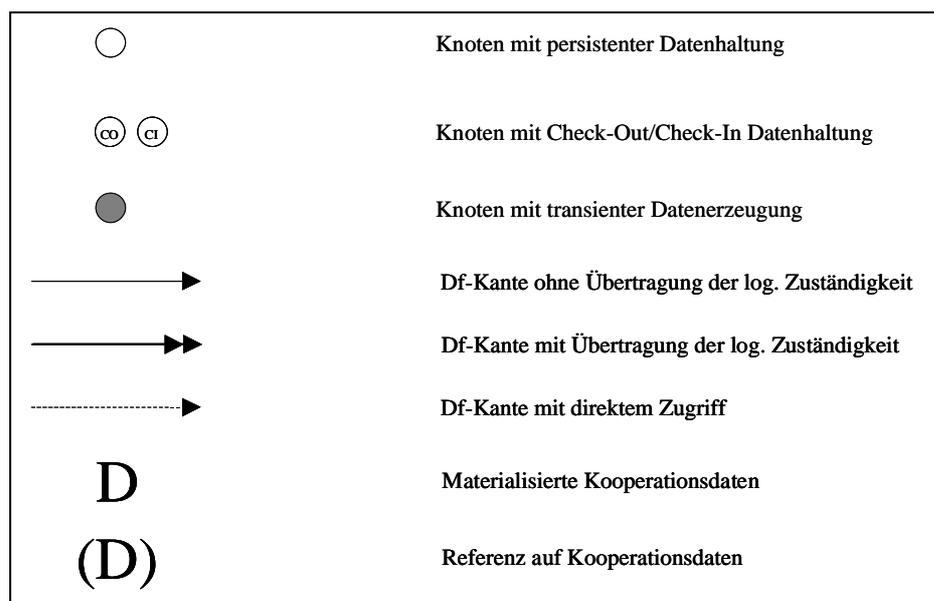


Abbildung 3 : Elemente zur Modellierung einer inselübergreifenden Datenflussabhängigkeit

### 1.4 Szenarien im Detail

Im diesem Abschnitt werden die in Tabelle 1 aufgelisteten Szenarien einzeln näher betrachtet.

### 1.4.1 Persistente Speicherung, materialisierte Übertragung

In diesem ersten Szenario werden die Kooperationsdaten D durch die Aktivität A<sub>1</sub> persistent in der Datenhaltungskomponente DS<sub>1</sub> gespeichert. Im Rahmen der üblichen lokalen Bearbeitung liefert die Aktivität dem WfMS eine Referenz auf diese Daten, über die sie wieder aufgefunden werden können. Da die Datenflusskante zwischen den Aktivitäten A<sub>1</sub> und A<sub>2</sub> eine materialisierte Übertragung vorsieht, müssen die Kooperationsdaten aus der Datenquelle DS<sub>1</sub> extrahiert und mit Hilfe der Integrations-Middleware zur Partnerinsel übertragen werden. Dort werden sie persistent in den Datenspeicher DS<sub>2</sub> eingebracht. Die Zielaktivität A<sub>2</sub> erhält vom WfMS die Referenz auf die neu eingespielten Daten und kann so auf diese lokal zugreifen. Abbildung 4 zeigt den gesamten Ablauf.

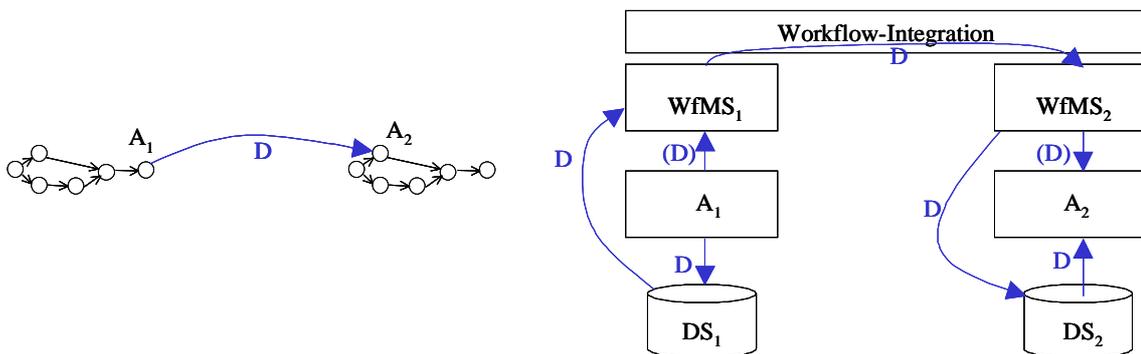


Abbildung 4 : Modellierung und Ablauf von Szenario 1

### 1.4.2 Check-In auf der Zielinsel

Dieses Szenario entspricht im wesentlichen dem vorigen, jedoch werden die Kooperationsdaten nicht einfach nur gespeichert, sondern ein Check-In Mechanismus benutzt, um die Daten einzuspielen. Diese Kante muss mit einer DfA vom Typ 7 oder 10 korrespondieren, mit deren Hilfe ein Check-Out der Daten vorgenommen wurde. Noch zu beachten gilt hier, dass der Check-In asymmetrisch durchgeführt wird, d. h., auf der Quell-Insel liegt weiterhin eine nicht von diesem Mechanismus verwaltete Version vor. Dies kann natürlich zu Problemen führen, lässt sich aber auf Grund der eingesetzten Systeme nicht grundsätzlich ausschließen. Die zugehörige Modellierung ist in Abbildung 5 zu sehen.

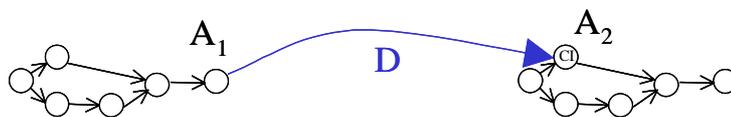


Abbildung 5 : Modellierung von Szenario 2

### 1.4.3 Direkte Verarbeitung durch Zielaktivität

Im Gegensatz zu den vorangegangenen Szenarios werden die Kooperationsdaten in diesem Fall auf der Zielinsel nicht dauerhaft gespeichert, sondern vielmehr direkt von der Zielaktivität verarbeitet. Nach der Übertragung der materialisierten Kooperationsdaten werden diese vom Ziel-WfMS direkt, wie in Abbildung 6 zu sehen ist, als Eingabedaten der Zielaktivität A<sub>2</sub> zur Verfügung gestellt.

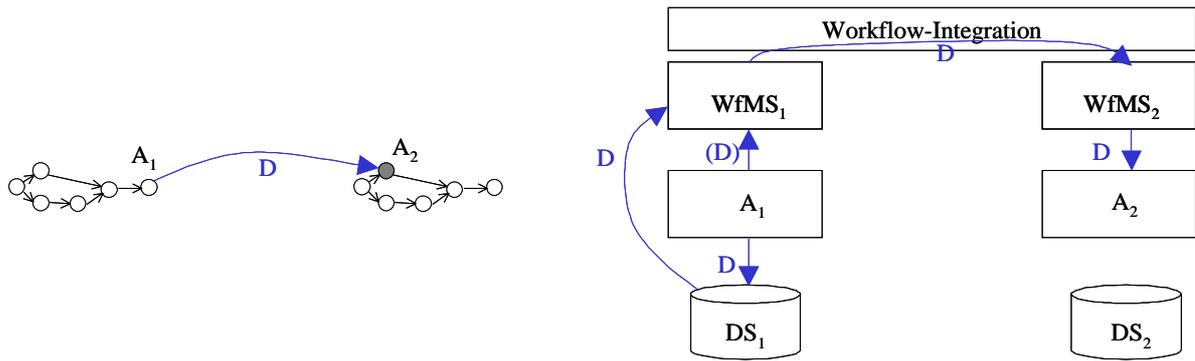


Abbildung 6 : Modellierung und Ablauf von Szenario 3

### 1.4.4 Verzögertes Bereitstellen

Dieses Szenario scheint auf den ersten Blick keinen großen Sinn zu ergeben: es wird lediglich eine Referenz auf die Kooperationsdaten übergeben, nichtsdestotrotz sollen sie auf der Quell-Seite gelesen und persistent auf der Zielinsel gespeichert werden. Hintergrund hierbei ist die nicht unerhebliche Größe von beispielsweise CAX-Dateien, die für ein großes Datenvolumen bei der Übertragung sorgen. Werden diese Daten aber nur bedingt benötigt, so kann es passieren, dass die Bereitstellung umsonst stattgefunden hat. Daher werden in diesem Szenario zuerst nur die Referenzdaten übertragen. Wird tatsächlich darauf zugegriffen, werden mit Hilfe einer „Magic“ die tatsächlichen Daten transferiert. Dieses Szenario ist somit auf Systeme beschränkt, die ein solches verzögertes Einbringen unterstützen. In Abbildung 7 sind Modellierung und interner Ablauf nochmals dargestellt.

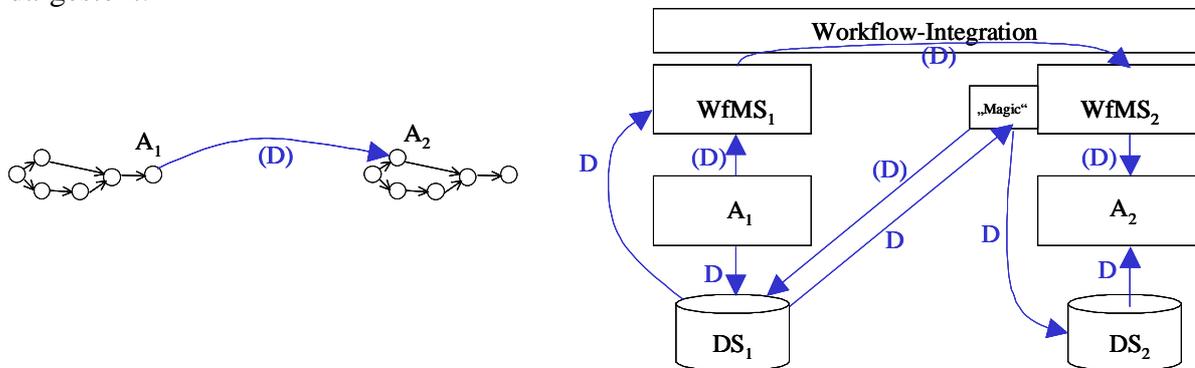


Abbildung 7 : Modellierung und Ablauf des verzögerten Bereitstellens

### 1.4.5 Verzögerter Check-In

Dieses Szenario entspricht bis auf die Einbringstrategie dem zuvor geschilderten Fall. Die zugehörige Modellierung zeigt Abbildung 8.

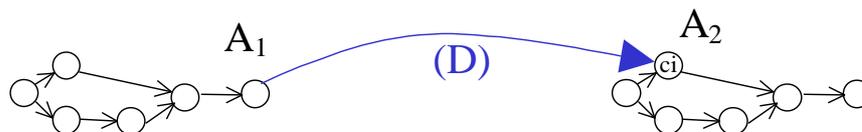


Abbildung 8 : Modellierung des verzögerten Check-Ins

### 1.4.6 Zugriff auf Originaldaten

Bisher sind wir davon ausgegangen, dass durch die Inselgrenzen kein Zugriff auf gemeinsame Daten möglich ist. In diesem Szenario betrachten wir den Fall, dass die Zielaktivität  $A_2$  direkt auf das Quellsystem  $DS_1$  zugreifen kann. Über die Datenflusskante muss somit lediglich eine Referenz zum Zielsystem übertragen werden. Der anschließende Zugriff erfolgt direkt auf die Originaldaten.

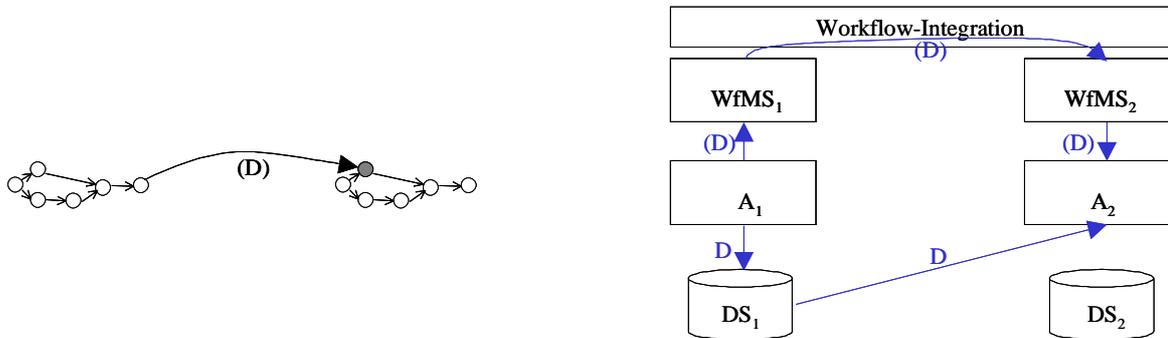


Abbildung 9 : Modellierung und Ablauf des sechsten Szenarios

### 1.4.7 Check-Out auf Quell-Insel

Dieser Fall bildet das Gegenstück zu Szenario 2. Da eine detaillierte Betrachtung hier nicht viel neues bringt, entfällt diese.

### 1.4.8 Check-Out, Check-In

In diesem Szenario werden die Kooperationsdaten von der Quellseite zur Zielseite übertragen, bzw. im Quellsystem  $DS_1$  die dort vorhandene Version als „nicht aktuell“ gekennzeichnet. Auf der Zielinsel wird entsprechend im System  $DS_2$  eine neue Version erzeugt, die lokal von der Aktivität  $A_2$  bearbeitet werden kann. Ein Zurückspielen der Daten kann durch eine weitere DfA mit vertauschten Rollen modelliert werden.



Abbildung 10 : Modellierung einer Check-Out Check-In Kante

### 1.4.9 Applikation zu Applikation

Es kann unter Umständen möglich sein, dass Ergebnisse der Aktivität  $A_1$  auf der Quellseite direkt von der Aktivität  $A_2$  weiterverarbeitet werden sollen, ohne in irgendeiner Form persistent gespeichert zu werden. Die Übertragung der Daten soll materialisiert erfolgen. In Abbildung 11 sind Modellierung und Ablauf dargestellt.

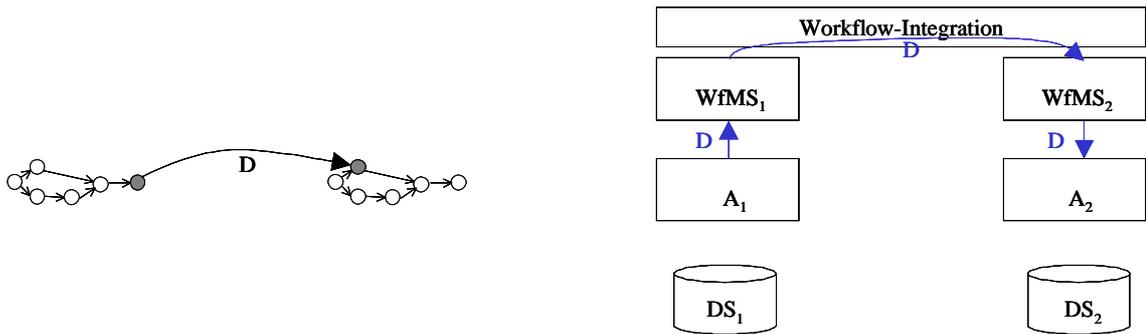


Abbildung 11 : Modellierung und Ablauf der transienten Verarbeitung

### 1.4.10 Direkter Zugriff

Als letztes soll noch Szenario 13 betrachtet werden, das aber nur ist nur bedingt zu den Datenflussabhängigkeiten zu zählen ist. Direkter Zugriff bedeutet, dass der Datenzugriff, bzw. die Datenbereitstellung, außerhalb der Kontrolle unseres Systems geschieht. Die Kante beschreibt somit nur die Existenz eines Datenflusses, gibt jedoch keinerlei Informationen über dessen Umsetzung.

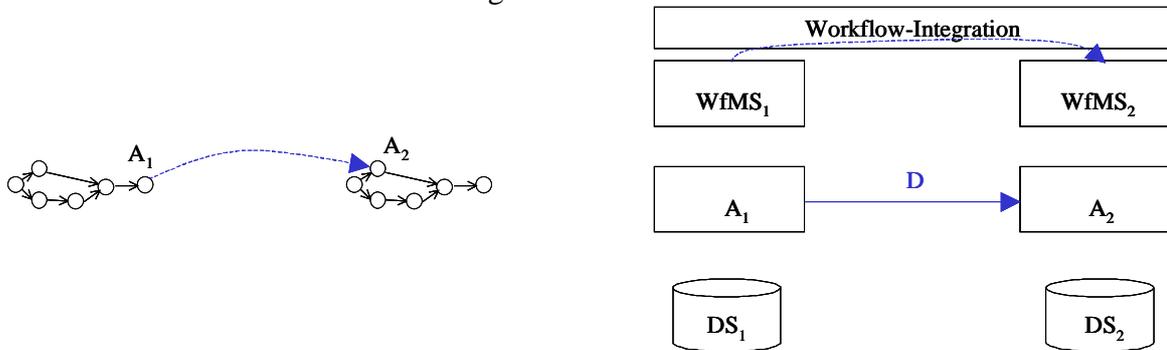


Abbildung 12 : Modellierung einer Direktzugriffskante und Anwendungsbeispiel

## 1.5 Filtern, Konvertieren und Verschlüsseln

Bisher sind wir bei allen Szenarios stillschweigend davon ausgegangen, dass die Kooperationsdaten ohne weitere Vorkehrungen von einem System zum anderen übertragen werden können. In der Praxis sieht das aber oftmals anders aus. Die Daten können in ganz verschiedenen Formaten vorliegen, so dass erst eine Konvertierung durchgeführt werden muss. Auf der Zielseite können Filter benutzt werden, um Daten bzw. Datenreferenzen anzupassen und zu ergänzen. Weiterhin ist es durchaus vorstellbar, auf der Quellseite vor Freigabe der Daten eine Art Filterung vorzunehmen, um Firmengeheimnisse zu bewahren und nur soviel an Information preiszugeben, wie tatsächlich notwendig ist. Ebenfalls zum Schutz von Firmengeheimnissen ist eine Verschlüsselung der Kooperationsdaten denkbar, um ein Abfangen durch unbefugte Dritte zu verhindern. In diesem Kapitel wollen wir betrachten, wann diese Mechanismen in unseren Szenarios zum Tragen kommen.

### 1.5.1 Filter

Filter sind kleine Anwendungen, die die „Rohdaten“ auf ihrem Weg von der Quell- zur Zielinsel passieren müssen. Dabei kann es sich um Filter handeln, die geheime Details ausblenden oder Teile geschickt in ihrer Funktionalität verändern [Naw2001]. Weiterhin können Filter benutzt werden, um auf der Quell-Seite Daten aufzubereiten, Referenzdaten

anzupassen oder auf andere Namenskonventionen zu übersetzen, zusätzliche Daten anzufügen und vieles mehr. Bei der Modellierung werden Filter durch Anhänge an der DfA-Kante repräsentiert, wie dies in Abbildung 13 zu sehen ist.

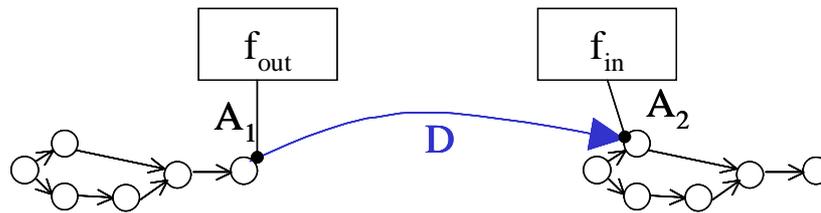


Abbildung 13 : Modellierung von Filtern

### 1.5.2 Konvertierung

Ein nicht zu vernachlässigendes Problem tritt durch die Verwendung verschiedener Datenformate auf. Zwei weitverbreitete CAD-Systeme sind beispielsweise CATIA von Dassault Systemes [Catia] oder UniGraphics der Firma PLM Solutions [UG]. Sollen nun CAD-Dateien, die mit dem einen Produkt erstellt wurden, mit dem anderen weiterverarbeitet werden, so muss erst eine Konvertierung vorgenommen werden. Hierzu sind verschiedene Ansätze denkbar:

- die Integrations-Middleware stellt entsprechende Konvertierungstools zur Verfügung. Dabei ist jedoch zu beachten, dass eine große Anzahl dieser Tools notwendig werden.
- die Kooperationsdaten werden auf den Inseln vor dem Verschicken in ein neutrales Zwischenformat übersetzt. Diese Funktionalität wird inzwischen von vielen PDM-Systemen direkt unterstützt, jedoch ist zu beachten, dass bei jedem Konvertierungsvorgang Qualitätsverluste in Kauf genommen werden müssen.
- dem Partner wird das zu benutzende Werkzeug vorgeschrieben, die Konvertierung kann entfallen. Dieser Ansatz wird von größeren Firmen durchaus verfolgt (bei der Firma Opel ist dies beispielsweise eine der Grundanforderungen an einen Zulieferer), jedoch kann nicht jeder Zulieferer es sich leisten, alle Kundensysteme bei sich einzusetzen. Gerade kleine, sehr spezialisierte Firmen mit großem Know-how werden diesen Weg kaum gehen.

Die Modellierung der Datenformate geschieht durch Angabe des vorliegenden Formats in eckigen Klammern am zugehörigen Knoten (Abbildung 14).



Abbildung 14 : Modellierung von Datenformaten

### 1.5.3 Verschlüsselung

Der Umgang mit Firmendaten ist nicht unkritisch, oft werden als Kooperationsdaten wichtige Details benötigt, die nicht in die Hände Dritter geraten dürfen. Aus diesem Grund kann eine verschlüsselte Übertragung notwendig werden. Bei der Verschlüsselung können zwei verschiedene Ebenen unterschieden werden. Zum einen kann die Verschlüsselung durch die Wf-Integrationskomponente erfolgen. Somit ist sichergestellt, dass die Daten geschützt sind, solange sie sich in deren Kontrollbereich befinden. Dieses Szenario ist in Abbildung 15 dargestellt. Da die Verschlüsselung komplett unter der Kontrolle der Integrationskomponente stattfindet, kann hier auf symmetrische Verfahren wie das bekannte DES zurückgegriffen werden. Durch Unterstützung von entsprechender Hardware kann dann eine sehr schnelle Verschlüsselung ermöglicht werden.

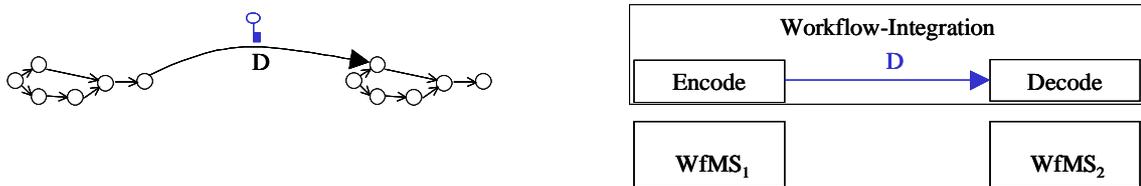


Abbildung 15 : Verschlüsselung der Kommunikation der Integrationskomponente

Eine weitere, höhere Sicherheitsstufe erhält man, wenn die Verschlüsselung bereits auf den Inseln geschieht, beispielsweise mit einem asymmetrischen Verfahren wie PGP [Zim95]. Hierdurch kann z.B. zusätzlich die Identität des Absenders geprüft werden. Die Integrationskomponente kann die Rolle des vertrauenswürdigen Verwalters der öffentlichen Schlüssel übernehmen. Modellierung und Ablauf sind in Abbildung 16 dargestellt.

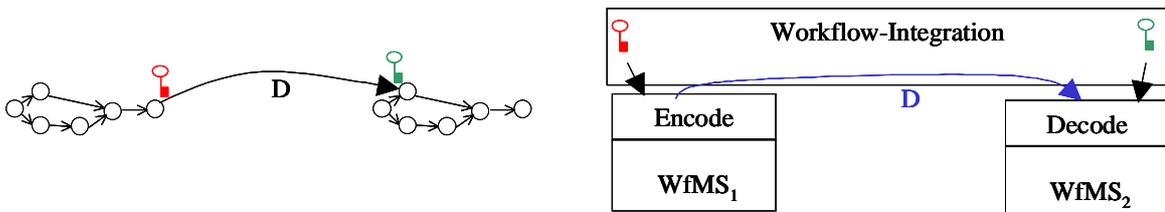


Abbildung 16 : Verschlüsselung der kompletten Übertragung

## 1.6 Formular zur Beschreibung einer DfA

Um die Modellierung zu unterstützen bietet es sich an, alle bisher identifizierten Parameter einer globalen DfA in einem geeigneten Formular zusammenzustellen. In Oist ein Entwurf für ein entsprechendes Formular zu sehen.

	<b>Spezifikation DFA</b>	
<b>Quell-Seite</b>		
	Wf-Typ	
	Wf-Aktivität	
	Datenquelle (persistent)	
	Zugriffsart (read / checkout / direct)	
	Materialisierte Daten oder Referenz	
	Format der Daten	
	Ausgangsfiler	
	Ausgangskonvertierung	
	Sendemechanismus	
	Verschlüsselung	
<b>Ziel-Seite</b>		
	Wf-Typ	
	Wf-Aktivität	
	Eingangsformat	
	Eingangsfiler	
	Eingangskonvertierung	
	Empfangsmechanismus	
	persistente Speicherung	
	Einspeicherungsart	

**Tabelle 2 : Formular zur Beschreibung einer globalen Dfa**

## 2 Literatur

- [BRZ00] Bon, M., Ritter, N., Zimmermann, J.: Interoperabilität heterogener Workflows, Proc. Grundlagen von Datenbanken, 2000: 11-15
- [BHR01] Bon, M., Härder, T., Ritter, N.: Produktdaten-Verwaltung in heterogenen Workflow-Umgebungen, Interner Bericht, Dez. 2001
- [BRH02] Bon, M., Ritter, N., Härder, T.: Sharing Product Data among Heterogeneous Workflow Environments, in Proc. Int. Conf. CAD 2002 - Corporate Engineering Research, Dresden, March 2002, pp. 139-149.
- [Catia] CATIA Users' Page, [www.catia.com](http://www.catia.com)
- [KRS01] Kulendik, Ottokar; Rothermel, Kurt; Siebert, Reiner: Cross-organizational workflow management - General Approaches and their Suitability for Engineering Processes. In: Schmid, Beat (ed.); Stanoevska-Slabeva, Katarina (ed.); Tschammer, Volker (ed.): Proceedings of the First IFIP-Conference on E-Commerce, E-Business, E-Government : I3E 2001 ; Zuerich, Switzerland, October 3-5, 2001
- [Naw01] Nawotki, A.: Eine selektive Methode zur Verschlüsselung von Konstruktionsdaten mit Wavelets, Dissertation, Kaiserslautern, 2001
- [Step92] Subcommittee 4 of ISO Technical Committee 184, Product Data Representation and Exchange - Part 11: The EXPRESS Language Reference Manual, ISO Dokument, ISO DIS 10303-11, August 1992
- [SZ98] Steiert, H.-P., Zimmermann, J.: JPMQ - An Advanced Persistent Message Queuing Service, in: Advances in Databases, Proc. 16th Nat. British Conf. on Databases (BNCOD16), LNCS 1405, Springer, 1998: 1-18
- [UG] PLM Solutions, Unigraphics product page, [www.unigraphics.de/index.shtml](http://www.unigraphics.de/index.shtml)
- [VDI99] Verein Deutscher Ingenieure, VDI-Richtlinien VDI2219, Datenverarbeitung in der Konstruktion – Einführung und Wirtschaftlichkeit von EDM/PDM Systemen, November 1999, Beuth Verlag GmbH, 10772 Berlin
- [Zim95] Zimmermann, P.: The Official PGP User's Guide, MIT Press, 1995