

Migration von EPK zu BPMN

Gero Decker, Willi Tscheschner
Signavio GmbH
(gero.decker,willi.tscheschner)@signavio.com

Jörg Puchan
Hochschule München
puchan@hm.edu

Abstract: Prozessmodelle sind wichtige Grundlagen zur effizienten und effektiven Gestaltung von Organisationen. Darüber hinaus fordern zahlreiche nationale und internationale Standards und Normen die Modellierung und Dokumentation der Geschäftsprozesse. Daher verfügen viele Unternehmen heute über eine große Menge an ausgereiften und aufwändig erstellten Prozessmodellen. Im deutschsprachigen Raum sind ereignisgesteuerte Prozessketten (EPK) weit verbreitet. International wird der Standard Business Process Modeling Notation (BPMN) zunehmend eingesetzt. Die internationale Zusammenarbeit und Verflechtung der Organisationen ist ein verstärkender Grund dafür, dass Unternehmen ihre EPKs in die BPMN-Notation überführen möchten. Hierzu wird in diesem Beitrag dargestellt, inwieweit dies automatisiert abgewickelt werden kann und welche Grenzen der Automatisierung gesetzt sind. Nach einem kurzen Überblick über die Konstrukte von erweiterten EPKs und BPMN werden sukzessive deren Transformationsmöglichkeiten dargestellt. Während bei zahlreichen Konstrukten die Transformation automatisiert erfolgen kann, ergeben sich in einigen Fällen Schwierigkeiten bzw. Mehrdeutigkeiten. In diesen Fällen ist i.d.R. nur ein semi-automatisches Vorgehen möglich. Durch die Einhaltung geeigneter Modellierungsrichtlinien für EPKs lassen sich einige dieser Problemfälle vermeiden. Eine Teilmenge der dargestellten Transformationsregeln wurde bereits in einer prototypischen Implementierung auf Grundlage des Modellierungswerkzeugs Signavio Process Editor umgesetzt.

1 Einleitung

Die Gründe für die Prozessmodellierung in Unternehmen sind vielfältig. Häufig werden Prozesse modelliert, um diese nachfolgend zu optimieren, im Unternehmen verbindlich zu machen oder prozessunterstützende Systeme zu entwickeln bzw. anzupassen. Weitere Motivationen zur Prozessmodellierung resultieren aus der Notwendigkeit, gesetzliche, aufsichtsrechtliche, revisorische oder ähnliche Anforderungen zu erfüllen (Compliance), freiwillig bzw. marktgetrieben die Prozessorientierung zu fördern (z.B. Zertifizierungen) oder Vergleiche mit Standardprozessmodellen (z.B. CMMI, ITIL) zu ermöglichen bzw. diese schrittweise einzuführen. In jedem dieser Fälle wird in Unternehmen und Behörden eine große Menge an Ressourcen, Zeit und Kosten aufgewandt, um diese Prozessmodelle zu erstellen, zu optimieren, zu implementieren und sie aktuell zu halten. Im deutschsprachigen Raum sind dabei seit Mitte der neunziger Jahre des vergangenen Jahrhunderts ereignisgesteuerte Prozessketten (EPK) und deren Erweiterungen (eEPK) auf der Grundlage bzw. nach den Vorgaben der "Architektur rechnergestützter Informationssysteme" (ARIS) zunehmend und heute weit verbreitet. Die Modellierungsmethode und -sprache wurde ursprünglich von Scheer entwickelt und dokumentiert [Sch92, Sch98] und dann im

Rahmen zahlreicher Projekte weiterentwickelt und verfeinert [SJ02, IDS06]. Heute sind EPKs bei zahlreichen Unternehmen im Einsatz. Es kann angenommen werden, dass ein Großteil der ca. 7.500 ARIS¹-Kunden der IDS Scheer AG EPKs zur Modellierung ihrer Prozesse einsetzen. Genaue Statistiken zum Einsatz und zum Bestand von EPKs liegen den Autoren allerdings nicht vor. Die internationale Entwicklung vor allem im englischsprachigen Raum ist zwar erst später in Gang gekommen, gewinnt aber durch die weltweite, unternehmensübergreifende Kooperation, Verflechtung und Standardisierung immer mehr an Bedeutung. Die Business Process Modeling Notation (BPMN [bpm09]) wurde 2004 erstmalig von der Business Process Management Initiative (BPMI) veröffentlicht. Der Standardisierungsprozess wurde von der Object Management Group (OMG) übernommen und eine überarbeitete Version wurde im Juni 2005 veröffentlicht. Die aktuelle Version 1.2 steht seit Februar 2009 zur Verfügung. Momentan wird BPMN 2.0 vorbereitet und eine Veröffentlichung ist im Laufe des Jahres 2010 zu erwarten.

Unternehmen arbeiten heute zunehmend international zusammen. Viele Unternehmen sind selbst internationale Konzerne. Auch internationale Standards und regulatorische Anforderungen verlangen zunehmend standardisierte und dokumentierte Prozesse, z.B. das US-Gesetz Sarbanes-Oxley Act 2002 (SOX, aufsichtsrechtliche Anforderungen für in den USA börsennotierte Unternehmen oder Unternehmen, deren Anteile in den USA außerbörslich gehandelt werden sowie deren Tochtergesellschaften), COBIT (Control Objects for Information and related Technology, internationaler Wirtschaftsprüferstandard [IT-05]) oder CMMI (Capability Maturity Model Integration, Best Practice-Modell des Software Engineering Instituts der Carnegie Mellon University mit 22 Referenzprozessen/sog. "Prozesskategorien" für die Optimierung der IT-Organisation [CKS08]). Daraus folgt, dass ein international anerkannter Standard wie BPMN zunehmend an Bedeutung gewinnt und gewinnen wird. Auch Unternehmen des deutschsprachigen Raums werden durch internationale Kooperationen bzw. Verflechtungen, aus Gründen der leichteren Gewinnung von qualifiziertem, international tätigen Personal oder aus anderen Gründen zunehmend BPMN als Modellierungssprache einsetzen wollen. Die Autoren begleiten derzeit einige Unternehmen, bei denen diese Entwicklung zu verzeichnen ist.

Anwenderunternehmen, die also wertvolle Prozessmodelle in Form von ereignisgesteuerten Prozessketten erstellt haben und durch externen Druck oder aus eigenem Antrieb zur Modellierungsmethode BPMN wechseln wollen, müssen also überlegen, wie sie diese "Wertgegenstände" möglichst aufwandsarm überführen können. In diesem Beitrag wird dargestellt, nach welchen Regeln ereignisgesteuerte Prozessketten möglichst automatisierbar in BPMN-Modelle überführt werden können und bei welchen Modellelementen eine automatisierte Lösung nicht unmittelbar möglich bzw. sicher ist, da die Semantik der Sprachen dies nicht zulässt.

Dieser Beitrag stellt somit eine direkte Transformation von EPK zu BPMN vor und ist wie folgt strukturiert. Nach einer kurzen Vorstellung der verwandten Arbeiten werden eEPKs und BPMN jeweils in einem eigenen Abschnitt vorgestellt. Die Transformation wird in Abschnitt 5 vorgestellt. Die Ergebnisse werden anschließend diskutiert. Der Beitrag schließt mit einer Zusammenfassung und einem Ausblick.

¹ ARIS ist eine geschützte Marke der IDS Scheer AG

2 Verwandte Arbeiten

Die Transformation von EPKs nach BPMN wird in der Literatur bisher kaum diskutiert. Ein Beitrag von Hoyer et al. [HBS07] ist ein erster Schritt in Richtung einer (semi-)automatischen Transformation. Neben einer Abbildung der Kontrollflusskonstrukte wird hier vor allem auf die Unterscheidung zwischen internen Aktivitäten und Kommunikationsaktivitäten (d.h. Senden und Empfangen von Nachrichten).

Grundlage für eine saubere Transformation ist eine präzise Definition der Semantik der Quell- und Zielsprache. Existierende Arbeiten zu EPK und BPMN konzentrieren sich hierbei vor allem auf die Kontrollflussesemantik. Während die Semantik von XOR- und UND-Konnektoren unstrittig ist, ist eine rege Diskussion bzgl. ODER-Konnektoren (OR-Joins) entstanden. Sowohl auf der EPK-Seite als auch auf der BPMN-Seite gibt es eine Reihe von Arbeiten, die jeweils über Transformationen zu einem Formalismus die Kontrollflussesemantik der einzelnen Modellierungskonstrukte abbilden [DGHW07, Men07]. Eine Übersicht über die Diskussion findet sich in [KMWL09].

Neben der Kontrollflussesemantik wurde in der Literatur auch die Prozessinstanziierungsemantik von EPK und BPMN beleuchtet. Dies ist besonders relevant, da sowohl EPK als auch BPMN mehrere Eintrittspunkte in einen Prozess zulassen und die Anzahl der Modellierungsfehler stark mit der Anzahl an Startknoten in einem Prozessmodell korreliert [DM09].

Als Alternative zu einem direkten Mapping von EPK nach BPMN kann auch ein zweistufiges Mapping vorgenommen werden, welches zunächst ein EPK-Modell in eine dritte Sprache abbildet und dieses Zwischenergebnis wiederum nach BPMN übersetzt. Ein solcher Ansatz wird propagiert in [VZHA05] unter Verwendung einer speziellen XML-Sprache. Zwar werden EPK und BPMN als motivierendes Beispiel in diesem Beitrag genannt, die genauen Mappingvorschriften werden allerdings nicht näher erläutert. Besonderes Augenmerk gebührt bei dieser Strategie der Frage, ob die dritte Sprache tatsächlich eine semantische Obermenge der Quell- oder Zielsprache bildet. Ist dies nicht der Fall, kann durch den “Umweg” mehr Information verloren gehen als bei einer direkten Abbildung. Insofern kommt der Einsatz von Petrinetzen oder BPEL nicht in Frage, obwohl entsprechende Mappings von EPKs / zu BPMN existieren [SKI08, KWL09, WDGW08, SKLN09].

Gerade in Situationen, bei denen die Zielsprache eine höhere Ausdrucksmächtigkeit besitzt als die Quellsprache, sind Spracherweiterungen / -verfeinerungen in der Quellsprache eine häufig verwendete Strategie, um eine genauere Abbildung zu erreichen. Für EPKs existieren Erweiterungen, die genau aus diesem Grund hinzugefügt wurden, z.B. für ein Mapping in Richtung Web-Service-Orchestrierungen [HW08].

3 EPKs im Überblick

Im Folgenden werden die Grundelemente der ereignisgesteuerten Prozessketten (EPKs) grob skizziert. Für Details wird auf das Methodenhandbuch ARIS [IDS06] bzw. die zahlreich vorhandene Literatur (z.B. [Sch98] oder [Gad08]) verwiesen. Ereignisgesteuerte

Prozessketten (EPKs) bzw. deren Erweiterungen sind nicht formal normiert und unterliegen pragmatischer, kontinuierlicher Verbesserung bzw. Veränderung, wie es für Industriestandards häufig der Fall ist. Darüber hinaus werden von unterschiedlichen Beratern und Anwendern gelegentlich eigene Erweiterungen oder Ergänzungen entwickelt und propagiert. Im Kern stimmen die bekannten Grundelemente der EPKs jedoch im Wesentlichen überein. Bei ereignisgesteuerten Prozessketten werden die Elemente der Funktionssicht (Funktionen) über Ereignisse verkettet. Ereignisse repräsentieren einen “betriebswirtschaftlich relevanten Zustand eines Informationsobjektes, der den weiteren Ablauf des Geschäftsprozesses steuert oder beeinflusst. Ereignisse lösen Funktionen aus und sind Ergebnisse von Funktionen. Im Gegensatz zu einer Funktion, die ein zeitverbrauchendes Geschehen darstellt, ist ein Ereignis auf einen Zeitpunkt bezogen.” (vgl. [IDS06]). Die Verkettung der Elemente erfolgt entweder sequenziell (repräsentiert durch eine gerichtete Kante) oder durch einen logischen Verknüpfungsoperator (UND, ODER, exklusives ODER) mit gerichteten Kanten.

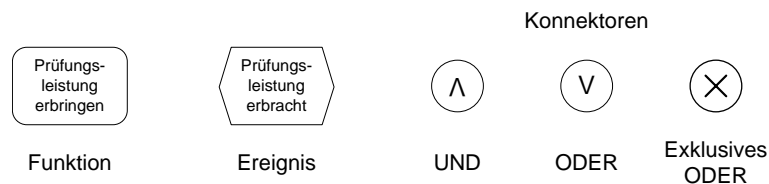


Abbildung 1: EPK-Konstrukte

Abbildung 1 zeigt die drei Hauptkategorien von EPK-Konstrukten: Funktionen, Ereignisse und Konnektoren. Die zulässigen Verknüpfungsregeln für Konnektoren sind in Abbildung 3 dargestellt (vgl. [IDS06], S. 110).

EPKs starten und enden stets mit einem Ereignis. In der Praxis kommt es auch vor, dass Modelle mehrere Startereignisse und mehrere Endereignisse haben.

Um die Ausdrucksfähigkeit nicht auf die Kontrollflussspezifikation zu beschränken, wurden die EPKs erweitert. Erweiterte EPKs (eEPKs) zeigen auch beteiligte Organisationseinheiten, Stellen oder Rollen (Organisationssicht), Ein- und Ausgabedaten (Datensicht), sowie die verwendeten Systeme (ein spezieller Aspekt der Funktionssicht) auf. Abbildung 2 zeigt ein eEPK-Beispiel für einen Teil des Prüfungsprozesses einer Hochschule.

Zur Beschränkung von Prozessen auf handhabbare Größen und auf überschaubare Ausschnitte bzw. Abstraktionsebenen werden Wertschöpfungskettendiagramme und Hinterlegungen verwendet. Wertschöpfungsketten abstrahieren EPKs und zeigen Abfolgen und Strukturen betrieblicher Funktionen auf höherer Ebene auf. Hinterlegungen sind Verfeinerungen von Funktionen, die diese mit den Mitteln der eEPK auf nächster Verfeinerungsstufe beschreiben.

Für die Nennung von Nachfolgeprozessen und Vorgängerprozessen in EPKs werden in der Praxis oft Prozessschnittstellen eingesetzt, die diese Prozesse symbolisieren. In manchen existierenden Prozessmodellen werden Prozessschnittstellen auch zur Darstellung von Unterprozessen eingesetzt. Diese Verwendung wird allerdings nicht empfohlen, da hierfür bereits das Konstrukt der Hinterlegung existiert. Generell sind Prozessschnittstellen

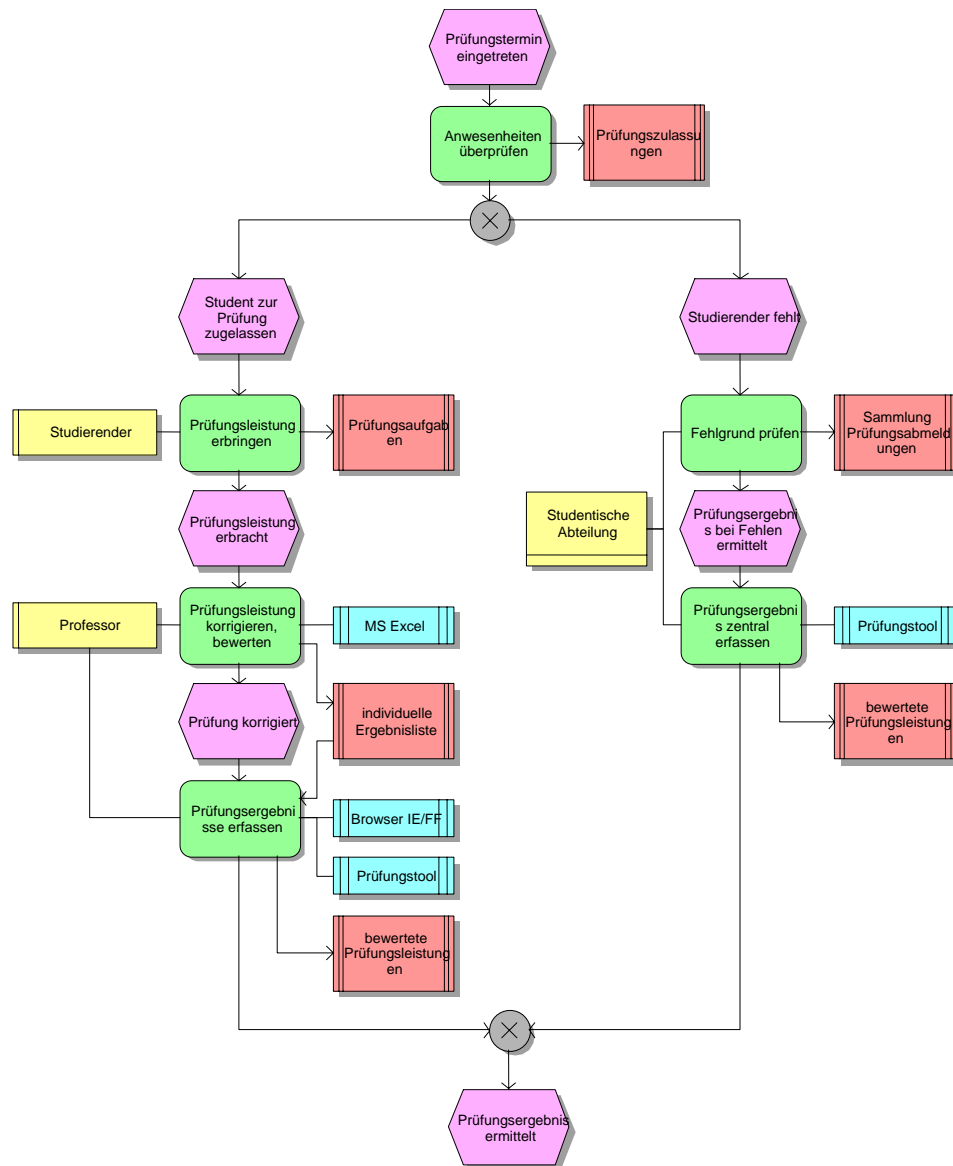


Abbildung 2: EPK-Beispiel

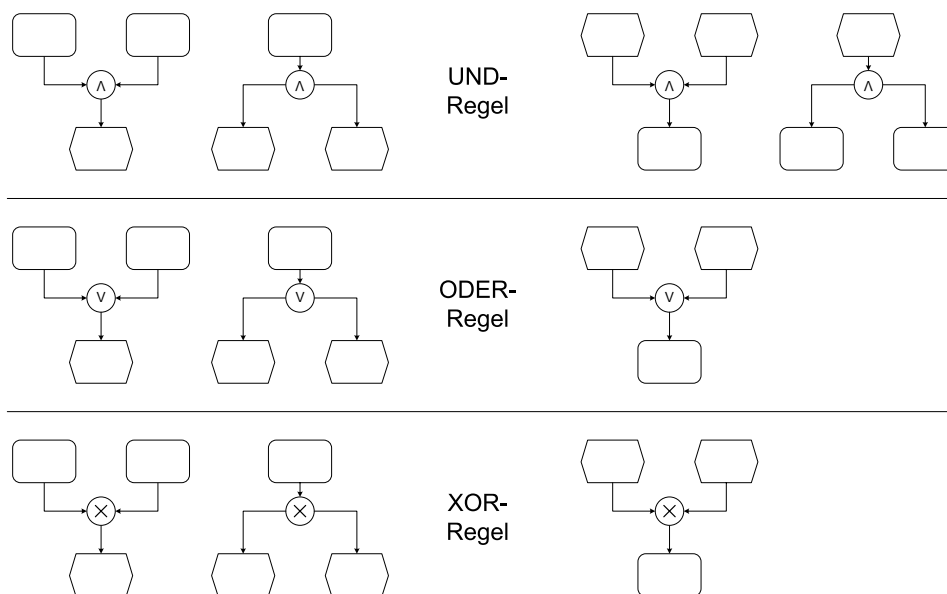


Abbildung 3: Verknüpfungsregeln nach [IDS06]

zurückhaltend einzusetzen, da sie Informationen redundant darstellen. Eine Definition des Konstrukts Prozessschnittstelle findet sich im Methodenhandbuch von ARIS [IDS06] nicht.

4 BPMN im Überblick

Der folgende Überblick über die Business Process Modeling Notation (BPMN) basiert auf der im Februar 2009 veröffentlichten Version 1.2 [bpm09].

Abbildung 4 zeigt die vier Hauptkategorien an BPMN-Konstrukten: Swimlanes, verknüpfende Objekte, Flussobjekte und Artefakte. Swimlanes repräsentieren Organisationseinheiten, Rollen oder Systeme. Eine Unterscheidung, ob es sich um eine Organisationseinheit oder eine Rolle handelt, wird nicht unterstützt. Die häufigste Verwendung ist dabei jedoch, dass Pools Organisationen darstellen und Lanes Organisationseinheiten darin.

BPMN unterscheidet Sequenzfluss und Nachrichtenfluss. Sequenzfluss stellt die Verhaltensabhängigkeiten innerhalb eines Pools dar, während Nachrichtenfluss die Kommunikation verschiedener Pools miteinander repräsentiert. Nachrichtenfluss innerhalb des gleichen Pools ist nicht erlaubt.

Es gibt drei Arten von Flussobjekten in BPMN: Aktivitäten repräsentieren die Arbeitsschritte innerhalb eines Prozesses. Diese sind entweder atomar (Tasks) oder können in Unterschritte (Unterprozesse) weiter unterteilt werden.

Ereignisse gibt es in den Ausprägungen "auslösend" und "eintretend". Bei auslösenden Er-

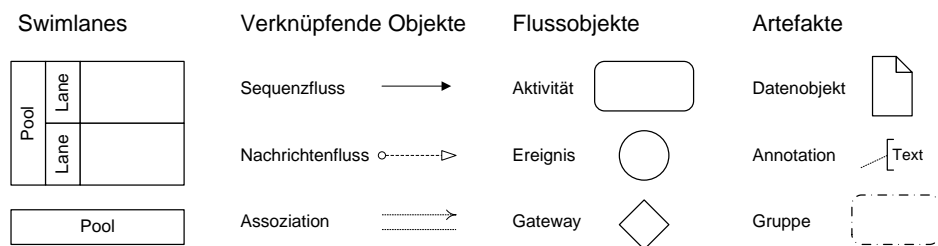


Abbildung 4: BPMN Modellierungskonstrukte

eignissen handelt es sich um spezielle Aktionen, z.B. werden Nachrichten an einen anderen Pool verschickt. Bei eintretenden Ereignissen wird der Prozessfluss solange blockiert, bis das entsprechende Ereignis eingetreten ist, z.B. bis eine Nachricht eingetroffen ist oder bis eine Zeitbedingung erreicht wurde.

Gateways werden für die Spezifikation von Kontrollflusslogik verwendet. Hier gibt es die fünf Typen daten-basiert exklusiv, ereignis-basiert exklusiv, parallel, inklusiv und komplex. Auf ein ereignis-basiertes Gateway folgt immer eine Reihe von eintretenden Ereignissen. Wird das Gateway erreicht, blockiert der Prozessfluss und es wird auf die Ereignisse gewartet. Sobald ein Ereignis eingetreten ist, wird nicht weiter auf die anderen Ereignisse gewartet und der Prozessfluss setzt sich an der Verzweigung fort entsprechend dem eingetretenen Ereignis. Dieses Verhalten wird im Workflow-Patterns-Katalog als “Deferred Choice” bezeichnet [AtHKB03].

BPMN bietet einige besondere Kontrollflusskonstrukte, wovon wir zwei ausgewählte kurz erwähnen. (1) Es können “Multiple Instanzen” spezifiziert werden, bei denen eine Menge von Instanzen von der gleichen Aktivität parallel ausgeführt werden können. Beispielsweise kann so spezifiziert werden, dass für jeden Bestellposten innerhalb einer Bestellung eine bestimmte Aktion nötig ist. (2) Weiterhin können mittels angehefteter Zwischen-Ereignisse Abbruchkriterien für Aktivitäten spezifiziert werden. Zum Beispiel können Eskalationsmaßnahmen definiert werden für den Fall, dass eine Aktivität nicht innerhalb einer bestimmten Zeit abgeschlossen wird.

Bei BPMN ist es möglich, mehrere Ende-Ereignisse zu verwenden. Abgesehen vom speziellen Terminierungs-Ende-Ereignis, bei dem die laufende Prozessinstanz als Ganzes abgebrochen wird, läuft eine Prozessinstanz solange weiter, bis keine Aktion mehr möglich ist. Dies wird auch “Implicit Termination” genannt [AtHKB03].

Abbildung 5 zeigt das Beispiel aus dem vorigen Abschnitt in BPMN-Darstellung.

5 Transformation von EPK zu BPMN

Dieser Abschnitt untersucht die Transformation von EPK zu BPMN. Dabei sind für jedes Konstrukt jeweils zwei Aspekte zu prüfen.

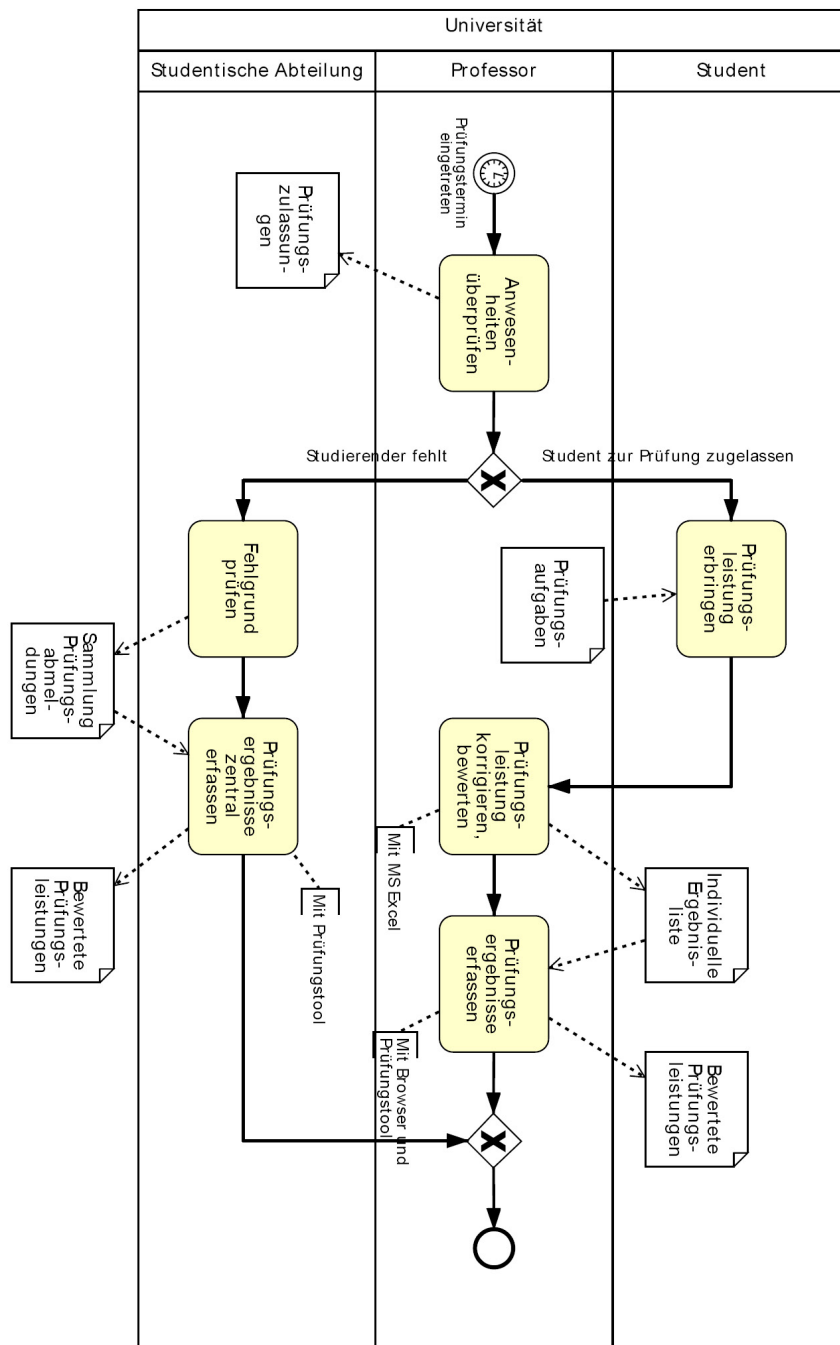


Abbildung 5: BPMN-Beispiel

1. Existiert ein BPMN-Konstrukt (oder eine Kombination von BPMN-Konstrukten), das die Semantik des EPK-Konstruktes abdeckt? Falls nicht, welche Konstrukte kommen der Semantik möglichst nahe?
2. Ist die Abbildung injektiv, d.h. werden keine zwei EPK-Konstrukte auf das gleiche BPMN-Konstrukt abgebildet? Falls eine nicht-injektive Abbildung vorliegt, handelt es sich um einen echten Informationsverlust?

Beide Fragen beschäftigen sich damit, ob der Informationsgehalt aus der EPK vollständig erhalten werden kann. Dies ist wichtig, da Modelle vor allem zur Beantwortung von Fragen dienen. Gehen Informationen verloren, kann dies zur Folge haben, dass bestimmte Fragen im Zielmodell nicht im gleichen Maße beantwortet werden können, wie dies im Quellmodell der Fall war.

Abbildung 6 zeigt eine tabellarische Übersicht über alle betrachteten EPK-Konstrukte und deren Entsprechungen in BPMN.

5.1 Funktionen

Funktionen repräsentieren einen Arbeitsschritt innerhalb des Prozesses und finden somit in BPMN-Tasks ihre Entsprechung. Ein Sonderfall ergibt sich im Falle von Hinterlegungen. Ist der Funktion ein Prozessmodell hinterlegt, so entspricht dies einem zugeklappten Subprozess in BPMN, der auf ein weiteres Diagramm zeigt.

Aktionen werden in BPMN nicht ausschließlich über Tasks und Subprozesse abgebildet. Auslösende Ereignisse repräsentieren kurzlebige Aktionen innerhalb eines Prozesses. Somit erlaubt BPMN eine weiterführende Typisierung von Aktionen. In ausgewählten Fällen können daher Funktionen mit auslösenden Zwischen- und Ende-Ereignissen übersetzt werden. Beispielsweise kann eine Funktion "Bescheid verschicken" als Zwischen- oder Ende-Nachrichten-Sende-Ereignis dargestellt werden. Eine detaillierte Diskussion zu dieser Unterscheidung ist in [HBS07] zu finden.

5.2 Ereignisse

Ereignisse in EPKs repräsentieren zumeist Zustände. So ist es ein typisches Beispiel, dass die Funktion "Prüfungsleistung erbringen" von einem Ereignis "Prüfungsleistung erbracht" gefolgt wird. In diesem Fall trägt das Ereignis keine nennenswerte Mehrinformation und scheint nur ein syntaktisches Füllelement zu sein. In BPMN ist eine explizite Darstellung von Prozesszuständen im Regelfall nicht vorgesehen. Der Fokus liegt vielmehr auf den Aktivitäten. Daher werden die meisten EPK-Ereignisse in einem Mapping auf BPMN ignoriert.

Soll ein Prozesszustand zwingend dargestellt werden, so stehen in BPMN Datenobjekte mit Zustandsattributen zur Verfügung. Ein Datenobjekt "Akte" könnte sich dabei im Zustand

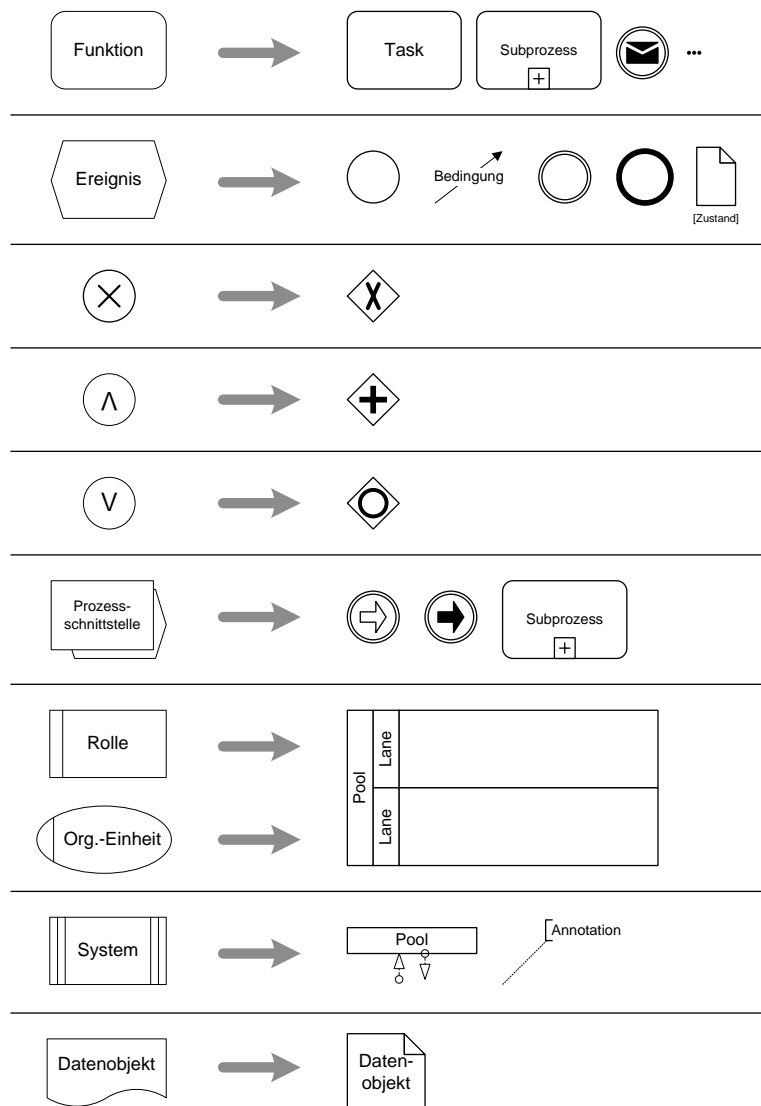


Abbildung 6: Transformation von EPK nach BPMN – Übersicht

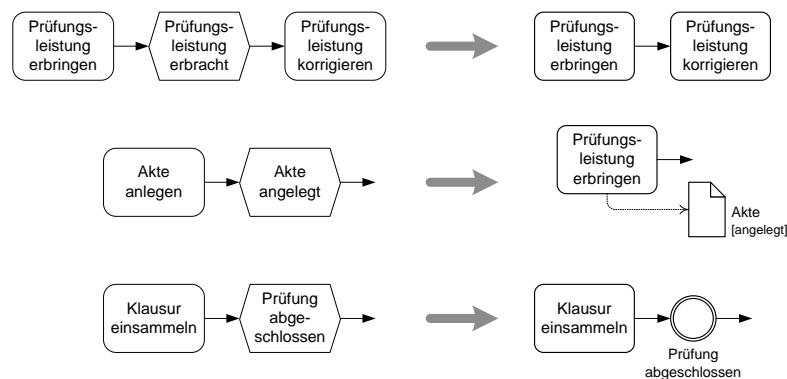


Abbildung 7: Abbildung von Zwischenereignissen

“angelegt” befinden. Als Alternative für die Darstellung von Prozesszuständen bieten sich weiterhin Blanko-Zwischenereignisse an. Diese Verwendung ist in der Praxis jedoch selten anzutreffen. Ein guter Hinweis darauf, dass Prozesszustände tatsächlich eine wichtige Information im Modell darstellen, besteht in folgendem Fall: Eine Funktion wird gefolgt von einem XOR- oder UND-Konnektor, dieser von einer Reihe von Ereignissen und diese wiederum von einem weiteren XOR- oder UND-Konnektor, der den Prozessfluss wieder zusammenführt. Hier haben die Konnektoren keinerlei Einfluss auf den Prozessfluss. Vielmehr findet hier eine explizite Aufzählung von möglichen Prozesszuständen statt. Zu diskutieren wäre hier allerdings, ob es sich hier um guten Modellierungsstil handelt, da die Aufspaltung im Kontrollfluss zu keinem Unterschied bzgl. der Funktionsausführungen führt.

Spannend wird es bei Startereignissen. Hier repräsentieren EPK-Ereignisse entweder Trigger oder Zustände [DM09]. Bei “Prüfungstermin eingetreten” handelt es sich um einen Trigger. In dem Moment, wo der Termin eintritt, wird der Prozess ausgelöst. In BPMN werden mehrere Typen von Triggern unterschieden. Der Typ kann lediglich durch die Beschriftung des Ereignisses oder durch den Prozesskontext ermittelt werden. Beispielsweise würde “Jeden Montagmorgen” auf ein zeitliches Ereignis hindeuten, “Zahlung ist eingegangen” auf eine eingehende Nachricht oder “Wasserstand hat kritisches Level erreicht” auf eine Business-Regel. Ist ein Ereignis-Typ nicht eindeutig auszumachen, so kann auch einfach ein untypisiertes Start-Ereignis (Blanko-Ereignis) verwendet werden.

“Kunde ist volljährig” ist ein Beispiel für eine Zustandsbeschreibung. Hier wird der Prozess nicht in dem Moment ausgelöst, wo der Kunde das 18. Lebensjahr vollendet. Eher ist diese Zustandsbeschreibung als Vorbedingung für die Prozessinstanziierung zu verstehen.

Sofern eine EPK nur über ein einziges Startereignis verfügt, ist eine Transformation offensichtlich: Im BPMN-Modell wird ein Startereignis erzeugt. Die einzige Frage ist, ob eine weitere Typisierung des Triggers sinnvoll ist.

Verfügt eine EPK über mehrere Startereignisse, so ist eine gültige Transformation nur in bestimmten Fällen möglich. Idealerweise handelt es sich um alternative Startereignisse, d.h. es erfolgt später eine Zusammenführung mittels XOR-Konnektoren. In diesem Fall

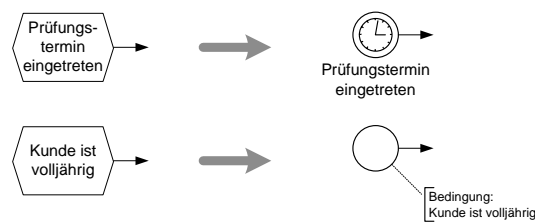


Abbildung 8: Abbildung von Startereignissen

kann jedes EPK-Startereignis auf ein BPMN-Startereignis abgebildet werden. Ein Beispiel für alternative Eintrittspunkte in einen Vertriebsprozess wären “Kunde ruft an” vs. “Messekontakt wurde protokolliert”.

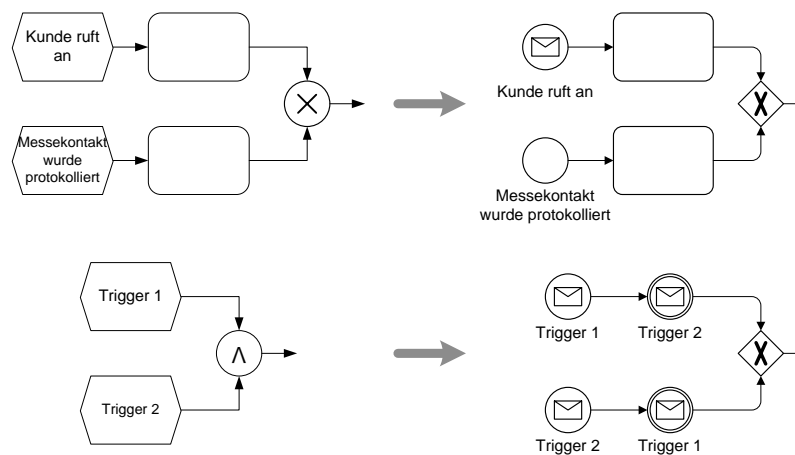


Abbildung 9: Abbildung von verknüpften Startereignissen

Sofern es sich nicht um alternative EPK-Startereignisse handelt, so ist anhand der Beschriftungen zu prüfen, ob es unter den Startereignissen mehr als einen Trigger gibt. Ist dies nicht der Fall, so wird das triggernde EPK-Startereignis wiederum mit einem BPMN-Startereignis übersetzt. Gibt es mehrere Trigger, die später über UND- oder ODER-Konnektoren zusammen geführt werden, ist eine korrekte Transformation in ein BPMN-Modell unter Beibehaltung einer vergleichbaren syntaktischen Struktur nicht möglich. Die parallele Struktur muss entsprechend sequenzialisiert werden, was eine Duplizierung von Ereignissen im Diagramm zur Folge hat (siehe Abbildung 9) und das Diagramm schwer lesbar macht. Eine detaillierte Diskussion zu diesem Fall findet sich bei Decker und Mendling [DM09].

BPMN-Prozessmodelle schließen mit Ende-Ereignissen ab. Im häufigsten Fall wird ein Blanko-Ende-Ereignis verwendet, welches optional mit einer kurzen Zustandsbeschreibung versehen ist, z.B. “Vertrag ist abgeschlossen”. Daher können EPK-Endereignisse einfach in BPMN-Endereignisse übersetzt werden.

An dieser Stelle sei darauf hingewiesen, dass Zwischen- und Endereignisse auch in bestimm-

ten Fällen auf Nachrichten-Empfangs-Ereignisse abgebildet werden können [KWL09].

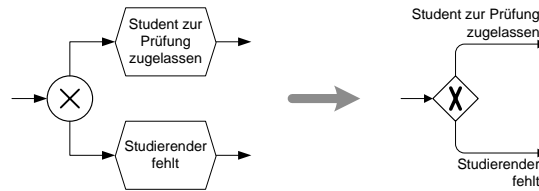


Abbildung 10: Abbildung von Ereignissen (4)

Bei denjenigen EPK-Ereignissen, die weder Start- noch Endereignisse sind, d.h. eingehende oder ausgehende Kontrollflusskanten besitzen, gibt es einen speziellen Fall, in dem das Ereignis auf jeden Fall in der Transformation beachtet werden muss: Entscheidungen werden in EPK über XOR- und ODER-Konnektoren mit mehreren ausgehenden Kanten modelliert. In diesem Fall geben die nachfolgenden Ereignisse die Verzweigungsbedingungen an, z.B. “Student zur Prüfung zugelassen”. Verzweigungsbedingungen werden in BPMN über Kantenbeschriftungen realisiert.

5.3 Konnektoren

Konnektoren realisieren Kontrollflussbeziehungen in EPKs. XOR-Konnektoren finden in daten-basierten exklusiven Gateways ihre Entsprechung, UND-Konnektoren in parallelen Gateways und ODER-Konnektoren in inklusiven Gateways. Während bei der semantischen Entsprechung im Falle von XOR- und UND-Konnektoren mit den entsprechenden BPMN-Konstrukten keine Zweifel bestehen, so ist der ODER-Konnektor insofern problematisch, da sowohl auf EPK- als auch auf BPMN-Seite Unsicherheit bzgl. der Semantik besteht (siehe Abschnitt 2). Praktische Relevanz haben die verschiedenen Nuancen in der Ausführungssemantik von ODER-Konnektoren / inklusiven Gateways aber wahrscheinlich nicht: Es besteht die Vermutung, dass für die allermeisten Prozessmodelle diese Nuancen keinen Unterschied in der Semantik bedeuten. Ein entsprechender empirischer Nachweis fehlt in der Literatur allerdings noch.

Zu beachten bei einem Mapping von Konnektoren ist lediglich, dass die Verzweigungsbedingungen in EPKs in Form von Ereignissen gegeben sind, während BPMN hierfür Kantenbeschriftungen vorsieht. Darüber hinaus sieht BPMN alternative Darstellungen von XOR-Joins, AND-Splits und OR-Splits vor: Hier werden mehrere Sequenzflusskanten mit Flussobjekten verbunden. Beispielsweise haben mehrere eingehende Kanten in einen Task XOR-Semantik. Eine Abbildung auf BPMN-Gateways, wie oben beschrieben, kann jedoch als üblicher Weg verwendet werden.

5.4 Prozessschnittstellen

Prozessschnittstellen bieten die Möglichkeit, ein EPK-Modell mit anderen Modellen in Beziehung zu setzen. Hier gibt es jedoch zwei semantische Ausprägungen, die jeweils unterschiedlich in BPMN abgebildet werden müssen. Zum einen werden Prozessschnittstellen verwendet, um Prozesshierarchien darzustellen. Hierbei verweist eine Prozessschnittstelle auf ein Modell als Ganzes oder auf ein Startereignis in einem anderen Modell. Die Semantik hierbei ist, dass der zweite Prozess beendet sein muss, bevor die auf die Prozessschnittstelle folgenden Funktionen angestoßen werden. Die Entsprechung auf BPMN-Seite wäre damit ein Unterprozess.

Alternativ realisieren Prozessschnittstellen zuweilen jedoch auch einfache Zerschneidungen von Prozessmodellen. Hierbei wird auf eine Prozessschnittstelle in einem anderen Modell verwiesen. Semantisch ist ein Pärchen von Link-Ereignissen also einer Kontrollflusskante, sofern die beiden Modelle in einem großen Modell zusammengeführt werden. BPMN sieht Link-Ereignisse zur Darstellung solcher Modell-Zerschneidungen vor.

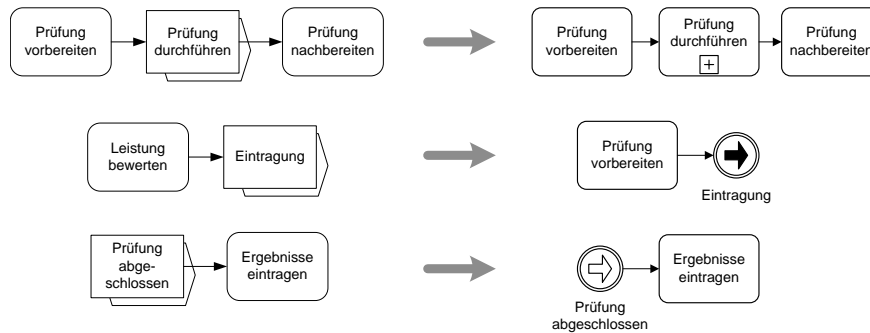


Abbildung 11: Abbildung von Prozessschnittstellen

Bei der Wahl, ob nun ein Unterprozess oder ein Link-Ereignis die passende Entsprechung ist, helfen eine Reihe von Indizien. Besitzt eine Prozessschnittstelle Vorgänger- und Nachfolgerelemente, so handelt es sich um einen Unterprozess. Wird eine Prozessschnittstelle jedoch als Eintrittspunkt in einen Prozess oder als Austrittspunkt verwendet, so sind wahrscheinlich Link-Ereignisse die passende Entsprechung. Informationen darüber, was von der Schnittstelle verlinkt wird (ganzes Modell vs. Prozessschnittstellen) gibt einen zusätzlichen Hinweis.

5.5 Organisationseinheiten / Rollen

Organisationseinheiten und Rollen werden in BPMN durch Pools und Lanes dargestellt. Anders als in BPMN, wo durch Enthaltenseinsbeziehungen von Pools, Lanes und Sub-Lanes eine hierarchische Beziehung ausgedrückt werden kann, werden Organisationseinheiten und Rollen in EPKs nicht weiter in Beziehung gesetzt. Dies geschieht zumeist außerhalb

des Modells in separaten Organigrammen. Da weiterhin keine Unterscheidung zwischen Sequenz- und Nachrichtenfluss in EPKs zu finden ist, können die Organisationseinheiten und Rollen am besten als nebeneinander angeordnete Lanes des gleichen Pools abgebildet werden. Ein Bezeichner für diesen Pool steht nicht zur Verfügung. Als Behelfslösung können generische Namen wie “Hauptpool” oder “Organisation” gewählt werden.

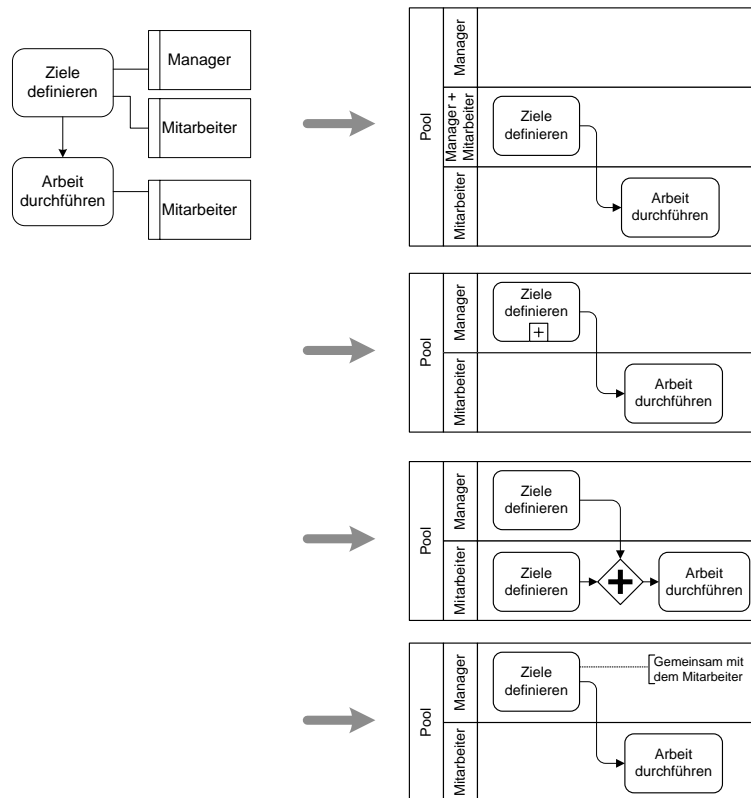


Abbildung 12: Abbildung von Rollen (und Organisationseinheiten)

Problematisch wird es, sobald mehrere Organisationseinheiten und/oder Rollen mit einer Funktion verknüpft sind. Dieser Sachverhalt kann in BPMN nicht sauber abgebildet werden. In BPMN-Diagramme sieht man des öfteren, dass eine Aktivität mehrere Lanes abdeckt. Dies ist von BPMN syntaktisch allerdings nicht vorgesehen. Verschiedene Workarounds sind hier in BPMN denkbar. (1) Zum einen können separate Lanes geschaffen werden, die explizit die Zusammenarbeit mehrerer Organisationseinheiten oder Rollen modellieren, z.B. “Team”. (2) Zum anderen ist es möglich, Unterprozesse zunächst einer bestimmten Lane zuzuordnen und bei einer Verfeinerung in einem separaten Diagramm die Aufgaben entsprechend den verschiedenen Lanes zuzuordnen. (3) Manchmal sieht man, dass eine gemeinschaftliche Aktivität dupliziert wird und mit gleicher Bezeichnung in verschiedenen Lanes erzeugt wird. (4) Die Aktivität wird nur einer Lane zugeordnet, und zwar der verantwortlichen oder führenden Organisationseinheit/Rolle, und über entsprechende

Textannotationen (Kommentare) wird die Beteiligung der anderen erwähnt. Abbildung 12 illustriert diese vier Strategien.

5.6 Systeme

Systeme werden typischerweise auf eine von vier Arten abgebildet. (1) Sofern das Modell eine reine Systemsicht liefert, d.h. Organisationseinheiten und Rollen kommen im Modell nicht vor, so können Systeme als Pools und Lanes dargestellt werden. Hier stellt sich wiederum die gleiche Frage wie bei Organisationseinheiten und Rollen: Wie kann dargestellt werden, dass mehrere Systeme einer Funktion zugeordnet sind. Schwieriger wird es außerdem, falls sowohl Systeme als auch Organisationseinheiten / Rollen in einem Modell definiert sind. (2) Alternativ können Systeme als zugeklappte Pools dargestellt werden, mit denen der Prozess Nachrichten austauscht. Von der Intuition her würde so z.B. repräsentiert, dass ein Mensch im Laufe einer Aktivität mit dem System über Eingabemasken und entsprechende Bildschirmanzeigen interagiert. (3) Als dritte Option kann ein Modellierer auch auf die Darstellung von Systemen in einem Modell verzichten. Mit Hilfe weiterer Modelle, die sich dann speziell mit einer Systemzuordnung beschäftigen, kann die Systemsicht wiederum abgedeckt werden. Es entstehen somit mehrere Perspektiven auf den gleichen Prozess (organisatorische Sicht vs. IT-Sicht). (4) Abschließend, und wahrscheinlich am häufigsten verwendet, können auch Textannotationen verwendet werden, um die Systemzuordnung widerzuspiegeln. Diese vier Optionen sind in Abbildung 13 illustriert.

5.7 Datenobjekte

Für Datenobjekte gibt es eine offensichtliche Abbildung von EPKs auf BPMN: Über BPMN-Datenobjekte und entsprechend gerichtete Assoziationen kann die EPK-Semantik direkt abgebildet werden.

6 Diskussion

Im vorigen Abschnitt wurden alternative Optionen vorgestellt, wie die verschiedenen EPK-Konstrukte abgebildet werden können. Die Wahl einer geeigneten Option kann bei einer Migration einer größeren Anzahl von Modellen oft ein einziges Mal für alle Modelle entschieden werden. Dies ist vor allem bei der Abbildung von mehreren Rollen / Organisationseinheiten oder bei der Abbildung von Systemen der Fall. Auch könnte z.B. entschieden werden, dass Ereignisse wenn immer möglich in einer Abbildung ignoriert werden.

Auch kann bei einer Abbildung behilflich sein, wenn die EPKs definierten Modellierungsrichtlinien folgen. Dies kann z.B. klären, wie Prozessschnittstellen bei einer Abbildung zu behandeln sind.

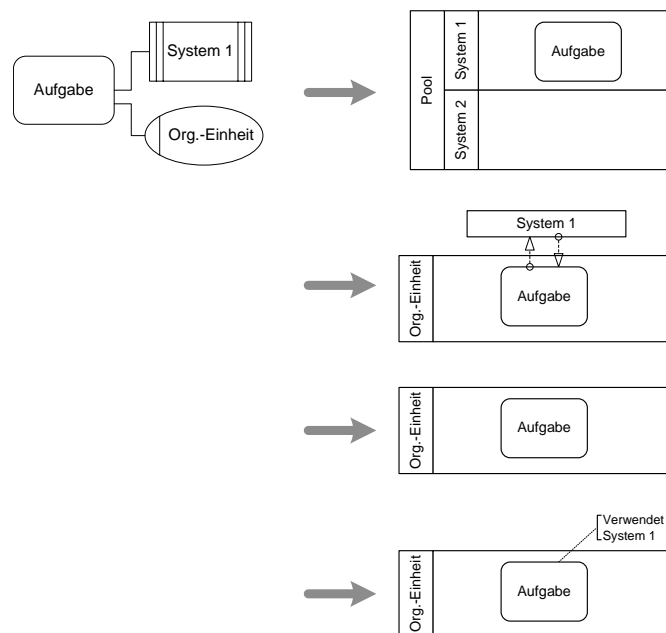


Abbildung 13: Abbildung von Systemen

In der ARIS-Methode sind EPKs nur eine von zahlreichen Diagrammtypen. Weitere häufig verwendete Diagrammtypen sind z.B. Wertschöpfungsketten und Organigramme. Organigramme können in BPMN nicht dargestellt werden. Zwar bietet BPMN die Möglichkeit, Hierarchien an Lanes zu definieren. Dies deckt jedoch nicht die Ausdrucksmächtigkeit von Organigrammen ab. Wertschöpfungskettendiagramme, in denen Prozesse landkartenartig aufgezählt werden, können in BPMN insofern abgebildet werden, als dass eine Menge von Subprozessen nebeneinander im Diagramm platziert werden.

Dieser Beitrag beschäftigt sich lediglich mit dem unidirektionalen Mapping von EPK nach BPMN. Es wird keine Antwort darauf gegeben, wie ein BPMN-Diagramm in eine EPK überführt werden kann. Ein Round-Tripping, bei dem das Modell in der Zielsprache editiert wird und wieder zurück in die Quellsprache überführt wird, bleibt somit erst recht undiskutiert.

Auch beantwortet dieser Artikel nicht die Frage "Wann setze ich welche Sprache ein?", bzw. die Frage nach Vor- und Nachteilen der beiden Sprachen in den unterschiedlichen Phasen des Entwurfs prozessorientierter Informationssysteme. Diese Fragen werden zu Beginn von BPM-Initiativen häufig gestellt. Dieser Beitrag hilft bei dieser Diskussion insofern, als dass er einen ersten Anhaltspunkt dafür bietet, wie im Falle der Entscheidung, zunächst mit EPKs zu arbeiten und später auf BPMN umzusteigen, hinterher eine Migration der Modelle aussehen kann.

Dieser Beitrag hat sich auf BPMN in der Version 1.2 konzentriert. Im Laufe des Jahres 2010 kann mit der Veröffentlichung der neuen Version 2.0 gerechnet werden. Die allermeisten

Konvertierungsregeln bleiben davon unberührt. Lediglich bei der Darstellung von Systemen erleichtert BPMN 2.0 eine Abbildung. Durch “Data Stores” sowie über selbst definierbare Artefakte können Systeme damit abgebildet werden.

7 Zusammenfassung und Ausblick

Dieser Beitrag hat gezeigt, dass weite Teile der EPKs automatisch bzw. semi-automatisch in bedeutungsgleiche BPMN-Modelle überführt werden können. Eine vorherige sachkundige Qualitätssicherung der EPKs sowie geeignete Modellierungskonventionen, die wiederum automatisch (z.B. mit sog. Reports) überprüft werden können, erhöhen zusätzlich das Einsatzpotenzial der automatischen Umstellung. Durch dieses Vorgehen können die in EPKs getätigten Investitionen gesichert werden, wenn das Anwenderunternehmen von EPKs auf BPMN wechseln möchte oder muss.

Eine Organisation sollte jedoch prüfen, ob nicht auch eine erneute BPMN-Modellierung, gepaart mit einer Anpassung und Aktualisierung der Modelle eine valide Alternative bildet. Dies hat gleichzeitig den Effekt, dass die Verwendung der BPMN eingeübt werden kann.

Eine Teilmenge der vorgestellten Konvertierungsregeln wurde im Rahmen der Signavio-Oryx Academic Initiative² als automatische Transformation realisiert. Dabei wird jede Funktion als Task, jedes Startereignis als Blanko-Start-Ereignis und jedes Endereignis als Blanko-Ende-Ereignisse umgesetzt. Zwischenereignisse werden ignoriert – außer im Falle von Verzweigungsbedingungen. Prozessschnittstellen werden bislang nicht unterstützt und bei Rollen/Organisationseinheiten wird pro Funktion einfach nur eine Rolle/Organisationseinheit ausgewählt (und die anderen werden ignoriert). Als nächster Schritt wird eine semi-automatische bzw. durch Konventionen konfigurierbare Transformation realisiert, um den Entscheidungspunkten gerecht zu werden, die in diesem Beitrag beschrieben wurden.

Literatur

- [AtHKB03] Wil M. P. van der Aalst, Arthur H. M. ter Hofstede, Bartek Kiepuszewski und Alistair P. Barros. Workflow Patterns. *Distributed and Parallel Databases*, 14(1):5–51, 2003.
- [bpm09] Business Process Modeling Notation (BPMN), Version 1.2. Bericht, Object Management Group (OMG), Feb 2009. <http://www.omg.org/spec/BPMN/1.2/>.
- [CKS08] Mary Beth Chrissis, Mike Konrad und Sandy Shrum. *CMMI, Guidelines for Process Integration and Product Improvement, Second Edition*. Addison Wesley, 2008.
- [DGHW07] Marlon Dumas, Alexander Grosskopf, Thomas Hettel und Moe Wynn. Semantics of BPMN Process Models with OR-joins. In *Proceedings of 15th International Conference on Cooperative Information Systems (CoopIS)*, Jgg. 4803 of LNCS, Seiten 41–58, Vilamoura, Portugal, November 2007. Springer Verlag.

²See <http://www.signavio.com/academic>

- [DM09] Gero Decker und Jan Mendling. Process Instantiation. *Data & Knowledge Engineering*, 2009.
- [Gad08] Gadatsch. *Grundkurs Geschäftsprozess-Management*, 5. Auflage. Vieweg, 2008.
- [HBS07] Volker Hoyer, Eva Bucherer und Florian Schnabel. Collaborative e-Business Process Modelling: Transforming Private EPC to Public BPMN Business Process Models. In *Business Process Management Workshops*, Jgg. 4928 of *LNCS*, Seiten 185–196, Brisbane, Australien, 2007. Springer Verlag.
- [HW08] Stefan Huth und Thomas Wieland. *Geschäftsprozessmodellierung mittels Software-Services auf Basis der EPK*, Seiten 61–76. 2008.
- [IDS06] ARIS Platform, Methode ARIS 7.0. Bericht, IDS Scheer, Saarbrücken, 2006.
- [IT-05] COBIT 4.1. Bericht, IT-Governance Institute, 2005.
- [KMWL09] Oliver Kopp, Daniel Martin, Daniel Wutke und Frank Leymann. The Difference Between Graph-Based and Block-Structured Business Process Modelling Languages. *Enterprise Modelling and Information Systems*, 4(1):3–13, June 2009.
- [KWL09] Oliver Kopp, Matthias Wieland und Frank Leymann. External and Internal Events in EPCs: e2EPCs. In *2nd International Workshop on Event-Driven Business Process Management (edBPM09)*. Springer Verlag, 2009.
- [Men07] Jan Mendling. On the Detection and Prediction of Errors in EPC Business Process Models. *EMISA Forum*, 27(2):52–59, 2007.
- [Sch92] August-Wilhelm Scheer. *Architektur integrierter Informationssysteme - Grundlagen der Unternehmensmodellierung*, 2. Auflage. Berlin, 1992.
- [Sch98] August-Wilhelm Scheer. *ARIS – Modellierungsmethoden, Metamodelle, Anwendungen*, 3. Auflage. Berlin, 1998.
- [SJ02] August-Wilhelm Scheer und W. Jost. *ARIS in der Praxis, Gestaltung, Implementierung und Optimierung von Geschäftsprozessen*. Berlin, Heidelberg, New York, 2002.
- [SKI08] Sebastian Stein, S. Kühne und K. Ivanov. Business to IT Transformations Revisited. In *MDE4BPM 2008*, 2008.
- [SKLN09] David Schumm, Dimka Karastoyanova, Frank Leymann und Jörg Nitzsche. On Visualizing and Modelling BPEL with BPMN. In *Proceedings of the 4th International Workshop on Workflow Management (ICWM2009)*. IEEE Computer Society, 2009.
- [VZHA05] D. Vanderhaeghen, S. Zang, A. Hofer und O. Adam. XML-based Transformation of Business Process Models–Enabler for Collaborative Business Process Management. *XML4BPM*, 2005.
- [WDGW08] Matthias Weidlich, Gero Decker, Alexander Grosskopf und Mathias Weske. BPEL to BPMN: The Myth of a Straight-Forward Mapping. In *Proceedings 16th International Conference on Cooperative Information Systems (CoopIS)*, Jgg. 5331 of *LNCS*, Seiten 265–282, Monterrey, Mexico, Nov 2008. Springer Verlag.