

Transformation von Prozessmodellen in Workflowbeschreibungen¹

Daniel Beer, Jörg Dümmler und Gudula Rünger

*Technische Universität Chemnitz, Fakultät für Informatik 09107 Chemnitz
Kontaktautor: Jörg Dümmler, Email: djo@informatik.tu-chemnitz.de*

Schlagworte: Prozessmodellierung, Workflowmanagement, Anwendungssoftware, Electronic Government

Abstract: In diesem Artikel wird ein transformationsbasiertes Übersetzungswerkzeug vorgestellt, das Prozessmodelle auf Basis ereignisgesteuerter Prozessketten halbautomatisch in elektronisch verarbeitbare Workflowbeschreibungen umwandeln kann. Der Übersetzungsvorgang, der in mehreren Phasen realisiert ist, wird detailliert beschrieben und anhand von Beispielen verdeutlicht.

1. Einleitung

Die Modellierung von Arbeitsprozessen in Firmen oder Behörden lassen sich in zwei Richtungen mit unterschiedlicher Herangehensweise unterscheiden. Auf der einen Seite steht die Erstellung von Geschäftsprozessmodellen zur Dokumentation oder Analyse im Rahmen der Prozessoptimierung. Die erstellten Modelle eignen sich häufig auch zur Durchführung von Simulationen, um Schwachstellen aufzudecken, aber eine direkte rechnergestützte Abarbeitung ist dagegen meist nicht vorgesehen. Die Beschreibung von Geschäftsprozessmodellen erfolgt häufig durch *ereignisgesteuerte Prozessketten (EPK)* [GKS92], die von einer Reihe von Modellierungswerkzeugen, unter anderem dem *ARIS-Toolset* [Sch98], unterstützt werden.

Auf der anderen Seite steht die Modellierung von Arbeitsabläufen in Form von Workflows, die mit Hilfe eines Workflowmanagementsystems (WfMS) rechnergestützt abgearbeitet werden können. Das

¹ Das Projekt RAfEG wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) im Rahmen der Forschungsoffensive „Software Engineering 2006“ unter dem Förderkennzeichen 01 IS C07 C gefördert. Das Projekt setzte sich zusammen aus dem Institut für Wirtschaftsinformatik (IWi) im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI), der NHConsult GmbH, der Professur Praktische Informatik der Technischen Universität Chemnitz und dem Regierungspräsidium Leipzig. Weitere Informationen sind auf der Projekthomepage <http://www.rafeg.de> zu finden.

WfMS verwaltet die anstehenden Aufgaben sowie die Dokumente und Daten des zugrunde liegenden Prozesses und weist den beteiligten Mitarbeitern die zu bearbeitenden Teilaufgaben zu. Einzelne Teilaufgaben können direkt durch das WfMS ohne Nutzerinteraktion ausgeführt werden, wobei auch die Integration von externen Anwendungen möglich ist. Der Einsatz von WfMS kann zu einer Verringerung der Bearbeitungsdauer und damit einer Kostenreduktion beitragen und hält deshalb auch zunehmend in öffentlichen Verwaltungen Einzug. Zur Beschreibung von Workflows wird insbesondere in quelloffener Software, wie z.B. *Enhydra Shark*[Enh05], häufig die *XML Process Definition Language(XPDL)*[Coa02] eingesetzt.

Die unterschiedlichen Ziele der Geschäftsprozessmodellierung und der Workflowmodellierung drücken sich auch in den unterschiedlichen verwendeten Beschreibungsmodellen aus. Zur Überwindung der entstehenden Lücke wurde das Übersetzungswerkzeug *ConvEX* (Converter from Event-driven process chains to XPDL) entwickelt, das in diesem Artikel vorgestellt wird. Der Übersetzungsvorgang in *ConvEX* gliedert sich in mehrere Arbeitsphasen, in denen eine eingelesene Modellbeschreibung schrittweise transformiert wird. Die Steuerung erfolgt über eine grafische Benutzeroberfläche, die zusätzlich das Verändern und Hinzufügen von Attributen und Annotationen der Modellelemente erlaubt.

Dieser Artikel ist wie folgt gegliedert. In Kapitel 2. werden die grundlegenden Modelleigenschaften von EPKs und dem XPDL-Workflowmodell erläutert. Die Beschreibung der einzelnen Transformationsschritte erfolgt in Kapitel 3. Kapitel 4 stellt das Übersetzungswerkzeug *ConvEX* vor, während Kapitel 5 mit einer Zusammenfassung abschließt.

2. Modelleigenschaften

Geschäftsprozesse, die Arbeitsabläufe in Unternehmen oder Verwaltungen beschreiben, können durch **ereignisgesteuerte Prozessketten(EPK)**[GKS92] modelliert und analysiert werden. Eine EPK ist ein gerichteter Graph bestehend aus *Ereignisknoten*, *Funktionsknoten*, *Konektorknoten* und Kanten, die die Richtung des Kontrollflusses vorgeben. Ereignisknoten stellen einen Zustand des modellierten Geschäftsprozesses dar, der unter bestimmten Bedingungen eintreten kann. Funktionsknoten sind mit dem Ausführen einer Aufgabe assoziiert, wobei dies entweder eine atomare Tätigkeit oder das Abarbeiten eines Teilgeschäftsvorgangs ist, der wiederum durch eine EPK dar-

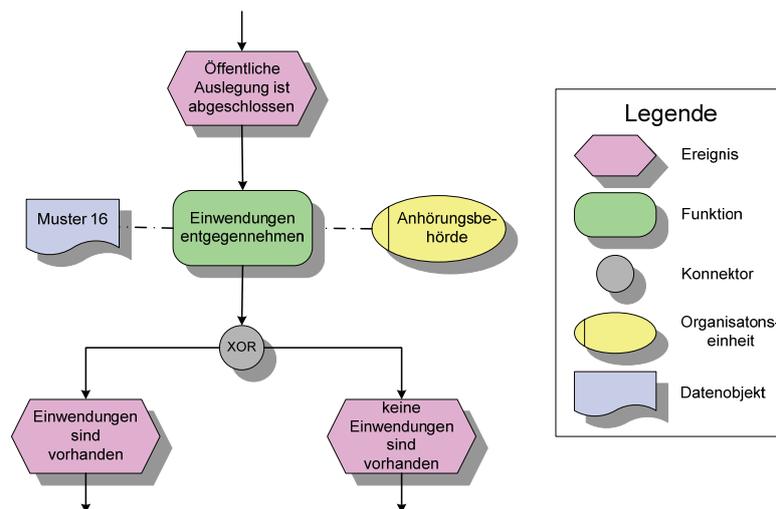


Abbildung 1: Beispiel einer erweiterten ereignisgesteuerten Prozesskette(links) und Überblick der verwendeten Knotentypen(rechts). Das alternative Eintreten der Ereignisse „Einwendungen sind vorhanden“ und „keine Einwendungen sind vorhanden“ steuert den Kontrollfluss.

gestellt wird. Eine grundsätzliche Regel besagt, dass Ereignisknoten und Funktionsknoten innerhalb einer EPK abwechselnd auftreten, also Ereignisknoten Funktionen auslösen, die wiederum Ereignisse als Ergebnis liefern. Konnektorknoten dienen dem Aufspalten und Zusammenführen des Kontrollflusses, z.B. wenn eine Funktion mehrere mögliche Ergebnisse haben kann. Den Konnektorknoten ist eine logische Operation zugeordnet, die festlegt, wie der Kontrollfluss von den eingehenden Kanten auf die auslaufenden Kanten verteilt wird. Als Operationen stehen AND (alle Fälle treten ein), OR (mindestens ein Fall tritt ein) und XOR (genau ein Fall tritt ein) zur Verfügung. Erweiterte EPKs (eEPKs)(s. bspw. [Sta01]) erlauben zusätzlich die Modellierung von Organisationseinheiten und Datenobjekten als Annotationen von Funktionsknoten. Abbildung 1 zeigt ein Beispiel einer EPK. Zur Beschreibung von Workflows wurde die **XML Process Definition Language (XPDL)**[Coa02] durch die Workflowmanagement Coalition, einem herstellerunabhängigen Gremium, spezifiziert. Ein Workflow in XPDL besteht aus Aktivitätsknoten und Kanten, denen Bedingungen

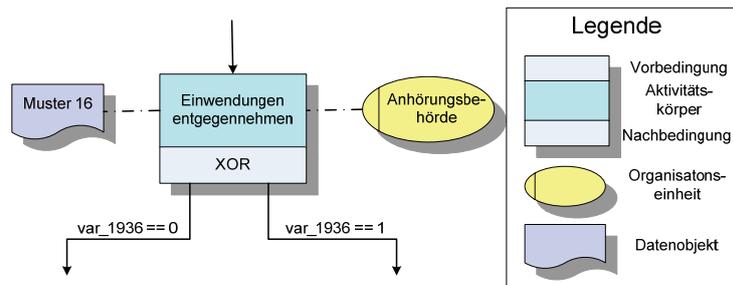


Abbildung 2: Beispiel einer transformierten Workflowbeschreibung im XPDL-Format (links) und bersicht der verwendeten Knotentypen (rechts). Der Inhalt der Prozessvariablen *var_1936* steuert den weiteren Ablauf des Workflows.

zugeordnet werden konnen, d.h. der Kontrollfluss kann nur uber eine Kante weitergegeben werden, wenn die zugehorige Bedingung wahr ergibt. Da diese Bedingungen innerhalb eines WfMS ausgewertet werden, ist eine spezielle Notation notwendig. Zur Angabe der Kantenbedingungen konnen Prozessvariablen, in denen der aktuelle Zustand des Workflows gespeichert werden kann, eingesetzt werden. Als Aktivitatsknoten werden manuelle Aufgaben (ausgefuhrt durch einen Bearbeiter), Systemaufgaben (ausgefuhrt durch das WfMS oder eine externe Anwendung), Unterprozessaufgaben (Ausfuhren eines anderen Workflows als Teilprozess) und Route-Aktivitaten (keine zugeordnete Aufgabe, nur Steuerung des Kontrollflusses) unterschieden. Besitzt ein Aufgabenknoten mehrere eingehende(auslaufende) Kanten muss eine Vorbedingung(Nachbedingung) angegeben werden, die festlegt, wie die entsprechenden Kanten synchronisiert werden. Als Operationen stehen XOR und AND zur Verfugung. Abbildung 2 zeigt ein Beispiel eines XPDL-Workflows, der durch Transformation der EPK aus Abbildung 1 entstand.

3. Technische Realisierung des Transformationsprozesses

Prozessmodelle dargestellt durch EPKs und XPDL-Workflowbeschreibungen basieren auf hierarchischen, gerichteten Graphen. Unterschiede bestehen vor allem in der semantischen Bedeutung der verschiedenen Knotentypen und der Steuerung des Kontrollflusses. Der

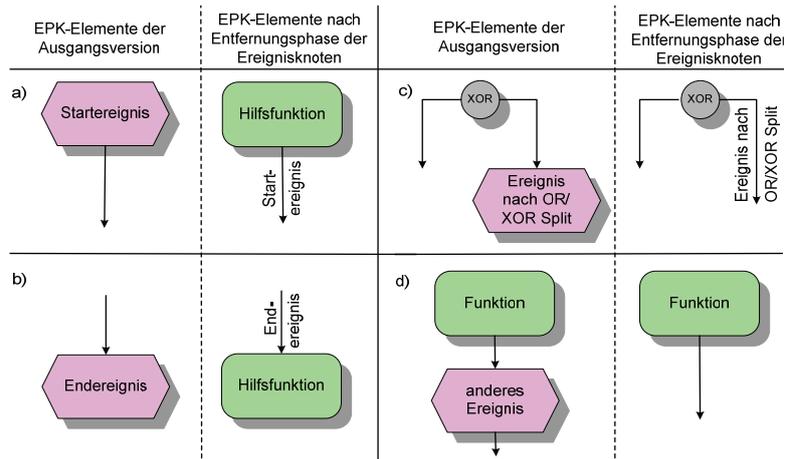


Abbildung 3: Elemente einer EPK vor und nach der Ereignisentfernungsphase für die Fälle a) Startereignis, b) Endereignis, c) Ereignis nach OR/XOR-Split und d) alle übrigen Fälle.

Transformationsprozess in *ConvEX* gliedert sich in mehrere Phasen, die nacheinander ausgeführt werden.

In der **EPK-Analysephase** wird ein EPK-Geschäftsprozessmodell eingelesen und in die interne Darstellung überführt. Zusätzlich werden fehlende Modellinformationen und Inkonsistenzen entdeckt. Weiterhin wird eine globale *Nutzernamentabelle* aufgebaut, die jeder Organisationseinheit eine eindeutige Identifikationsnummer zuordnet.

Die Ereignisknoten einer EPK dienen der Beeinflussung des Kontrollflusses. In XPDL gibt es kein natürliches Äquivalent zu diesem Knotentyp. Deshalb werden die Ereignisknoten in der **Entfernungsphase** aus der eingelesenen EPK entfernt. Dabei werden vier Fälle unterschieden: Startereignisse ohne eingehende Kante und Endereignisse ohne ausgehende Kante werden durch eine Hilfsfunktion ersetzt und der symbolische Name an der entsprechenden Kante annotiert. Ereignisknoten, die nach einem OR-Split-Konnektor oder einem XOR-Split-Konnektor auftreten, können eine kontrollflussrelevante Entscheidung treffen. Deshalb wird der symbolische Name dieser Ereignisse nach der Entfernung an der angrenzenden Kante annotiert. Alle anderen Ereignisknoten werden ersatzlos entfernt. Abbildung 3 stellt die vier möglichen Fälle grafisch dar.

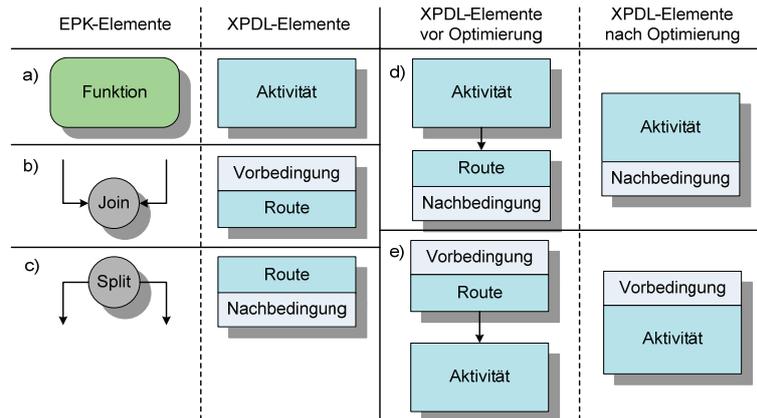


Abbildung 4: Darstellung der Transformation der Graphknoten in der Übersetzungsphase für die Knotentypen a) Funktion, b) Join-Konnektor und c) Split-Konnektor sowie die Verschmelzung von Knoten während der Optimierungsphase für die Fälle d) Aktivität und Route mit Nachbedingung und e) Route mit Vorbedingung und nachfolgender Aktivität.

In der **Übersetzungsphase** der Graphstrukturen werden die verbliebenen EPK-Modellelemente in ihre Äquivalente in XPDL übersetzt. Funktionsknoten einer EPK werden in Aktivitätsknoten transformiert, wobei die in der Entfernungsphase eingefügten Hilfsfunktionen in automatische Aktivitäten, Funktionen, die mit einer EPK hinterlegt sind, in Unterprozessaktivitäten und alle restlichen Funktionen in manuelle Aktivitäten übersetzt werden. Konnektorknoten werden in Route-Aktivitäten mit entsprechender Vor- bzw. Nachbedingung transformiert. Zusätzlich werden die Organisationseinheiten unter Vergabe einer neuen Identifikationsnummer aus der Nutzernamentabelle und die Datenobjekte in ihre XPDL-Äquivalente umgewandelt. Abbildung 4(links) verdeutlicht die Übersetzungsphase.

In der nachfolgenden **Optimierungsphase** werden redundante Knoten und Kanten durch Zusammenfassung von Aktivitätsknoten mit Route-Knoten entfernt. Dabei kann entweder eine Aktivität mit nachfolgendem Route-Knoten oder ein Route-Knoten mit nachfolgender Aktivität kombiniert werden. Abbildung 4(rechts) visualisiert diese beiden Optimierungsmöglichkeiten.

Zur Steuerung des Kontrollflusses werden in der **Erzeugungsphase** Prozessvariablen hinzugefügt. Zur Identifikation der benötigten Variab-

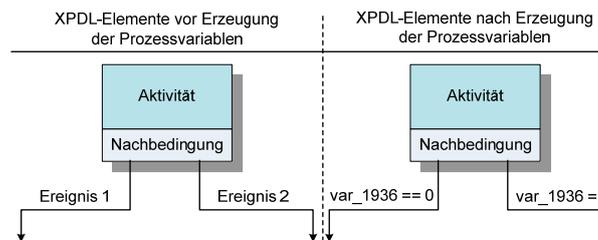


Abbildung 5: Ausschnitt eines XPDL-Workflows vor und nach der Erzeugung der Prozessvariablen.

len werden für jeden Aufgabenknoten alle möglichen Ergebnisse durch einen Tiefensuchlauf ermittelt. Können für einen Aufgabenknoten unterschiedliche Ergebnisse eintreten, wird eine Prozessvariable erzeugt, in der zur Ausführungszeit das tatsächliche Ergebnis gespeichert wird. Mit Hilfe der erzeugten Prozessvariablen und den Annotationen aus der Entfernungphase werden Kantenbedingungen konstruiert, die durch ein WfMS automatisch ausgewertet werden können. Zur Parameterübergabe und -rückgabe an Unterprozesse werden in dieser Phase zusätzlich Systemaufgabenknoten erzeugt, die den Eintritts- bzw. Austrittspunkt eines Workflows bilden. Abbildung 5 illustriert die Umwandlung symbolischer Kantenannotationen in automatisch auswertbare Kantenbedingungen mit Hilfe von Prozessvariablen.

4. Das Werkzeug *ConvEX*

ConvEX (Converter from EPC to XPDL) ist ein Softwarewerkzeug, das den in Kapitel 3 beschriebenen Übersetzungsvorgang integriert und gebündelt mit einer graphischen Benutzeroberfläche (GUI) zur Verfügung stellt. Die Eingabe der EPKs in *ConvEX* erfolgt im EPML-Format, das sich mit Hilfe eines Transformationsskriptes[MN04] aus Prozessmodellen des ARIS-Toolsets erzeugen lässt. Während des Übersetzungsvorgangs werden Modellierungsinconsistenzen, z.B. fehlende Funktionsdefinitionen, erkannt und dem Nutzer über die GUI mitgeteilt. Weiterhin erlaubt die GUI das Verändern von Annotationen an Knoten und Kanten des Workflowgraphen sowie die Visualisierung des bearbeiteten Prozesses. Der Export der erstellten Workflowbeschreibungen kann wahlweise in das lokale Dateisystem oder in ein WebDAV-basiertes Dokumentenmanagement-System, wie es bspw. in [KRS06] beschrieben wird, erfolgen.

ConvEX kann kostenfrei uber die Homepage der Professur Praktische Informatik der TU Chemnitz² heruntergeladen werden.

5. Zusammenfassung

Aufgrund des stetig zunehmenden Kostendrucks geht ein Trend in Richtung der rechnergestutzten Abarbeitung von betrieblichen Geschaftsprozessen und behordlichen Verwaltungsvorgangen in Form von Workflows. In diesem Artikel wurde das ubersetzungswerkzeug *ConvEX* vorgestellt, das Geschaftsprozessmodelle auf Basis ereignisgesteuerter Prozessketten in Workflowbeschreibungen im XPDL-Format transformieren kann. Dadurch kann eine komplette Neumodellierung von Verwaltungsvorgangen, die bereits als Geschaftsprozessmodell vorliegen, vermieden werden.

Literaturverzeichnis

- [Coa02] Workflow Management Coalition. Spezifikation von XPDL Version 1.0: http://www.wfmc.org/standards/docs/TC-1025_10_xpdl_102502.pdf, October 2002.
- [Enh05] Enhydra Shark 1.1 Documentation <http://shark.objectweb.org/doc/1.1/>, 2005.
- [GKS92] G. Keller, M. Nuttgens, and A.-W. Scheer. Semantische Prozessmodellierung auf der Grundlage „Ereignisgesteuerter Prozessketten (EPK)“. *Veroffentlichungen des Instituts fur Wirtschaftsinformatik (IWi)*, Heft 89, 1992.
- [KRS06] R. Kunis, G. Runger, and M. Schwind. Dokumentenmanagement fur Verwaltungsvorgange im E-Government. *Chemnitzer Informatik Berichte, CSR-06-10*, 2006.
- [MN04] J. Mendling and M. Nuttgens. Transformation of ARIS Markup Language to EPML. In *Proc. of the 3rd GI Workshop on Event-Driven Process Chains (EPK 2004)*, 2004.
- [Sch98] A.-W. Scheer. *ARIS – Modellierungsmethoden, Metamodelle, Anwendungen*. Springer-Verlag, 1998.
- [Sta01] J.L. Staud. *Geschaftsprozessanalyse – ereignisgesteuerte Prozessketten und objektorientierte Geschaftsprozessmodellierung fur betriebswirtschaftliche Standardsoftware*. Springer-Verlag, 2001.

² Die Homepage der Professur Praktische Informatik ist unter <http://www.tu-chemnitz.de/informatik/PI/> zu erreichen.