

Wissensbasierte Modellierung von Payment-Prozessen

Andreas Fasel

*Technische Universität Ilmenau, Fakultät für Mathematik, Institut für Medien- und Kommunikationswissenschaften, Fachgebiet Multimediale Anwendungen
Am Eichicht 1, D-98684 Ilmenau
andreas.fasel@tu-ilmenau.de*

Schlagworte: Payment, Ontologie, Wissensmanagement, E-Commerce, Inferenz

Abstract: Eine wissensbasierte Modellierung von Prozessen digitaler Paymentssysteme, ua für Prozesse mobiler Paymentssysteme, dient einer transparenten und intelligenten Präsentation von Wissen gegenüber den Anwendern solcher Systeme (Käufer, Verkäufer, Banken, Integratoren). Zum Beispiel ist bei der Integration in übergeordnete Systeme und Geschäftsprozesse (Warenwirtschaft, Buchhaltung, Auftragsverfolgung etc) ein intelligentes Modell für schnellen Zugriff bzw Wissenszugriff auf Prozesse oder Datenflüsse sehr hilfreich. An dieser Stelle soll auf einen eigenen Entwurf einer Prozessmodell-Ontologie zurückgegriffen werden, welcher über die Beschreibungskraft von Freitexten, Daten- oder Informationspools hinausgeht. Im Wesentlichen lässt sich eine formale Semantik definieren und durch die zusätzliche Nutzung des Inferenz-Mechanismus kann logisch geschlussfolgertes Wissen innerhalb der Prozess-Ontologie durch manuelle oder automatisierte Anfragen gewonnen werden, zum Beispiel für die Situationsanalyse von Nutzern. Für die Modellierung der Prozess-Ontologie wird die Sprache DAML+OIL/OWL aus dem Semantic Web verwendet.

1. Einleitung

Der Elektronische Handel im Internet hat in den letzten Jahren einen rasanten Aufschwung erlebt. Die Online-Einkaufsmöglichkeit wird von der wachsenden Zahl der Internet-User vermehrt genutzt, wobei die Prozessketten-Effektivierung ein anzustrebendes Ziel darstellt. Mittlerweile ist auch die Transparenz der Prozessketten für alle Marktbeteiligten zur Bildung einer Vertrauensbasis eine wichtige Voraussetzung für den E-Commerce und M-Commerce geworden. Bei der Zahlungsabwicklung müssen die komplexen Prozessmodelle von elektronischen Zahlungssystemen wie moxmo, GeldHandy, ehemals paybox, Paypal, Moneybookers, Firstgate Click&Buy, T-Pay, Paysafe-card, Geldkarte, 3-D Secure ("Verified by VISA"), "MasterCard

SecureCode“, Online-Überweisung und viele mehr auch gut verstanden werden, um einen vertrauenswürdigeren Umgang mit solchen Systemen zu fördern.

Ein verbesserter Know-How-Transfer über elektronische Zahlungssysteme auf der Basis systematisch aufgearbeiteter Fakten ist nötig, wenn alle Beteiligten bei einer Transaktion bewusst agieren sollen und begründen zum Teil die folgende Arbeitsweise.

2. Ausgangspunkt für die Analyse von Zahlungssystemen

Der Ausgangspunkt für die Analyse der Zahlungssysteme ist das funktionale Testen aus Sicht des Käufers und Verkäufers innerhalb einer studentischen Praxiswerkstatt. Konkret heißt dies ein Auftreten als Web-Verkäufer und somit das "Hinein-Programmieren" in die unterschiedlichen Schnittstellen der Zahlungssysteme: Eine Untersuchung von Zahlungssystemen aus Käufersicht bedeutet zunächst die Erfüllung der Nutzungsvoraussetzungen (technische und bankliche) und danach das Austesten von möglichen Anmeldungen, Ladevorgängen, Transaktionen (Kauf), Hilfen, Supports, Prüfungen und sonstigen Funktionen (siehe [GF2003]) und hierarchische Kriterienstruktur in Abbildung 1). Gering strukturierte Formalisierung der statischen Begriffe aus dem Kriteriensystem.

Die geringe Berücksichtigung des Prozessmodells (zeitliche Zusammenhänge).

Die Erweiterung und Änderung von Wissen im Begriffssystem (Kriteriensystem siehe Abb 1) ist durch die unformellen Texte teilweise schwierig, zeitaufwendig und lässt unexakte Beschreibungsmöglichkeiten zu. Besser geeignet sind Ontologien.

Bei der Auswahl von geeigneten Ontologie-Sprachen lassen sich die Sprachen des Semantic Web (siehe [W3C2001b], [BHL2001]), dem Nachfolger des WWW, als geeignet herausstellen. Neben der syntaktischen Standardisierung von Dokumenten des WWW (HTML, XML) wird darüber hinaus auch die semantische Beschreibung von Wissensbeständen ermöglicht, zum Beispiel für semantische Suchmaschinen und Agenten für Verträge, Kommunikation, Information, Vergleich, Wissenssammlung usw.

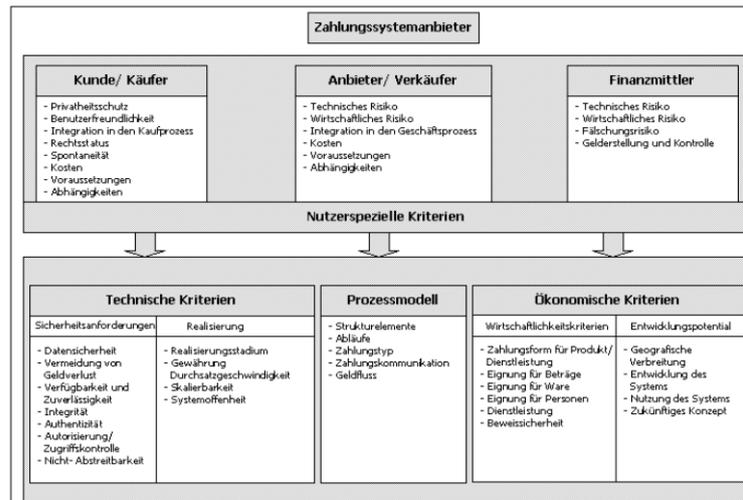


Abb 1: Kriterienstruktur als hierarchisches System

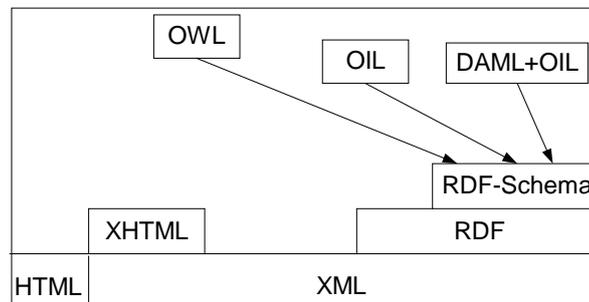


Abb 2: Schichtenmodell des Semantic Web

OWL: Ontology Web Language

DAML+OIL: DARPA Agent Markup Language

OIL: Ontology Inference Layer

RDFS: Resource Description Framework Schema

RDF: Resource Description Framework

XML: Extended Markup Language

XHTML: Extensible HyperText Markup Language (siehe Beschreibungen für die Sprachen unter [DAML2002a],[DAML2002b], [W3C2001a], [W3C2001b], [W3C2004a], [W3C2004b])

Folgende positive Eigenschaften der Semantic Web-Sprachen aus Abb 2 sollen noch einmal aufgeführt werden:

1. Die modellierten Ontologien sind maschinenlesbar.
2. Interoperabilität: Die Kommunikation wird zwischen verschiedenen Parteien wegen des „Semantic Web“ als Web-Standard leichter ermöglicht.
3. Ontologiebasierte Wissensrepräsentation ist sehr gut möglich.
4. Erweiterbarkeit der Modelle ist machbar.
5. Inferenz (logisches Schließen) wird ermöglicht, was bei unformalen Texten (nicht automatisch) nur durch langes Einlesen und Verstehen mit Zeitverlust und meist nicht vollständig möglich ist.

3. Eigener wissenschaftlicher Ansatz für eine Prozessontologie

Nachdem kurz auf semantische Modellierungsmöglichkeiten und die Bedeutung der ontologiebasierten Formalisierung von Wissen eingegangen wurde, werden nun Funktionen und Prozesse bei Zahlungssystemen analysiert. Je nach Entwurf eines Paymentsystems gibt es unterschiedliche Funktionalitäten festzuhalten. Hinter jeder Funktion liegen entsprechende Prozesse zur Realisierung.

Über die Summe von einschlägigen und marktpräsenten Zahlungssystemen lassen sich verschiedenste Teilfunktionen bzw Prozesse erkennen, die je nach Wahl des Zahlungssystems unterschiedlich (in Rücksicht auf limitierte Seitenzahlen werden keine weiteren Details zu Paymentsystemen dargelegt) sind:

Anmeldung als Käufer/Verkäufer
Aufladen von Konten
Entladen von Konten
Transaktion (Kauf von Produkt/Dienstleistung)
Verwaltung (für vergangene und gegenwärtige Daten: Stammdaten im System, alte & aktuelle Bewegungsdaten)
Zusatzfunktionen über den reinen Kauf hinweg
Abmeldung vom System
Hilfe / Support

Um die verschiedensten Prozesse, wie sie auch immer bei Paymentssystemen gelagert sein können, nach einem immer gleichen Wissensschema zu erfassen, wird ein Prozess-Ontologie-Schema vorgeschlagen (siehe Abb 3). Einige wichtige Teile dieses Wissensschemas werden hier näher vorgestellt (keine Komplettbeschreibung wegen hoher Komplexität):

Jedes Paymentssystem hat ein „Prozessmodell“, welches prinzipiell für die Gesamtheit aller Prozesse und Subprozesse verantwortlich ist und diese quasi durch die Relation „hat_Prozess“ zur Klasse „Prozess“ zusammenfasst. Im Schema wird dem „Prozess“ eine „Prozess-Version“ zugeordnet. Ein „Prozess“ kann n Versionen von „Prozess-Version“ besitzen (siehe Relation „hat_Version“). Dies geschieht für den Fall, dass Prozessversionen von Beginn an existieren oder Prozessänderungen vom Paymentssystemanbieter im Laufe der Nutzungszeit am Markt eingebaut werden. Zum Beispiel sollen statt einer Prüfung einer einzigen Zahlungsbedingung jetzt zwei oder drei Bedingungen mit UND-Verknüpfung in den Prozess eingebaut werden (wegen Sicherheit). Dies stellt den „Prozess“ nicht auf den Kopf, bedeutet aber eine kleinere Abänderung. Änderungen wirken sich somit nicht auf den Prozess aus, sondern lediglich auf die Versionen des Prozesses. Andernfalls müsste man den „Prozess“ für kleine Änderungen immer wieder neu anlegen, obwohl der primäre Inhalt von „Prozess“ immer gleich bliebe.

Ein Prozess bzw eine Prozess-Version besitzen natürlich entsprechende Eingangsdaten, Steuerparameter, Ausgangsdaten, Eingangsbedingungen und Ausgangseffekte. Diese sind hier nicht weiter dargestellt (wegen der Größe der Grafik).

Die „Prozess-Versions-Relation“ dient dem Speichern der Änderungen zwischen den Versionen eines Prozesses untereinander (Differenzbeschreibung / „Änderung“).

Die „Prozess-Version“ besitzt noch eine „Substitutions-Relation“, bei der jeweilige Alternativen als Austauschvariante für eine vorhandene „Prozess-Version“ verwendet werden könnten. Hiermit sind Änderungen im Prozess gemeint, welche von größerem Ausmaß als die kleineren Versionsänderungen sind (regelrechte Prozess-Neugestaltungen).

Die „Prozess-Version“ besitzt entsprechende Werte für die Dauer, die Anfangszeit, die Endzeit und den kritischen Pfad (Pfad mit geringsten Zeitübergängen von einem Sub-Prozess zum nächsten über das gesamte Prozessnetz hinweg).

Es gibt zwei Unterklassen von „Prozess- Version“:

- „Einfacher-Prozess“ dient der Darstellung eines Vorganges, welcher in einem Schritt durchgeführt werden soll oder dessen Zerlegung nicht weiter möglich oder sinnvoll ist. Quasi als Black-box ist dann dieser Prozess zu betrachten.
- „Zusammengesetzter-Prozess“ kann aus einfachen und nicht einfachen Prozessen bestehen.

Die Umsetzung dieser Aussagen in DAML+OIL/OWL sieht für einen Teilausschnitt so aus (Prozess-Version kann nur einer von zwei Typen sein):

```
<daml:Class rdf:ID="Prozess-Version">
  <rdfs:comment> Die Prozess-Versionsklasse
</rdfs:comment>
  <daml:disjointOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#Einfacher-Prozess"/>
    <daml:Class rdf:about="#Zusammengesetzter-Prozess"/>
  </daml:disjointOf>
</daml:Class>
```

Der einfache Prozess ist Unterklasse von Prozess-Version. Zusammengesetzter Prozess ist Unterklasse von Prozess-Version. „Steuerkonstrukt“ besitzt eine Reihe von Unterklassen, welche die gesteuerte Abarbeitung von Prozessen entsprechend vornehmen können. Dazu zählen:

- *Reihenfolge: Eine Prozessliste von nacheinander folgenden Prozessen (Sequenz) wird in fester Reihenfolge (Netzplan) abgearbeitet.*
- *Ungeordnet: Eine Prozess-Liste, deren Prozesse in beliebiger Reihenfolge ausgeführt werden können, wird abgearbeitet (random-Liste von Prozessen wird durchlaufen).*
- *Parallel: Eine Prozessliste mit Prozessen, die gleichzeitig (parallel) auszuführen sind, wird abgearbeitet. Die Prozesse werden aber nicht miteinander abgestimmt.*
- *Parallel mit synchron: Eine Prozessliste mit Prozessen, die gleichzeitig (parallel) auszuführen sind, wird abgearbeitet und deren Synchronisation (gemeinsame Wartepunkte und danach planmäßiges weiterarbeiten) wird vorgenommen.*

Weitere Konstrukte wurden jetzt an dieser Stelle eingespart.

Verwendung von logischen Zusammenhängen zwischen den Begriffen der Ontologie. Somit besteht der Zugriff nicht nur auf explizites Wissen, welches direkt vom Nutzer eingegeben wurde, sondern auch auf das abgeleitete, geschlussfolgertes Wissen.

Beispiel einer Ontologie (siehe dazu Abb 4):

- Relationen aufbauen:
 hat(Paymentsystem, Kommunikations-Typ)
 SubClass(Online- Kommunikations-Typ, Kommunikations-Typ)
 SubClass(Offline- Kommunikations-Typ, Kommunikations-Typ)
 prüfen_von(Online- Kommunikations-Typ, Prüfungsmöglichkeit_bei_Kauf)
- Eine Regel aufbauen:
 Wenn „Paymentsystem“ einen „Kommunikations-Typ“ hat und der Typ ein „Online- Kommunikations-Typ“ ist, dann besteht die „Prüfungsmöglichkeit“ von „Kontostand“
- Anfrage an System stellen:
 ?= Welches „Paymentsystem“ hat die „Prüfungsmöglichkeit_bei_Kauf“ des Kontostandes?
- Antwort vom System erhalten:
 „Paymentsystem P1“

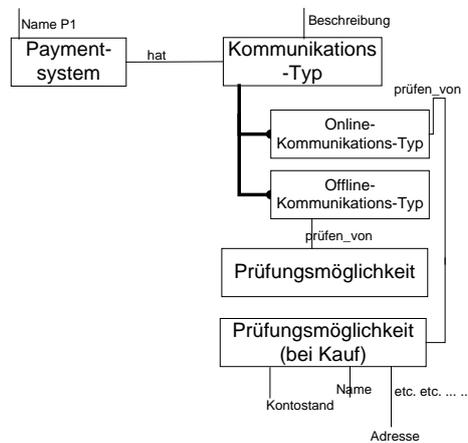


Abb 4: Beispiel-Ontologie (zum Thema Kommunikationstyp)

Das Inferenzsystem kann demnach den Aussagen und Regeln der Ontologie einfach folgen. Solche logische Schlussfolgerungen können

auch automatisch gezogen werden, aufgrund einer jeweiligen Situationsanalyse.

Zusammenfassung

Die Verwendung von Wissensmanagement-Methoden (Ontologie-Modelle) für Prozesse elektronischer Zahlungssysteme bzw auch für Geschäftsprozesse verbessern die Beschreibungs- und Anwendungsmöglichkeiten solcher Systeme.

4. Literatur

[BHL2001] Berners-Lee T., Hendler J., Lassila O.: The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. 05/2001. <http://www.scientificamerican.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2> (05/2004)

[DAML2002a] The DARPA Agent Markup Language Homepage: DAML Language. <http://www.w3.org/TR/daml+oil-reference>. (05/2004)

[DAML2002b] The DARPA Agent Markup Language Homepage. <http://www.w3.org/TR/> (05/2004).

[ECIN] ECIN – Electronic Commerce InfoNet: Home page – Herausgeber von ECIN ist das FTK (Forschungsinstitut für Telekommunikation. Dortmund, <http://www.ecin.de/> (04/2004).

[ECIN2001] ECIN – Electronic Commerce InfoNet: ePayment - Alte Besen kehren noch am besten. 02/2001, <http://www.ecin.de/zahlungssysteme/epayment> (04/2004).

[GF2003] Grimm R., Fasel A.: Handbuch / Praxisbericht Elektronische Zahlungssysteme – Studentische Arbeiten im Rahmen der Praxiswerkstatt Zahlungssysteme. Version 4.0.2 10/2003. Ilmenau, bei Nachfrage/Registrierung unter http://www.stud.tu-ilmue.de/~fasel/Handbuch_downloadframe.htm (04/2004).

[Gru1993] Gruber T.R.: A Translation Approach to Portable Ontology Specifications. In: Knowledge Systems Laboratory Technical Report KSL 92-71. Computer Science Department, Stanford University in Stanford: 1993.

[Gru1995] Gruber T.R.: Towards principles for the design of ontologies used for knowledge sharing In: International Journal of Human-Computer Studies. Vol 43, No 5/6, 1995

[KS2002] Ketterer K.-H.(Hrsg.), Stroborn K.: Handbuch ePayment – Zahlungsverkehr im Internet, Systeme, Trends, Perspektiven. Köln: Fachverlag Deutscher Wirtschaftsdienst GmbH&Co.KG, 2002.

[SEMOSSST] Studer R., Erdmann M., Mädche A., Schnurr H-P., Staab S., Sure Y., Tempich Ch.: Arbeitsgerechte Bereitstellung von Wissen – Ontologien

für das Wissensmanagement. http://www.ontoprise.de/documents/Ontologien_fuer_das_WM_Arbeitsgerechte_Bereitstellung_von_Wissen.pdf (05/2004)

[UG1996] Uschold,M.; Grüninger,M.: Ontologies: principles, methods and applications. In: Knowledge Engineering Review 11. 1996, Nr.2, S.93-155.

[W3CX] World Wide Web Consortium: „Extensible Markup Language (XML). <http://www.w3.org/XML> (04/2004).

[W3C2001a] World Wide Web Consortium: „DAML+OIL (March2001) Reference Description – W3C Note 18 December 2001”.

<http://www.w3.org/TR/daml+oil-reference> (05/2004).

[W3C2001b] World Wide Web Consortium: „Semantic Web”.

<http://www.w3.org/2001/sw/> (04/2004).

[W3C2004a] World Wide Web Consortium: „OWL Web Ontology Language Reference – W3C Recommendation 10 February 2004”.

<http://www.w3.org/TR/2004/REC-owl-ref-20040210/> (05/2004).

[W3C2004b] World Wide Web Consortium: OWL Web Ontology Language Guide – W3C Recommendation 10 February 2004.

<http://www.w3.org/TR/2004/REC-owl-guide-20040210/> (05/2004).

[Web1999] Weber R.: Chablis – Market Analysis Digital Payment Systems Technical Report (TUM 19819). Version 1.5, 08/1999, Das Dokument (market-pay.pdf) ist per Anfrage an folgende Mail-Adresse zu bekommen <ricarda.weber@mchp.siemens.de>.

[Zob2002] Zobel A.: „Kriterien zur Entwicklung und Test von Strukturkriterien für elektronische Zahlungssysteme auf Basis der Praxiswerkstatt „Elektronische Zahlungssysteme“ Technische Universität Ilmenau, Diplomarbeit, 09/2002.