

IT-Governance bei Wiederverwendung von Services

Gabriela Loosli, David Heim, Gerhard F. Knolmayer

Institut für Wirtschaftsinformatik, Universität Bern
Engehaldenstrasse 8, CH-3012 Bern
{gabriela.loosli, gerhard.knolmayer}@iwi.unibe.ch
davidheim@bluewin.ch

Abstract: In Service-orientierten Architekturen ist eine automatisierte Auswahl geeigneter Services von Vorteil, bei einer Bindung zur Laufzeit sogar eine Notwendigkeit. Werden Services in einem anderen Kontext wieder verwendet, ist zu berücksichtigen, dass dabei unterschiedliche Regulatorien relevant sein können. In diesem Beitrag wird ein Lösungsansatz für die Auswahl konformer Services auf Basis semantischer Konzepte dargestellt. Anhand eines Beispiels wird gezeigt, wie der konzeptionelle Entwurf mit unterschiedlichen semantischen Beschreibungssprachen implementiert werden kann und welche Unterschiede dabei bestehen. Ausgewählte Sprachen und ihre spezifischen Eigenschaften werden anhand dieses Anwendungsfalls verglichen und die Umsetzung mit zwei Sprachen skizziert.

1 Einleitung

Unternehmen müssen immer mehr Vorschriften erfüllen; nach [Li07] seien allein in den USA seit 1981 mehr als 118.000 Regulatorien festgelegt worden. Insbesondere international tätige Unternehmen sind mit rechtlichen Rahmenbedingungen verschiedener Gesetzgeber und Regulatoren konfrontiert, die Unterschiede und Widersprüchlichkeiten aufweisen können [Kn07, S99]. Neben Vorschriften zur Rechnungslegung existieren viele weitere gesetzliche Vorgaben, branchenspezifische Vorschriften, Standards oder interne Regeln, zu welchen die Unternehmen "compliant" sein sollen.

Da immer mehr Geschäftsprozesse durch IT-Systeme ausgeführt oder unterstützt werden, sollten IT-Systeme die Einhaltung dieser Regulatorien sicherstellen oder zumindest unterstützen. Diese Anforderung ist eine wesentliche Komponente der IT-Governance, die sich aus der Corporate Governance ableitet [KL06, 451; Kn07, S98; SS07, 69].

Als Basis der Gestaltung von IT-Systemen werden heute oft Service-orientierte Architekturen (SOA) empfohlen [PG03; Er04; KBS04; Do05; Jo08]. Vorteile einer SOA werden u. a. in der Wiederverwendbarkeit von Services und in der Flexibilität, Änderungen der Geschäftsprozesse zeitnah in IT-Systeme umsetzen zu können, gesehen. Fundamentales Element einer SOA sind Services. Diese sind Software-Einheiten verschiedener Granularität, welche zu komplexeren Services bis hin zu Applikationen aggregiert und in unterschiedlichen Anwendungsfällen wieder verwendet werden können. Im SOA-

Konzept wird zwischen statischer und dynamischer Bindung unterschieden; letztere entspricht einer Bindung zur Laufzeit. Bei der Auswahl von Services sind verschiedene Kriterien zu beachten; unter anderem können anwendungsspezifische regulatorische Vorschriften dazu führen, dass in anderem Kontext geeignete Services nicht verwendet werden dürfen. In diesem Sinn leistet eine SOA-Governance [BI07; JG07; SS07; KSH08; Ni08; No09] einen Beitrag zur IT-Governance.

Der Beitrag ist wie folgt gegliedert: Im Abschnitt 2 wird ein Fall betrachtet, in dem ein Service für unterschiedlichen Regulatorien unterliegenden Prozessen auszuwählen ist. Das ebenfalls in Abschnitt 2 dargestellte Lösungskonzept zeigt die Bedeutung der Wissensrepräsentation, auf die in Abschnitt 3 eingegangen wird. Abschnitt 4 skizziert zur Umsetzung des Lösungskonzepts verfügbare Semantic-Web-Services-Beschreibungssprachen. Abschnitt 5 diskutiert die Umsetzung des Lösungskonzepts in zwei als geeignet bewerteten Sprachen. In Abschnitt 6 werden verwandte Lösungsansätze betrachtet. Abschnitt 7 fasst die Ergebnisse zusammen und gibt einen Ausblick.

2 Ein Anwendungsfall der Service-Auswahl: Konzeptionelle Lösung

2.1 Problemstellung

Bei dynamischer Bindung sind automatisierte Lösungen zwingend notwendig, wenn Services anhand von Bedingungen oder Eigenschaften ausgewählt werden sollen [Lo08, 14]. Wir beschreiben im Folgenden eine Anwendung, in der ein Service in einem zuvor nicht berücksichtigten Kontext wieder verwendet wird.

Ein Service, der Log-Daten speichert, wird auf Einhaltung der Regulatorien geprüft, bevor er in Produktion geht. Später wird der Service in einem Bestellprozess wieder verwendet, in dem er neben den allgemeinen Bestellinformationen auch Kreditkartendaten speichert. Nun unterliegt dieser Service zusätzlich den Regeln des Payment Card Industry Data Security Standard (PCI DSS), ohne dass er selbst verändert und neu in Produktion gebracht wurde. Nur die Art der Verwendung, also der Kontext, hat sich verändert. Die PCI-DSS-Regeln untersagen beispielsweise das Speichern bzw. Aufbewahren des PIN oder der vollständigen Magnetstreifen-Information [Fo06; Pc08].

Für einen bestehenden Prozess 1, der Kreditkartendaten benötigt, sind zwei Log-Services verfügbar, ein billigerer und langsamerer sowie ein teurerer und schnellerer. Beide erfüllen die in diesem Zusammenhang relevante PCI-DSS-Regulation. Der Prozess sucht den billigsten Service und wählt den Log-Service 1 (Abbildung 1; vgl. [Lo08]).

Später wird ein Prozess 2 erstellt, der einen eigenen Log-Service besitzt. Da dieser gute nicht-funktionale Eigenschaften besitzt (noch billiger als Log-Service 1 und trotzdem schnell, da er die Aussortierung kritischer Daten unterlässt), wird er in das Repository als Log-Service 3 aufgenommen. Allerdings erfüllt er die PCI-DSS-Regulation nicht, was für den Prozess 2 kein Problem darstellt, da er keine Kreditkartendaten benötigt.

Bei der nächsten Ausführung des Prozesses 1 greift dieser automatisch auf den nun billigsten, jedoch für ihn nicht konformen Log-Service 3 zu: Ohne Berücksichtigung der regulatorischen Eigenschaften wird durch Änderung des Service-Bestands im Repository zur Laufzeit ein nicht konformer Service gewählt. Somit genügt es nicht, neu erstellte oder geänderte Prozesse vor ihrer Ausführung zu prüfen; auch unveränderte Prozesse können von Änderungen im Service-Repository betroffen sein.

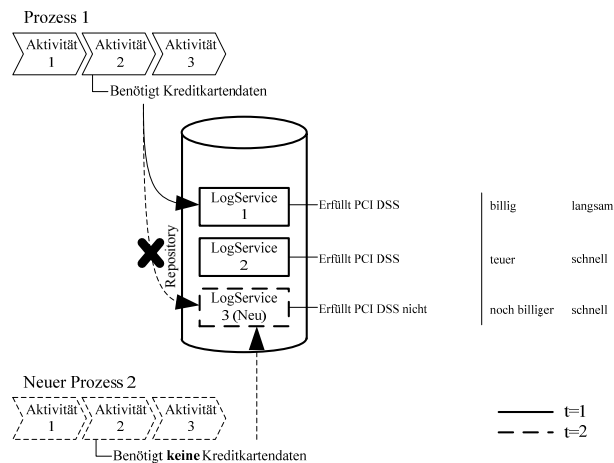


Abbildung 1: Wiederverwendung der Services in einem anderen Kontext

2.2 Konzeptionelle Lösung

Wie das Beispiel zeigt, hängen die zu erfüllenden Regularien vom Kontext der Servicenutzung ab. Dieser muss also über Inputparameter berücksichtigt werden, um die Regularien zu bestimmen, welche durch den Service erfüllt werden müssen (Ex-ante-Prüfung). Dazu bedienen wir uns semantischer Konzepte.

Um eine Beziehung zu den zu erfüllenden Regularien herzustellen, eignet sich eine Ontologie, die logische Schlussfolgerungen ermöglicht. Damit ersichtlich ist, welcher Service welche Regularien erfüllt, ist die Beschreibung der angebotenen Services entsprechend zu erweitern. Für extern bezogene Services könnte die Angabe des Service-Anbieters, zu welchen Regularien ihr Service compliant ist, durch eine Prüfungs- oder Zertifizierungsgesellschaft bestätigt werden. Dies ist wichtig, wenn der Service-Nutzer nachweisen muss, dass die Compliance-Anforderungen erfüllt werden [Do05, 261; Kn07, S101].

Dieser Lösungsansatz behandelt die teils umfangreichen und komplizierten Prüfungen zur Einhaltung der Regularien als separate Problemstellung und reduziert so die Komplexität der Prüfungen. Die Einhaltung kann von unterschiedlichen Systemen oder auch manuell geprüft werden, was insbesondere in heterogen verteilten Umgebungen hilfreich ist. Das Lösungskonzept zeigt die Bedeutung der Wissensrepräsentation, auf die im folgenden Abschnitt eingegangen wird.

3 Wissensrepräsentation

3.1 Semantic Web Services

Web Services, die zurzeit am häufigsten betrachtete Implementierungsform von Services in einer SOA, werden meist in der Sprache WSDL beschrieben. Eine WSDL-Datei enthält nur syntaktische Informationen (*wie* wird der Service aufgerufen), beispielsweise den Namen und Datentyp eines Parameters, nicht jedoch seine Bedeutung (Semantik; *was* leistet der Service und *was* bedeutet der Parameter). Diese wird unstrukturiert in einem Textfeld oder in externen Dokumenten beschrieben.

Mit der Anfügung von semantischen Metadaten, also weiteren Beschreibungselementen, sowie den Möglichkeiten von Referenzen innerhalb der Metadaten wird die Text-Beschreibung strukturiert. Web Services, welche in dieser Art semantisch beschrieben werden, werden als Semantic Web Services bezeichnet [Do05, 283; PLL06, 509; Be07, 69ff]. Für eine automatisierte Verarbeitung ist eine einheitliche Begrifflichkeit für die Metadaten notwendig. Für Semantic Web Services gibt es standardisierte Beschreibungssprachen, auf die im Abschnitt 4.1 näher eingegangen wird.

3.2 Ontologien

Auch wenn die Services detailliert und standardisiert beschrieben sind, kann es Missverständnisse geben bezüglich des Inhalts der Metadaten. Insbesondere besteht dieses Problem dann, wenn Services aus unterschiedlichen Quellen bezogen werden [MSS01, 394]. Zur Lösung dieses Problems werden Ontologien vorgeschlagen. Eine Ontologie ist eine formale Beschreibung von Begriffen (Konzepten) und ihren Beziehungen (Relationen), welche innerhalb einer Domäne für eine Gruppe von Personen Gültigkeit hat [Gr93; MSS01; He02]. Sie kann auch Aussagen enthalten, die logische Schlussfolgerungen ermöglichen [HF08, 13]. Semantic-Web-Services-Beschreibungssprachen können Referenzen zu Ontologien herstellen.

Ein Unterscheidungsmerkmal von Ontologien ist ihre Expressivität [He08b, 8ff]. Je höher die Expressivität einer Beschreibungssprache, desto komplexere Zusammenhänge können in der Ontologie ausgedrückt werden. Dies ermöglicht weitergehende logische Schlussfolgerungen. Mit der Expressivität steigen aber auch die Komplexität und damit der Rechenaufwand zur Verwendung der Ontologie [GHA07, 60f].

3.3 Abgleich von Anfrage und Angebot (Matching)

Zur Strukturierung der Abläufe bei Verwendung eines Dienstes werden unterschiedliche Phasen betrachtet [PLL06, 6; Fe07, 115; Gr07, 214; Pr07, 168ff]. In unserem Zusammenhang ist vor allem die Discovery-Phase relevant, in der abstrakte Servicebeschreibungen abgeglichen werden. Die Abstraktion führt z. B. zur Eliminierung oder Klassifikation von Daten: So können Namen in statistischen Auswertungen ignoriert und Zahlenangaben in Klassen zusammengefasst werden.

Ein Abgleich von Anfrage und Angebot ist komplex. Bereits bei Wahl des Inferenztyps ist die Modellierung einzubeziehen: Wird z. B. Intersection (der Dienst kann nicht alles liefern, bietet aber auch nicht Relevantes an) gewählt und sind in den Beschreibungen nicht alle relevanten Disjunkt-Beziehungen eingefügt, können unerwünschte Resultate entstehen [Gr07, 229ff], die für den Anfrager oft nicht nachvollziehbar sind [K106, 90].

4 Darstellung und Vergleich ausgewählter Semantic-Web-Services-Beschreibungssprachen

Während die Web Services Description Language (WSDL) ein Quasi-Standard für die syntaktische Beschreibung von Web Services ist, bestehen für die semantische Beschreibung mehrere voneinander unabhängige Ansätze. Im Folgenden werden die wichtigsten aufgeführt und verglichen [He08a].

4.1 Darstellung von Semantic-Web-Services-Beschreibungssprachen

4.1.1 Web Service Modeling Ontology (WSMO)

WSMO ist ein konzeptuelles Modell zur Beschreibung verschiedener Aspekte von semantischen Web Services [Ci08, 29]. Ein Schwerpunkt liegt auf der Integration von Web Services in einem heterogenen Umfeld [La07, 180f]. Zur Beschreibung wurde eine Ontologiesprache, die Web Service Modeling Language (WSML), konzipiert. Die WSMO enthält vier Hauptelemente:

Ontologie: Die Beschreibung einer Ontologie in WSMO sieht fünf Elemente vor: Konzepte (Klassen), Relationen, Funktionen (Unterklasse der Relationen), Instanzen (von Konzepten; innerhalb der Ontologie oder Referenz zu externen Datenspeichern) und Axiome (logische Ausdrücke). Zusätzlich enthält eine Ontologie nichtfunktionale Elemente, kann Mediatoren verwenden und andere Ontologien importieren [RLK06].

Web Service: Beschreibt die Dienste, welche von Anbietern bereitgestellt werden, in Form von Funktionalitäten (Fähigkeiten) der Services: Vorbedingungen (im Wesentlichen die Eingabeparameter), Annahmen (Voraussetzungen für einen erfolgreichen Aufruf, werden nicht überprüft), Nachbedingungen (Rückgabewerte) und Effekte (Auswirkungen eines erfolgreichen Aufrufs, z. B. geänderter Saldo eines Bankkontos). Zudem können Schnittstellen ("hasChoreographie" und "hasOrchestration") und nichtfunktionale Eigenschaften definiert werden. Für die Servicebeschreibung steht das Vokabular zur Verfügung, das von der Ontologie definiert wird.

Ziel: Modelliert die Nachfragersicht und beschreibt die Anforderungen an die Funktionalität eines Web Service, ebenfalls unter Verwendung des Vokabulars der Ontologie. Die Struktur der Beschreibung entspricht der des Web Services. Anfrage und Angebot können auf Basis von Fähigkeiten, Vorbedingungen, Effekten sowie nichtfunktionaler Aspekte abgeglichen werden [Ci08, 49f].

Mediator: Stellt die Kompatibilität von Komponenten in einem heterogenen Umfeld sicher. Mediationen können für Daten (heterogene Datenquellen, Integration verschiedener Ontologien), Kommunikationsprotokolle und Geschäftsprozesse erfolgen.

4.1.2 Semantic Markup for Web Services (OWL-S)

OWL-S basiert auf der Ontologiesprache OWL und erweitert diese um spezifische Aspekte zur Beschreibung von Web Services. OWL-S enthält ebenfalls vier Hauptelemente [Ma04]:

Serviceprofil: Beschreibt die angebotene Funktionalität eines Web Service aus Sicht des Anbieters oder die gewünschte Funktionalität aus Sicht eines Nachfragers [Ma04]. Obwohl OWL-S nicht explizit zwischen Anfrage und Angebot unterscheidet, können Abgleiche von Serviceprofilen basierend auf Ein- und Ausgabeparametern stattfinden [Gr07, 223]. Serviceprofile können hierarchisch angeordnet sein. Ihre funktionale Beschreibung beinhaltet Eingaben, Vorbedingungen, Ausgaben und Effekte. Ein weiteres Beschreibungselement bilden die Charakteristika, zu denen Klassifikationen und Beurteilungen von nichtfunktionalen Aspekten (z. B. der Zuverlässigkeit) zählen.

Servicemodell: Sobald sich ein Nachfrager für einen Web Service entschieden hat, wird statt des Serviceprofils das Servicemodell verwendet. In ihm wird detailliert festgelegt, welche Operationen zur Verfügung stehen und welche Voraussetzungen für einen erfolgreichen Aufruf erfüllt sein müssen. Spezifiziert werden auch die Bedingungen, welche zu unterschiedlichen Ausgaben führen. Die Struktur der Beschreibung der Funktionalität entspricht der des Serviceprofils. Zusätzlich enthält das Servicemodell ein Prozessmodell.

Service Grounding: Beschreibt aus technischer Sicht, wie der Service genutzt werden kann und beinhaltet Angaben über z. B. das Transportprotokoll oder das Nachrichtenformat. Das Grounding-Element wurde in WSDL implementiert [SD05].

Service: Kombiniert Serviceprofil, -modell und -Grounding zu einer Einheit, die angeboten und aufgerufen werden kann. Ein Service kann von mehreren Serviceprofilen, jedoch von höchstens einem Servicemodell beschrieben werden. Ein Servicemodell kann mehreren Services zugeordnet sein.

4.1.3 Weitere Beschreibungssprachen

Das *Semantic Web Services Framework (SWSF)* ist eine Erweiterung von OWL-S. Es umfasst die drei Hauptkomponenten *Service Descriptor* (Funktionalität), *Process Model* und *Grounding* (Aufrufmöglichkeiten auf technischer Ebene) [Ba05]. Um die Expressivität zu verbessern, wurde als Ontologiesprache die Semantic Web Services Language (SWSL) entwickelt, zu der jedoch keine relevanten Implementierungen oder Tools existieren [La07, 181].

METEOR-S unterscheidet sich grundlegend von den oben beschriebenen Ansätzen: Statt ein komplett neues Framework (*Top-Down-Ansatz*) zur semantischen Servicebeschreibung zu definieren, werden existierende Standards, insbesondere WSDL, um Semantik ergänzt (*Bottom-Up-Ansatz*). Vorteil dieser Strategie ist die Wiederverwendung bewährter Technologie. Die Sprache zur Definition des semantischen Modells ist nicht Bestandteil der Spezifikation. Somit können beliebige Ontologien referenziert werden, was Flexibilität bietet. Allerdings werden damit Fragestellungen wie Expressivität und Komplexität nicht gelöst, sondern nur verlagert. WSDL-S und SAWSDL wurden als Teil des Projekts METEOR-S bzw. aufgrund seiner Ergebnisse erarbeitet [NPF08, 297ff].

Web Service Semantics (WSDL-S) nutzt die Erweiterungselemente von WSDL, um Operationen, Ein- und Ausgaben sowie Vorbedingungen und Effekte mit externen Ontologien zu verknüpfen. Diese können in beliebigen Sprachen wie RDF, OWL, WSML oder UML definiert sein. Jedoch können verschiedene Sprachen inkompatibel sein [Ak05].

Semantic Annotations for WSDL and XML Schema (SAWSDL) ist eine Weiterentwicklung von WSDL-S und wurde 2007 als Empfehlung des W3C publiziert [FL07].

4.2 Vergleich von Semantic-Web-Services-Beschreibungssprachen

Für den Vergleich wurden die Sprachen WSMO, OWL-S und WSDL-S ausgewählt. Kriterien waren einerseits die Verbreitung der Sprache, die Tool-Unterstützung sowie die Berücksichtigung sowohl des Top-Down- als auch des Bottom-Up-Ansatzes. Bisher veröffentlichte Vergleiche wurden oft von Mitentwicklern einer Sprache vorgenommen und sind daher möglicherweise nicht als neutral einzuschätzen. Dies zeigt sich beispielsweise daran, dass zukünftige Entwicklungspläne für eine Sprache berücksichtigt werden, für andere jedoch nicht [La04, 254]. In diesem Beitrag wurden für den Vergleich nur in den Spezifikationen enthaltene Elemente übernommen.

4.2.1 Vergleichskriterien

Der Vergleich konzentriert sich auf die Typen der Ontologien, auf die Hauptelemente der Sprachen, die Abdeckung der in den einzelnen Phasen ablaufenden Aktivitäten, die Dokumentation und die Unterstützung durch Tools. Zudem greifen wir weitere in [K106, 39f, 81] verwendete Kriterien auf.

4.2.2 Vergleichsergebnisse

Die Auswertung zeigt bei vielen Kriterien Übereinstimmungen zwischen WSMO und OWL-S. Insbesondere enthalten beide Sprachen Elemente zur Beschreibung von Ontologien mit verschiedenen Stufen der Expressivität und Komplexität. Während OWL-S eine direkte Erweiterung von OWL ist, implementiert WSMO eine eigene Ontologiesprache, die teilweise zu OWL und RDF kompatibel ist. WSDL-S basiert auf einem grundlegend anderen Ansatz: Die darin unumgängliche Einbindung von externen Ontologien wird in der Literatur eher kritisch betrachtet [La07, 181].

Die drei Sprachen enthalten unterschiedliche Hauptelemente, was auf verschiedene Schwerpunkte zurückzuführen ist. So berücksichtigt WSMO die Integration heterogener Daten, während OWL-S eine umfassende Abbildung von Prozessen unterstützt.

Der Abgleich zwischen Anforderungen eines Nachfragers und der Funktionalität eines Web Service ist in WSMO und OWL-S unterschiedlich implementiert: WSMO stellt ein Hauptelement zur Verfügung, das es dem Nachfrager erlaubt, seine Anforderungen zu formulieren. OWL-S sieht explizit kein solches Element vor.

Alle drei Sprachen sind gut dokumentiert: OWL-S eher mathematisch-formal, WSMO konzeptuell, WSDL-S konzentriert sich auf die Implementierungsebene. Auf der Zwischenstufe ist keine Literatur vorhanden: Wünschenswert wäre z. B. die Formulierung einer betriebswirtschaftlichen Problemstellung und die Implementierung einer semantisch beschriebenen SOA zur Lösung dieses Problems. Dokumentierte Implementierungen größerer Anwendungsfälle scheinen nicht zu existieren.

Mit WSMO-Studio steht ein Plugin für Eclipse, mit dem OWL-S Editor eines für Protégé zur Verfügung. WSDL-S verfügt mit Radiant über ein Plugin zu Eclipse; allerdings wird in führenden kommerziellen XML-Editoren die semantische Erweiterung nicht unterstützt.

Die Beurteilung von Klein [Kl06, 81ff] zeigt weiteren Forschungsbedarf auf: Keine der untersuchten Beschreibungssprachen wurde durchgehend positiv evaluiert und für den praktischen Einsatz als geeignet bezeichnet. Eine eindeutige Beschreibung des Angebots ist nur in WSMO möglich. Nur OWL-S bietet einen teilweise funktionierenden Abgleich zwischen Anforderungen und Funktionalität.

5 Ein Anwendungsfall der Service-Auswahl: Umsetzung

Nachfolgend wird der konzeptionelle Ansatz aus Abschnitt 2.2 in den Beschreibungssprachen WSMO und OWL-S modelliert. Diese Umsetzungen wurden im Rahmen einer an unserem Institut erstellten Masterarbeit [He08a] vorgenommen. Auf eine Beschreibung in WSDL-S wird verzichtet, da sie wenig verbreitet ist und das Fehlen einer integrierten Ontologiesprache eher kritisch beurteilt wird.

Abbildung 2 zeigt den Ablauf bei Auswahl eines Services: Der Nachfrager (Prozess) will eine Transaktion protokollieren und sendet eine SOAP-Nachricht als Serviceanfrage an den Web Service. Aufgrund dieser Anfrage wird die korrekte Protokollierungsweise bestimmt. Semantische Konzepte erlauben Schlussfolgerungen über die Art der Daten und die passende Regulation. Beispiele für Regeln sind:

- Wenn eine Transaktion einen Namen, eine Kreditkartennummer und ein Ablaufdatum enthält, handelt es sich um eine Kreditkartenzahlung.
- Alle Kreditkartenzahlungen müssen gemäß PCI DSS Version 1.2 protokolliert werden [Pc08].

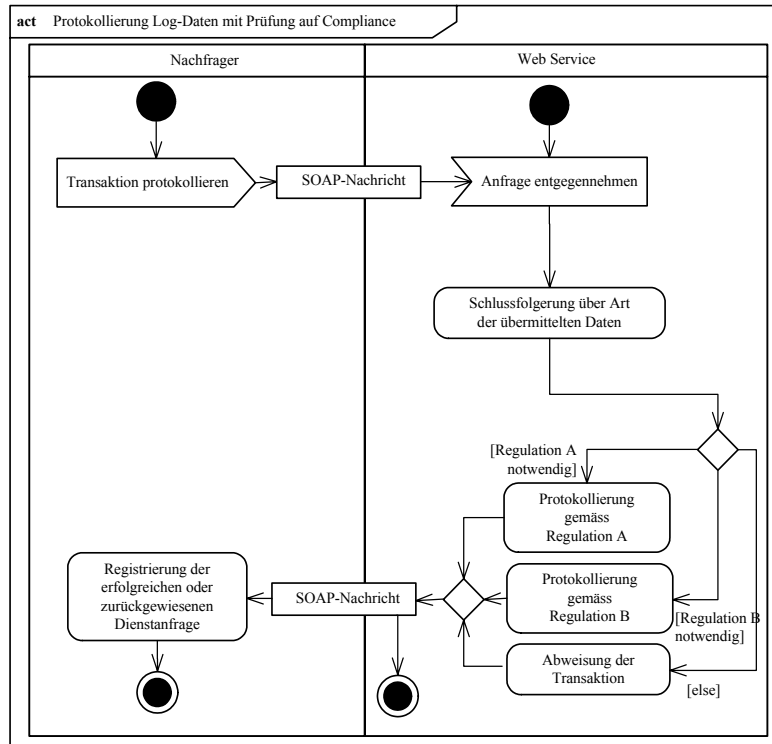


Abbildung 2: Protokollierung von Log-Daten mit Prüfung auf Compliance [He08a, 91]

5.1 Umsetzung in WSMO

Zur Umsetzung des Fallbeispiels wurde das Eclipse-Plugin WSMO Studio Version 0.7.3 eingesetzt. Als Variante für die Ontologiesprache wurde unter Berücksichtigung von Expressivität und Entscheidbarkeit WSML-Flight gewählt: Durch die Unterstützung von Elementen der logischen Programmierung können komplexere Zusammenhänge modelliert werden als in WSML-Core. So können beispielsweise Kardinalitäten festgelegt werden ("Eine Kreditkarte muss genau einen Namen haben"). WSML-Full wurde nicht verwendet, um die Entscheidbarkeit zu gewährleisten.

In Abbildung 3 sind die Anbieter der Kreditkarten, welche die Regulation PCI DSS verlangen, angegeben. Mit einer Hierarchiebeziehung sind sie dem übergeordneten Konzept CreditCard zugeordnet. Bei Regulationen werden zusätzlich Versionen erfasst, um bei ihrer Veränderung nur die Beziehung in der Ontologie anpassen zu müssen. Beim Abgleich werden ausschließlich Services berücksichtigt, welche die aktuelle Version der Regulation erfüllen. Daten wie Kreditkartennummer, Name und Ablaufdatum sind eigenständige Konzepte und werden dem Konzept CreditCard über Attribute zugeordnet. Da die einzelnen Transaktionsarten (z. B. hinsichtlich Speicherort) unterschiedlich gehandhabt werden müssen und sich die angebotenen Services auch darin unterscheiden, wurden diese ebenfalls in der Ontologie berücksichtigt.

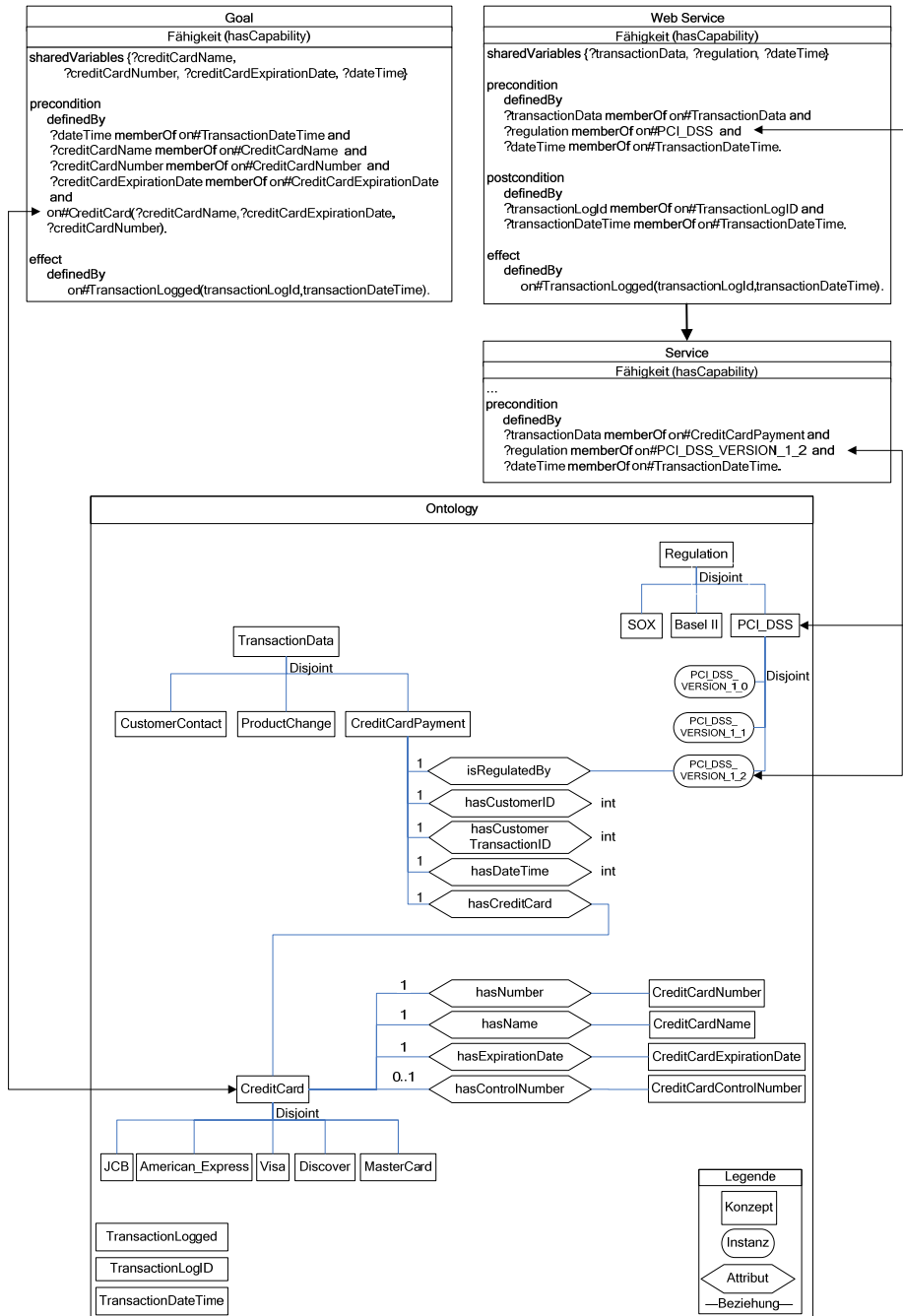


Abbildung 3: Discovery-Phase für den Anwendungsfall auf Basis von WSMO

In den Effekten wird gefordert, dass nach Ausführung des Service die Transaktion geloggt ist. Weiter ist definiert, dass es sich um Kreditkartendaten handelt: Die Kreditkartennummer, das Ablaufdatum und der Name werden auf das Konzept CreditCard übertragen. Gemäß Schlussfolgerung der Ontologie muss CreditCardPayment als Transaktionsart angewandt werden. Da der Web Service angibt, alle Transaktionsarten zu unterstützen, gilt er als wählbar.

Aufgrund der Transaktionsart CreditCardPayment wird abgeleitet, dass die Regulation PCI_DSS_VERSION_1_2 angewandt werden muss. Der Web Service gibt an, alle Versionen von PCI DSS zu unterstützen. Er gilt also als konform. Er ruft den entsprechenden Service auf, welcher die Transaktion nach der Art Kreditkartenzahlung und konform zu Version 1.2 von PCI DSS protokolliert.

5.2 Umsetzung in OWL-S

Zur Umsetzung des Fallbeispiels wurde der OWL-S Editor, ein Plugin für Protégé, eingesetzt. Aus den gleichen Gründen wie bei WSMO wurde als Variante für die Ontologiesprache OWL-DL gewählt. Im Folgenden werden primär die Unterschiede dargestellt.

Die Ontologien stimmen weitgehend überein. Die in OWL-S verwendete enthält zusätzlich notwendige und hinreichende Bedingungen. Eine notwendige Bedingung lautet: "Jede Kreditkarte muss genau einen Namen, eine Nummer und ein Ablaufdatum haben"; eine hinreichende Bedingung ist: "Jede Instanz, die einen Namen, eine Nummer und ein Ablaufdatum hat, muss eine Kreditkarte sein." Damit kann der Web Service den richtigen Schluss ziehen. In WSMO muss dafür ein entsprechendes Axiom deklariert werden.

Da OWL-S verschiedene Serviceprofile erlaubt, wurde für jede angebotene Transaktionsart (u. a. für CreditCardPayment) und für die Anfrage ein eigenes Profil erstellt. Eingabe sind Transaktionsdaten in Form von CreditCardPayment und Ausgabe die TransactionLogID. Im Gegensatz zum die externe Sicht repräsentierenden Serviceprofil ist nur ein für die Implementierung der Funktionalität zuständiges Servicemodell zugelassen. Im Anwendungsfall besteht diese Funktionalität im Wesentlichen aus drei Teilen: Der Erkennung der korrekten Transaktionsart, damit die Daten regulationskonform gespeichert werden können, der revisionssicheren Speicherung der Transaktion und der Rückgabe einer Bestätigung an den Kunden.

6 Verwandte Arbeiten

In der Einleitung wurde bereits auf Veröffentlichungen zu *SOA-Governance* hingewiesen. Jedoch wird dieser Begriff oft umfassender verwendet als nur im Hinblick auf Compliance-Nachweise und umfasst auch traditionelle Fragen des IT-Managements (zur Abgrenzung vgl. [KL06, 451ff]). In [JG07, 191] wird die Wiederverwendung in anderen Kontexten betrachtet, ohne auf die damit möglicherweise verbundenen Compliance-Probleme im Detail einzugehen. In [Fo06] werden zusätzliche Regulatorien und Aus-

wirkungen der Wiederverwendung von *bestehenden Services* in *neuen Prozessen* betrachtet; allerdings werden die in der anderen Richtung bestehenden Auswirkungen von *neu* im Repository verfügbaren *Services* auf *bestehende Prozesse* nicht untersucht.

Einen Überblick über aktuelle Arbeiten und offene Forschungsfragen zur Compliance-Prüfung in Geschäftsprozessen bietet [El08]. In [Og04] werden generische, allen Regulatorien zugrunde liegende Compliance-Services (z. B. Zugriffskontrollen) vorgeschlagen; jedoch ist zu beachten, dass gerade unterschiedliche Details der Regulatorien relevant sein können. Für die Erbringung der Compliance-Nachweise existieren Lösungsansätze, welche Prozesse, teilweise sogar zur Laufzeit, überprüfen. Allerdings sind einige nicht spezifisch auf eine SOA ausgerichtet [Gi05; Ag06; NS07a; NS07b; Ka08], andere prüfen Prozessänderungen nicht zur Laufzeit [LMX07; NS07a; NS07b]. Zusammengefasst lässt sich sagen, dass die meisten Ansätze detaillierte Vorgaben aus den Regulatorien modellieren. Allerdings ist fraglich, ob es möglich ist, *alle* Regulatorien in *allen* Aspekten detailliert zu modellieren und zur Laufzeit zu überprüfen.

Eine Übersicht über Arbeiten in den Bereichen semantische Konzepte und dynamische Bindung gibt [Ku08]. Die Arbeiten beziehen sich u. a. auf fehlerhafte und nicht passende Dienste, jedoch nicht unmittelbar auf Compliance zu Regulatorien.

7 Zusammenfassung und Ausblick

In diesem Beitrag wurde gezeigt, dass bei Wiederverwendung von Services in einem zuvor unberücksichtigten Kontext nicht konforme Services gewählt werden können und daher die Gefahr besteht, nicht alle erforderlichen Regulatorien zu erfüllen. Es wurde ein Ansatz vorgestellt, wie mit semantischen Konzepten der Kontext der Servicenutzung und, daraus folgend, die anzuwendenden Regulatorien bestimmt werden können. Zurzeit werden semantische Web Services in der Praxis kaum verwendet [HKZ08], stellen jedoch ein aktuelles Forschungsgebiet dar. Diese Konzepte können nicht nur bei der Service-Auswahl, sondern auch für Compliance-Nachweise hilfreich sein.

Zur Umsetzung des für einen Anwendungsfall entwickelten konzeptionellen Ansatzes wurden semantische Beschreibungssprachen verglichen und zwei Sprachen für die Umsetzung ausgewählt. Für die Ontologieerstellung wurde ein Bottom-Up-Ansatz gewählt [SH05, 82]: Für diesen Anwendungsfall und einer zugehörigen Regulation wurde eine erste Version erstellt; diese soll in einem nächsten Schritt durch Berücksichtigung weiterer Regulationen und Anwendungsfälle schrittweise erweitert werden. Zudem sollen bereits bestehende Ontologien berücksichtigt werden. Rechtsontologien (vgl. z. B. [GST05]) erscheinen als Basis wenig geeignet; Erfolg versprechender sind Regulationsontologien [Gi05]. Schließlich sollen alternative Implementierungsformen mit bewährten Technologien (wie z. B. Rules Engines; vgl. [JS04]) geprüft werden.

Literaturverzeichnis

- [Ag06] Agrawal, R.; Johnson, C.; Kiernan, J.; Leymann, F.: Taming Compliance with Sarbanes-Oxley Internal Controls Using Database Technology. In (Liu, L.; Reuter, A.; Whang, K.-Y.; Zhang, J. Hrsg.): Proc. 22nd International Conference on Data Engineering (ICDE 2006), Atlanta, 2006. IEEE Computer Society, 2006; S. 92.
- [Ak05] Akkiraju, R.; Farrell, J.; Miller, J.; Nagarajan, M.; Schmidt, M.-T.; Sheth, A.; Verma, K.: Web Service Semantics - WSDL-S, W3C Member Submission, <http://www.w3.org/Submission/WSDL-S/>, 2005.
- [Ba05] Battle, S.; Bernstein, A.; Boley, H.; Grosz, B.; Gruninger, M.; Hull, R.; Kifer, M.; Martin, D.; McIlraith, S.; McGuinness, D.L.; Su, J.; Tabet, S.: Semantic Web Services Ontology (SWSO), W3C Member Submission, <http://www.w3.org/Submission/SWSF-SWSO/>, 2005.
- [Be07] Bell, D.; De Cesare, S.; Iacovelli, N.; Lycett, M.; Merico, A.: A framework for deriving semantic web services. In: Information Systems Frontiers 9 (2007) 1; S. 69-84.
- [Bl07] Bloomberg, J.: The SOA Governance Timeline, Zapthink White Paper, <http://www.zapthink.com/report.html?id=WP-0153>, 2007.
- [Ci08] Cimpian, E.; Meyer, H.; Dumitru, R.; Sirbu, A.; Steinmetz, N.; Staab, S.; Toma, I.: Ontologies and Matchmaking. In (Kuropka, D.; Tröger, P.; Staab, S.; Weske, M. Hrsg.): Semantic Service Provisioning. Springer, Berlin, Heidelberg, 2008; S. 19-54.
- [Do05] Dostal, W.; Jeckle, M.; Melzer, I.; Zengler, B.: Service-orientierte Architekturen mit Web Services. Spektrum - Akademischer Verlag, München, 2005.
- [El08] El Kharbili, M.; Alves de Medeiros, A.K.; Stein, S.; van der Aalst, W.M.P.: Business Process Compliance Checking: Current State and Future Challenges. In (Loos, P.; Nüttgens, M.; Turowski, K.; Werth, D. Hrsg.): Modellierung betrieblicher Informationssysteme (MobIS 2008) - Modellierung zwischen SOA und Compliance Management, Lecture Notes in Informatics, P-141. Gesellschaft für Informatik, Bonn, 2008; S. 107-113.
- [Er04] Erl, T.: Service-Oriented Architecture. Prentice Hall PTR, Upper Saddle River, 2004.
- [Fe07] Fensel, D.; Lausen, H.; Polleres, A.; de Bruijn, J.; Stollberg, M.; Roman, D.; Domingue, J.: Enabling Semantic Web Services - The Web Service Modeling Ontology. Springer, Berlin, Heidelberg, 2007.
- [FL07] Farrell, J.; Lausen, H.: Semantic Annotations for WSDL and XML Schema, W3C Recommendation, <http://www.w3.org/TR/sawSDL/>, 2007.
- [Fo06] Foody, D.: The Challenges of SOA - Which rules are necessary and which are just nice to have. In: SOA Web Services Journal 6 (2006) 9; <http://webservices.sys-con.com/read/284550.htm>.
- [GHA07] Grimm, S.; Hitzler, P.; Abecker, A.: Knowledge Representation and Ontologies - Logic, Ontologies and Semantic Web Languages. In (Studer, R.; Grimm, S.; Abecker, A. Hrsg.): Semantic Web Services - Concepts, Technologies and Applications. Springer, Berlin, Heidelberg, 2007; S. 51-105.
- [Gi05] Giblin, C.; Liu, A.Y.; Müller, S.; Pfitzmann, B.; Zhou, X.: Regulations Expressed As Logical Models (REALM). In (Moens, M.-F.; Spyns, P. Hrsg.): Legal Knowledge and Information Systems (JURIX 2005) - Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, 2005; S. 37-48.
- [Gr93] Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. In: Knowledge Acquisition 5 (1993) 2; S. 199-220.
- [Gr07] Grimm, S.: Discovery - Identifying Relevant Services. In (Studer, R.; Grimm, S.; Abecker, A. Hrsg.): Semantic Web Services - Concepts, Technologies and Applications. Springer, Berlin, Heidelberg, 2007; S. 211-244.
- [GST05] Gangemi, A.; Sagri, M.; Tiscornia, D.: A constructive framework for legal ontologies. In (Benjamins, V.; Casanovas, P.; Breuker, J.; Gangemi, A. Hrsg.): Law and the Semantic Web. Springer, Berlin, Heidelberg, 2005; S. 97-124.

- [He02] Hesse, W.: Ontologie(n) - Aktuelles Schlagwort. In: Informatik-Spektrum 25 (2002) 6; S. 477-480.
- [He08a] Heim, D.: Darstellung und Vergleich ausgewählter Semantic-Web-Services-Beschreibungssprachen, Masterarbeit, Institut für Wirtschaftsinformatik der Universität Bern, 2008.
- [He08b] Hepp, M.: Ontologies: State of the Art, Business Potential, and Grand Challenges. In (Hepp, M.; De Leenheer, P.; de Moor, A.; Sure, Y. Hrsg.): *Ontology Management - Semantic Web, Semantic Web Services, and Business Applications*. Springer, Berlin, Heidelberg, 2008; S. 3-22.
- [HF08] Hench, G.; Fensel, D.: From Web to Semantic Web. In (Fensel, D.; Kerrigan, M.; Zarella, M. Hrsg.): *Implementing Semantic Web Services, The SESA Framework*. Springer, Berlin, Heidelberg, 2008; S. 3-25.
- [HKZ08] Haniewicz, K.; Kaczmarek, M.; Zyskowski, D.: Semantic Web Services Applications - a Reality Check. In: *Wirtschaftsinformatik 50 (2008) 1*; S. 39-46.
- [JG07] Johannsen, W.; Goeken, M.: Referenzmodelle für IT-Governance - Strategische Effektivität und Effizienz mit COBIT, ITIL & Co. dpunkt.verlag, Heidelberg, 2007.
- [Jo08] Josuttis, N.: *SOA in der Praxis - System-Design für verteilte Geschäftsprozesse*. dpunkt.verlag, Heidelberg, 2008.
- [JS04] Jang, M.; Sohn, J.-C.: Bossam: An Extended Rule Engine for OWL Inferencing. In (Antonioni, G.; Boley, H. Hrsg.): *RuleML 2004 - Rules and Rule Markup Languages for the Semantic Web, Lecture Notes in Computer Science 3323/2004*. Springer, Berlin, Heidelberg, 2004; S. 128-138.
- [Ka08] Karagiannis, D.: A Business Process-Based Modelling Extension for Regulatory Compliance. In (Bichler, M.; Hess, T.; Krcmar, H.; Lechner, U.; Matthes, F.; Picot, A.; Speitkamp, B.; Wolf, P. Hrsg.): *Proc. der Multikonferenz Wirtschaftsinformatik (MKWI 2008)*, München, 2008. GITO-Verlag, Berlin, 2008; S. 1159-1173.
- [KBS04] Krafzig, D.; Banke, K.; Slama, D.: *Enterprise SOA - Service-Oriented Architecture Best Practices*. Prentice Hall PTR, Upper Saddle River et al., 2004.
- [Kl06] Klein, M.: *Automatisierung dienstorientierten Rechnens durch semantische Dienstbeschreibungen*, Dissertation, Fakultät für Mathematik und Informatik der Friedrich-Schiller-Universität Jena. Universitätsverlag Karlsruhe, Karlsruhe, 2006.
- [KL06] Knolmayer, G.; Loosli, G.: IT Governance. In (Zaug, R.J. Hrsg.): *Handbuch Kompetenzmanagement*. Haupt Verlag, Bern, Stuttgart, Wien, 2006; S. 449-457.
- [Kn07] Knolmayer, G.: Compliance-Nachweise bei Outsourcing von IT-Aufgaben. In: *Wirtschaftsinformatik 49 (2007) Sonderheft*; S. S98-S106.
- [KSH08] Kohnke, O.; Scheffler, T.; Hock, C.: SOA-Governance - Ein Ansatz zum Management serviceorientierter Architekturen. In: *Wirtschaftsinformatik 50 (2008) 5*; S. 408-412.
- [Ku08] Kuroпка, D.; Tröger, P.; Staab, S.; Weske, M. (Hrsg.): *Semantic Service Provisioning*. Springer, Berlin, Heidelberg, 2008.
- [La04] Lara, R.; Dumitru, R.; Polleres, A.; Fensel, D.: A Conceptual Comparison of WSMO and OWL-S. In (Zhang, L.-J.; Jeckle, M. Hrsg.): *Web Services, Lecture Notes in Computer Science 3250/2004*. Springer, Berlin, Heidelberg, 2004; S. 254-269.
- [La07] Lausen, H.; Lara, R.; Polleres, A.; de Bruijn, J.; Roman, D.: Description - Semantic Annotation for Web Services. In (Studer, R.; Grimm, S.; Abecker, A. Hrsg.): *Semantic Web Services - Concepts, Technologies and Applications*. Springer, Berlin, Heidelberg, 2007; S. 179-209.
- [Li07] Lickel, C.W.: Introduction. In: *IBM Systems Journal 46 (2007) 2*; S. 202.
- [LMX07] Liu, Y.; Müller, S.; Xu, K.: A static compliance-checking framework for business process models. In: *IBM Systems Journal 46 (2007) 2*; S. 335-361.

- [Lo08] Loosli, G.: Compliance-Prüfung bei Anwendung dynamischer Bindung in service-orientierten Architekturen. In (Loos, P.; Nüttgens, M.; Turowski, K.; Werth, D. Hrsg.): Modellierung betrieblicher Informationssysteme (MobIS 2008) - Modellierung zwischen SOA und Compliance Management, Lecture Notes in Informatics, P-141. Gesellschaft für Informatik, Bonn, 2008; S. 7-21; erscheint in überarbeiteter Form in: Enterprise Modelling and Information Systems Architectures 4 (2009).
- [Ma04] Martin, D.; Burstein, M.; Hobbs, J.; Lassila, O.; McDermott, D.; McIlraith, S.; Narayanan, S.; Paolucci, M.; Parsia, B.; Payne, T.; Sirin, E.; Srinivasan, N.; Sycara, K.: OWL-S: Semantic Markup for Web Services, W3C Member Submission, <http://www.w3.org/Submission/OWL-S/>, 2004.
- [MSS01] Mädche, A.; Staab, S.; Studer, R.: Ontologien. In: Wirtschaftsinformatik 43 (2001) 4; S. 393-395.
- [Ni08] Niemann, M.; Eckert, J.; Repp, N.; Steinmetz, R.: Towards a Generic Governance Model for Service-oriented Architectures. In (Association for Information Systems (AIS) Hrsg.): Proc. of the Fourteenth Americas Conference on Information Systems (AMCIS 2008), Toronto, 2008, <http://aisel.aisnet.org/amcis2008/361/>, 2008.
- [No09] NorthPage Research: SOA Governance Resource Guide, <http://soagovsource.com/>, 2009.
- [NPF08] Norton, B.; Pedrinaci, C.; Fensel, D.: Compatible and Related Systems. In (Fensel, D.; Kerrigan, M.; Zaremba, M. Hrsg.): Implementing Semantic Web Services, The SESA Framework. Springer, Berlin, Heidelberg, 2008; S. 285-301.
- [NS07a] Namiri, K.; Stojanovic, N.: A Model-driven Approach for Internal Controls Compliance in Business Processes. In (Hepp, M.; Hinkelmann, K.; Karagiannis, D.; Klein, R.; Stojanovic, N. Hrsg.): Proc. of the Workshop on Semantic Business Process and Product Lifecycle Management (SBPM 2007), Innsbruck, 2007. CEUR Workshop Proceedings, Vol. 251, 2007; S. 40-43.
- [NS07b] Namiri, K.; Stojanovic, N.: Applying Semantics to Sarbanes Oxley Internal Controls Compliance. In (Koschke, R.; Herzog, O.; Rödiger, K.-H.; Ronthaler, M. Hrsg.): INFORMATIK 2007: Informatik trifft Logistik, Beiträge der 37. Jahrestagung der Gesellschaft für Informatik, Lecture Notes in Informatics, P-109. Gesellschaft für Informatik, Bonn, 2007; S. 222-226.
- [Og04] O'Grady, S.: SOA Meets Compliance: Compliance Oriented Architecture, Study, RedMonk, http://redmonk.com/public/COA_Final.pdf, 2004.
- [Pc08] Payment Card Industry (PCI) Security Standards Council: Payment Application Data Security Standard (PA-DSS), http://www.pcisecuritystandards.org/security_standards/pa_dss.shtml, 2008.
- [PG03] Papazoglou, M.P.; Georgakopoulos, D.: Service-Oriented Computing. In: Communications of the ACM 46 (2003) 10; S. 25-28.
- [PLL06] Polleres, A.; Lausen, H.; Lara, R.: Semantische Beschreibung von Web Services. In (Pellegrini, T.; Blumauer, A. Hrsg.): Semantic Web - Wege zur vernetzten Wissensgesellschaft. Springer, Berlin, Heidelberg, 2006; S. 505-524.
- [Pr07] Preist, C.: Goals and Vision - Combining Web Services with Semantic Web Technology. In (Studer, R.; Grimm, S.; Abecker, A. Hrsg.): Semantic Web Services - Concepts, Technologies and Applications. Springer, Berlin, Heidelberg, 2007; S. 159-178.
- [RLK06] Roman, D.; Lausen, H.; Keller, U.: Web Service Modeling Ontology (WSMO), <http://www.wsmo.org/TR/d2/v1.3/>, 2006.
- [SD05] Saadadi, S.; Denker, G.: An OWL-S Editor Tutorial, <http://owlseditor.semwebcentral.org/documents/tutorial.pdf>, 2005.
- [SH05] Stuckenschmidt, H.; van Harmelen, F.: Information Sharing on the Semantic Web. Springer, Berlin, Heidelberg, 2005.
- [SS07] Schelp, J.; Stutz, M.: SOA Governance. In: HMD - Praxis der Wirtschaftsinformatik 43 (2007) 253; S. 66-73.