

Semantic Service Finder

Autoren: Eddie Mönch, Daniel Rabus,
Michael Erdmann und Jürgen Angele

im Juni 2007

Inhaltsverzeichnis

1	Änderungsdokumentation	3
2	Fachliche Kurzbeschreibung.....	4
3	Einführung	5
3.1	Architektur.....	5
4	Semantische Annotierung von Web Services	7
5	Semantisches Finden von Web Services	9
6	Einbinden unterschiedlicher Repositories	11
7	Literatur	13
8	Abbildungsverzeichnis	14
9	Impressum	15

1 Änderungsdokumentation

Änderungen	Autor	Datum
Erstellt	E. Mönch	01.06.2007
Erweiterung um das Einbinden unterschiedlicher Repositories	E. Mönch	19.06.2007

2 Fachliche Kurzbeschreibung

Dieses Konzept beschreibt eine Suchmaschine, die es Benutzern erlaubt, aktuelle Informationen zu verfügbaren Web Services zu finden. Das Konzept beruht auf zwei Technologien: Zum einen im Bereich des dem Erstellens bzw. des Deployments von Web Services auf der semantischen Annotation und zum anderen zum Wiederauffinden der Dienste auf dem Ansatz der semantischen Suche.

Die Annotierung von Web Services nach dem [SAWSDL] Standard des W3C [Word Wide Web Consortium - <http://www.w3.org/>] erlaubt eine semantische Beschreibung der Web Services.

Die semantische Suche ermöglicht zu einer gegebenen Anfrage, die aus Kontextinformationen, Parametern wie Input- und Outputwerten, der textuellen Beschreibung des Services und der semantischen Annotierung des Web Service besteht, das sogenannte Service Discovery auf Basis der Metadaten und Artefakten der Dienste.

Ein Redaktionsteam, das Webservices beim Erstellen derer semantisch annotiert kann sehr schnell zu einem Engpass werden. Oft sind semantische Annotationen von Web Services auch überhaupt nicht möglich, da diese von einer Software automatisch erstellt werden. Deshalb ist der Ansatz der semantischen Suche unabhängig vom Vorhandensein einer semantischen Annotierung (wobei diese natürlich die Trefferqualität verbessert). Das Prinzip der semantischen Suche wurde grundlegend beschrieben in [Moench et al. 2003] und [Angele et al. 2006]. In diesem Whitepaper hier stellen wir dieses Prinzip nochmals für die spezielle Aufgabe des Web Service Discovery vor.

3 Einführung

Ein Web Service ist im Prinzip eine Software-Anwendung, deren Schnittstellen als XML-Artefakte definiert, beschrieben und gefunden werden können. Ein Web Service unterstützt die direkte Interaktion mit anderen Software-Agenten unter Verwendung XML-basierter Nachrichten durch den Austausch über internetbasierte Protokolle.

3.1 Architektur

Web Services folgen dem Client/Server Paradigma. Sie sind Programme die auf Servern laufen und von Clients im Allgemeinen Anfragen gesendet bekommen diesen mit der gewünschten Information antworten.

Web Services orientieren sich an der Serviceorientierten Architektur (SOA) und vereinen daher verteilte und objektorientierte Programmierstandards und richten sich auf betriebswirtschaftliche Lösungen im Internet.

Es lassen sich die Instanzen Konsument, Anbieter und Verzeichnis identifizieren.

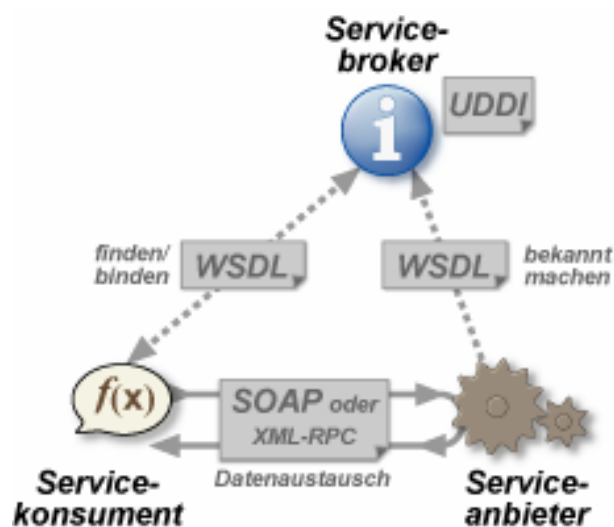


Abb. 1 Funktionsweise von Web Services

Der Anbieter veröffentlicht in einem Verzeichnis die Beschreibung seiner Dienste. Der Konsument durchsucht das Verzeichnis und wählt den gewünschten Dienst aus. Nachdem eventuell weitere Protokolldetails ausgetauscht werden, findet die dynamische Anbindung des Konsumenten an den Anbieter statt. Der Konsument greift nun auf Methoden zurück.

Die Grundlage hierbei bilden drei Standards, die jeweils auf XML basieren:

UDDI als Verzeichnisdienst zur Registrierung von Web Services. Es ermöglicht das dynamische Finden des Web Services (z. B. den Dienst wsCustomerCheck) durch den Konsumenten.

WSDL zur Beschreibung der unterstützten Methoden (z. B. CreditRating) und deren Parametern (z. B. Datum) für den Programmierer.

SOAP (oder XML-RPC) zur Kommunikation. Hier wird der eigentliche Aufruf gestartet.

Web Services bilden die drei wichtigsten Teile der Zusammenarbeit zwischen Client und Server ab: Das Zusammenfinden, Binden und den Datenaustausch.

Erreichbar sind Web Services über einen eindeutigen URI. Die verwendeten plattformunabhängigen Standards sind in der Lage, entfernte Methodenaufrufe beliebiger Plattformen zu dekodieren und einer Anwendung weiterzuleiten. Auf diese Weise entsteht eine verteilte Architektur. Die Kommunikation mit Web Services erfolgt über Nachrichten, die über mehrere Protokolle transportiert werden können.

4 Semantische Annotierung von Web Services

Verschiedene semantische Web Service Beschreibungsverfahren existieren: WSMO, OWL-S und SAWSDL. Semantic Annotations for WSDL [SAWSDL] ist eine Empfehlung des W3C (Candidate Recommendation). Es basiert auf der W3C Member Submission für WSDL-S, die unter anderem von IBM getrieben wurde. Dabei nutzt es den bereits standardisierten WSDL 2.0 Erweiterungsmechanismus aus um auf einfache und generische Art die semantische Annotation von Web Services zu unterstützen. Aufgrund dieser Konformität zum WSDL-Standard sind durch SAWSDL annotierte Web Services kompatibel zu gängigen Web Service Clients und Servern. Somit können zur Unterstützung von semantischen Annotationen für Web Services vorhandene Web Service Systeme verwendet werden. Diese Kompatibilität und der leichtgewichtige Ansatz ist genau der Vorteil gegenüber WSMO und OWL-S.

Stellen Sie nun sich vor, wir hätten folgende zwei Webservice vorliegen:

Einen Web Service, der die Bonität von Personen aufgrund ihrer Sozialversicherungsnummer ermittelt (wsCreditRating) und einen Web Service, der den aktuellen Kurs einer Aktie ermittelt aufgrund seiner Wertpapierkennnummer.

wsCreditRating:	in(string),out(float)
wsStockInfo:	in(string), out(float)

Man sieht, dass beide Web Services eine identische WSDL Definition im Bezug auf In- und Output haben. Das heißt, eine Standardsuche, wie Sie von den gängigen UDDI-Repositories durchgeführt wird, reicht nicht aus um die Services zu unterscheiden.

Die Dienste unterscheiden sich auf syntaktischer Ebene nicht. Man benötigt vielmehr eine semantische Beschreibung auf WSDL Basis.

SAWSDL erweitert WSDL 2.0 um Ontologie-basierte Annotationen (Konzepte) des Web Service durch die sogenannte modelReference und zwei weiteren Attributen.

Unter Verwendung von SAWSDL würden die In- und Output Informationen in obigem Beispiel wie folgt aussehen:

wsCreditRating:	in(string/Person.ssn), out(float/Person.hasRating)
wsStockInfo:	in(string/Stock.id), out(float/Stock.value)

Die komplette Beschreibung der beiden Web Services sieht im F-Logic Formalismus (vgl. [F-Logic]) wie folgt aus:

Der Web Service für die Bonitätsprüfung:

```
// Schema information
Person[ssn => xsd#string; hasRating => xsd#float].

// Credit rating service
wsCreditRating:WebService[
  desc → "Simple credit rating service, requires the SSN of a person.";
  input → p(Person, ssn);
  output → p(Person, hasRating);
  encryption → true;
  wsdl → "http://xxx"].
```

Und der Web Service für die Abfrage der Aktienkurse:

```
// Schema information
Stock[id => xsd#string; value => xsd#float].

// Stock info service
wsStockInfo:WebService[
  desc → "Returns the value of a stock, identified by its ID.";
  input → p(Stock, id);
  output → p(Stock, value);
  encryption → false;
  wsdl → "http://yyy"].
```

5 Semantisches Finden von Web Services

Das Web Service Discovery zu einer Suchanfrage findet stufenweise statt. Im ersten Schritt erfolgt eine semantische Suche in den Feldern

Name:

Zerlegung des Namens mittels linguistischer und Regel-basierter Verfahren in seine Kernbestandteile. In den Beispielen:

- “wsCreditRating” wird zu “ws Credit Rating”
- “wsStockInfo” wird zu “ws Stock Info”

Zu den einzelnen Wörtern können dann mittels der semantischen Suchtechnologie des SemanticMiner, unter Berücksichtigung von Tippfehlern und Mehrworten, Synonyme und Verfeinerungen hinzugezogen werden, die das Service Discovery verbessern.

Beschreibungsfeld:

Das Beschreibungsfeld wird mittels der linguistischen und semantischen Technologie des SemanticMiner durchsucht. Für eine Beschreibung der Funktionsweise dessen siehe [Moench 2006].

SAWSDL Annotationen (falls vorhanden):

Die Annotationen der Web Services stellen ein direktes Matching mit der Ontologie dar und geben direkt Auskunft über den Kontext des Web Services.

Danach oder gleichzeitig kann die Ergebnismenge noch über Filter wie die folgenden eingeschränkt werden:

Kontext des Web Services bzw. des Benutzers:

Der Kontext spiegelt eine in Regeln beschriebene und Ontologie-basierte Abbildung von Profilen für gewisse Szenarien und Use-Cases dar.

Inputparameter

Die annotierten Input-Parameter können unter Ausnutzung der Ontologie und den darauf basierenden Matchingregeln nicht nur 100% Treffer (also ja/nein Hits) liefern, sondern auch Matchings zwischen 0% und 100%, je nachdem wie weit entfernt ein Treffer in der Ontologie zum eingegebenen Suchbegriff und des annotierten Konzepts liegt.

Outputparameter:

Die annotierten Output-Parameter können unter Ausnutzung der Ontologie und den darauf basierenden Matchingregeln nicht nur 100% Treffer (also ja/nein Hits) liefern, sondern auch Matchings zwischen 0% und 100%, je nachdem wie weit entfernt ein Hit in der Ontologie zum eingegebenen Suchbegriff und des annotierten Konzepts liegt.

Die unten stehende Abbildung illustriert den Prozess nochmals:

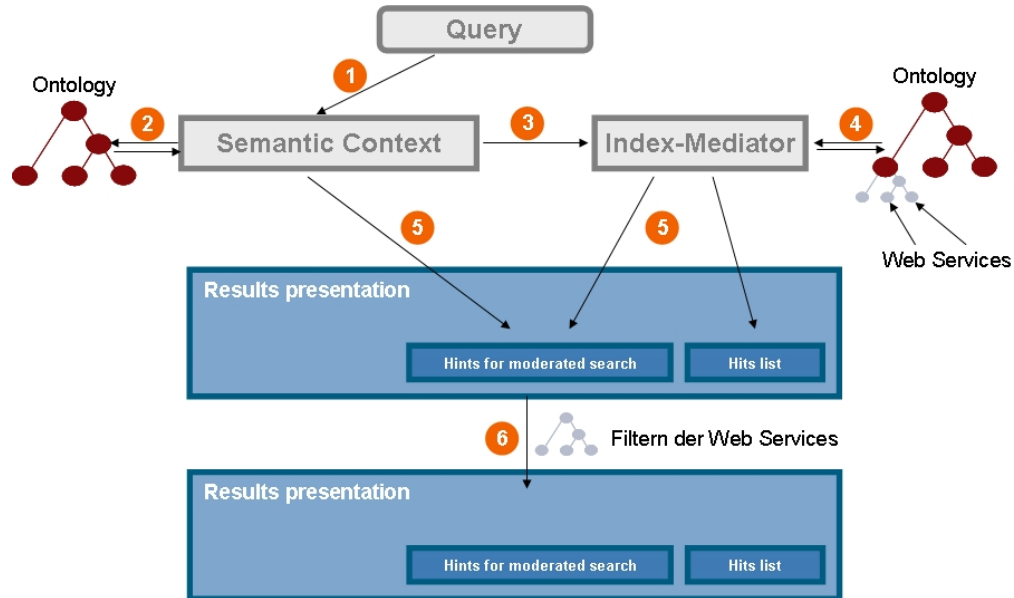


Abb. 2 Ablaufdiagramm des semantischen Web Service Discovery

Somit werden gängige Service Discovery Konzepte erweitert durch Matching von In- und Output Spezifikationen, semantischer Suche und Regel basierte Kontext bzw. Profileinschränkungen.

Eine beispielhafte Web Serviceanfrage sieht somit auf Ontologie-Beschreibungsebene wie folgt aus:

```
// Find a credit rating service
FORALL ws, wsdl ←
  ws:WebService AND
  matchesInput(ws, p(Adult, ssn)) AND
  matchesOutput(ws, p(Adult, hasRating)) AND
  matchesDescription(ws, "loan") AND
  matchesContext(ws, banking) AND
  ws[wsdl->wsdl].
```

Diese Anfrage gibt somit hier im Beispiel den Service wsCreditRating zurück.

6 Einbinden unterschiedlicher Repositories

Web Service Verzeichnisdienste ermöglichen das Finden und Wiederverwenden von Diensten und der zugehörigen Metadaten und Artefakte. Sie enthalten sowohl die fachlichen Beschreibungen der Service-Komponenten auf Basis der Unternehmensprozesse als auch die technische Dokumentation der angebotenen Services wie Metadaten und Annotationen.

Die Verzeichnisdienste gliedern sich in zwei Funktionsbereiche:

- Service Repository: Hier werden die
 - Definitionen der Prozesse und Services, die
 - Metadaten der Services, sowie das
 - zentrale Modell gespeichert.
- *Service Registry*: Hier werden die
 - „Gelben Seiten“ der Services,
 - Deployment Informationen, das
 - Service Management, sowie die
 - Nutzungsinformationen wie Navigation, Auffinden, und Nutzung der Services abgelegt.

An Spezifikationen dieser Verzeichnisdienste wären zu nennen

- Universal Description, Discovery, and Integration (UDDI),
- ebXML Registry und
- Directory Services Markup Language (DSML).

Wobei sich UDDI mittlerweile als De-Facto-Standard in der Industrie durchgesetzt hat. Zu nennen wären hier beispielsweise drei SOA Registries und Repositories namhafter Hersteller wie das

- WebSphere Service Registry and Repository (WSSR) von IBM, das
- Enterprise Services Repository (ESR) von SAP und
- CentraSite von der Software AG.

All diese Verzeichnisdienste implementieren das UDDI.

Die semantische Suche nach Web Services, also das Service Discovery, auf Basis solcher Verzeichnisdienste gestaltet sich somit wie folgt:

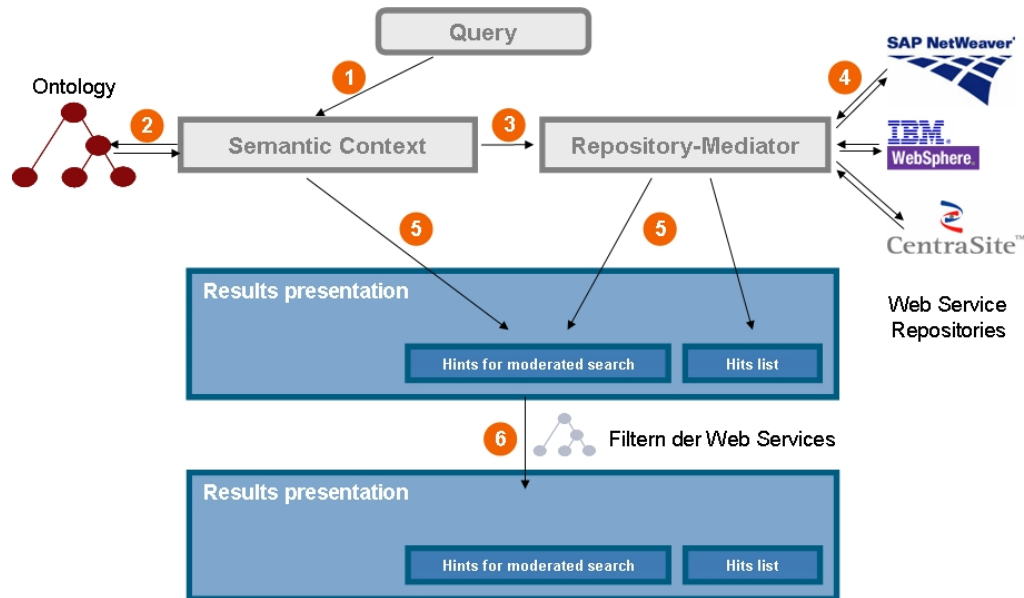


Abb. 3 Ablaufdiagramm der semantischen Web Service Discovery in SOA Registries und Repositories

Wie in Abb. 3 illustriert, besteht der Unterschied zum oben dargestellten Prozess des Service Finders darin, dass die Repositories im Repository Mediator registriert werden, um eine Anfrage auf die jeweilige Syntax der Repositories über die Semantik des Ontologie-basierten Modells zu übersetzen. Es können beispielsweise auch SOA Governance Regularien in Logik Formalismen (Regeln) abgebildet werden um den Prozess des Service Discovery konform zu Unternehmensrichtlinien gestalten zu können. In diesen Verzeichnisdiensten liegen in den Metadaten auch die Annotationen der Services, die vom Repository Mediator zur semantischen Suche verwendet werden. IBM WebSphere bietet darüber hinaus noch die Möglichkeit eine Ontologie in OWL (Web Ontology Language) im Repository zu hinterlegen, die als Klassifikationssystem im WSRR verwendet werden kann und somit noch weitere Einschränkungen und schärfere Treffer in den Repositories erlaubt.

7 Literatur

[Angele et al. 2006] Jürgen Angele, Eddie Mönch, Andreas Nierlich, Heiko Rudat, Hans-Peter Schnurr: *Anwendungen und Good Practices Semantischer Technologien*. In: *Semantic Web*, X.media.press, Springer Berlin Heidelberg, Seiten 337-356, ISBN 978-3-540-29324-8, 2006.

[F-Logic] ontoprise GmbH: *How to Write F-Logic Programs. A Tutorial for the Language F-Logic*.

http://www.ontoprise.de/documents/tutorial_flogic.pdf.

[Moench 2006] Eddie Moench: *Ähnlichkeitssuche auf Basis der SemanticMiner® Technologie*, Tutorial Series, Januar 2006.

[Moench et al. 2003] Eddie Moench, Mike Ullrich, Hans-Peter Schnurr, Juergen Angele: *SemanticMiner - Ontology-Based Knowledge Retrieval*. In: Special Issue of selected papers of the WM2003 in the Journal of Universal Computer Science (J.UCS).

[SAWSDL] Semantic Annotations for Web Services Description Language - <http://www.w3.org/2002/ws/sawSDL/>.

8 **Abbildungsverzeichnis**

Abb. 1	Funktionsweise von Web Services	5
Abb. 2	Ablaufdiagramm des semantischen Web Service Discovery	10
Abb. 3	Ablaufdiagramm der semantischen Web Service Discovery in SOA Registries und Repositories	12

9 Impressum

Editor

ontoprise GmbH
Amalienbadstraße 36 (Raumfabrik 29)
76227 Karlsruhe
Germany

Telefon +49 (0) 721 / 509 809 10

Telefax +49 (0) 721 / 509 809 11

Email info@ontoprise.de

Internet <http://www.ontoprise.de>

© ontoprise GmbH

Alle Rechte vorbehalten

Dieses Werk ist einschließlich all seiner Teile urheberrechtlich geschützt. Jede Verwertung, auch in Teilen, die über die Grenzen des engeren Urheberrechtsgesetzes hinausgeht, ist ohne schriftliche Zustimmung der ontoprise GmbH unzulässig und strafbar. Dies gilt insbesondere für Übersetzungen, Vervielfältigungen und Weitergaben jeder Art. Alle weiteren Rechte bleiben vorbehalten. Jegliche kommerzielle Nutzung, insbesondere die Verbreitung der Texte unter Erhebung eines Entgeltes für die Übertragung der Texte, für ein Medium oder für die Nutzung eines Informations-Systems, bleibt vorbehalten und ist nicht gestattet.

Für die hier zugänglichen Informationen übernehmen die Autoren und die ontoprise GmbH keine Form der Gewährleistung. Weiterhin wird keine Haftung für die Korrektheit, Vollständigkeit, Wirksamkeit oder Anwendbarkeit der enthaltenen Informationen und der vorgeschlagenen Maßnahmen übernommen. Insbesondere wird keine Verantwortung für die Verwendung der enthaltenen Informationen und eventuell dadurch entstehende Schäden übernommen. Viele Zitate und Wiedergaben von Informationen Dritter stammen aus öffentlich zugänglichen Dokumenten. Für die Richtigkeit von Angaben dieser Art wird im Speziellen keinerlei Gewährleistung übernommen.

Es wird darauf hingewiesen, dass die in dieser Dokumentation verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen und damit Eigentum Ihrer Inhaber sind.

Karlsruhe, im Juni 2007