

Java aktuell

Praxis. Wissen. Networking. Das Magazin für Entwickler

Java trägt Früchte

Neue Technologie

Java-API für Batch-Anwendungen, Seite 8

Entwicklungsstrategien

Software modular aufbauen, Seite 19

Mehr selbst entwickeln, Seite 23

Web-Entwicklung

Das Web als Social Network, Seite 32

Neu: Wicket 6, Seite 40

Java und Oracle

ADF Essential Mobile, Seite 52

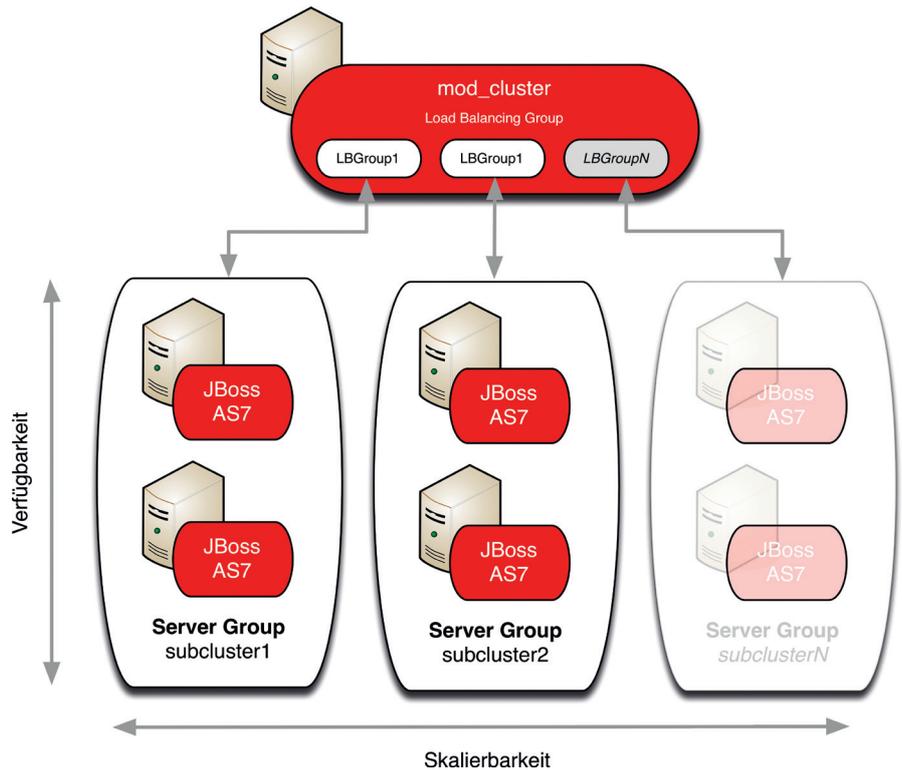


Sonderdruck

D: 4,90 EUR A: 5,60 EUR CH: 9,80 CHF Benelux: 5,80 EUR ISSN 2191-6977



- 3 Editorial
- 5 Das Java-Tagebuch
*Andreas Badelt,
Leiter der DOAG SIG Java*
- 8 Java-API für Batch-Anwendungen
Peter Doschkinow
- 11 Skalierbare Cluster-Topologien mit dem JBoss AS 7
Heinz Wilming und Immanuel Sims
- 15 Lightweight AOP mit CDI und JPA
Prof. Dr. Michael Helbig
- 19 Software modular bauen
Ulf Fildebrandt
- 23 Software – weniger kaufen, mehr bauen
Erik Dörnenburg
- 27 Diagramm-Bibliotheken für Android
Falko Krische
- 32 WebID: Das Web als Social Network
Angelo Veltens
- 37 Vermittlungsinstanzen sind gefragt
Tobias Hartwig
- 40 Anspruchsvolle Web-Anwendungen mit Wicket 6
Jochen Mader
- 44 „Java ist Grundlage für meine Software-Projekte ...“
Interview mit Dirk Dittert
- 45 Nachgefragt ...
Uwe Sauerbrei
- 49 Den Rollläden steuern mit Pi
René Jahn
- 52 ADF Essentials Mobile – Ein unlösbares Problem?
Markus Klenke
- 55 Performance-Falle beim Logging: Die Zeitstempel-Formatierung
Jürgen Lampe
- 61 Unbekannte Kostbarkeiten des SDK Heute: Web-Services
Bernd Müller
- 63 Integration von Java via JNI in traditionelle Cobol-Batch-Anwendungen
Marc Bauer und Martin Kiefer
- 43 Unsere Inserenten
- 66 Impressum



Ein Lastverteiler sorgt dafür, dass Lastverteilung und Failover nur innerhalb eines Subclusters stattfinden. Skalierbare Cluster-Topologien mit dem JBoss AS 7, Seite 11



Wer auf keine Funktion verzichten möchte und schon immer von einer eigenen Rollladensteuerung träumte, findet hier eine Anleitung für den Raspberry Pi, Seite 49



WebID: Das Web als Social Network

Angelo Veltens, <http://datenwissen.de>

Mit Linked Data lassen sich Daten im Web verbinden. Weitet man das Konzept auf Personen und deren Interaktion aus, entsteht ein „Social Web of Data“. Anstelle Dutzender Accounts bei unzähligen Diensten tritt eine globale Identität, die WebID. Sie dient Angeboten im WWW als Quelle von Profil-Informationen und zur Authentifizierung. Auch als Single Sign-on im Unternehmens-Intranet kann WebID eine Option sein.

Wer heute im Web aktiv ist, muss eine Vielzahl von Nutzer-Accounts bei unterschiedlichen Diensten anlegen und verwalten. Dutzende Profile wollen ausgefüllt und gepflegt werden, verlangen Nutzernamen und Passwörter. Dies alles scheint im Social Web nötig zu sein, um Nutzer zu identifizieren und sich mit anderen Nutzern in Kontakt bringen zu können.

Doch die soziale Interaktion ist in der Regel auf die jeweiligen Dienste begrenzt und lässt sich bestenfalls auf sogenannte Social Networks wie Facebook, Twitter und Google+ ausweiten. Letztere werden auch oft als Authentifizierungsdienst herangezogen. Dies führt zu einer Zentralisierung des eigentlich dezentralen Social Webs zu monolithischen Social-Network-Plattformen. Dienste, die sich zur Authentifizierung auf diese Plattformen verlassen, werden von deren API und ihren Beschränkungen abhängig.

Mit Linked Data und dem WebID-Projekoll ist ein Ausweg aus dem Dilemma in

Sicht. Statt unzähliger Accounts erhalten die Nutzer eine Identität im Web, die sie bei Bedarf selbst hosten können und über die sie den genutzten Diensten beliebige Profil-Informationen bereitstellen können. Dadurch wird das Web selbst zu einem dezentralen sozialen Netzwerk.

Rückblick: Daten verlinken

In den letzten Ausgaben wurde Linked Data thematisiert. Das WWW ist bereits heute ein Netz aus miteinander verlinkten Dokumenten. Linked Data weitet dieses Konzept der Links auf Daten aus, indem es nicht nur Dokumente, sondern auch Dinge über URIs identifiziert und miteinander in Beziehung setzt. Dazu dient das Datenmodell RDF, das Daten als „Tripel“ abbildet. Ein solches Tripel besteht aus Subjekt, Prädikat und Objekt. Subjekt und Objekt werden über das Prädikat miteinander verlinkt.

Im Social Web bestehen vielerlei Beziehungen: Nutzer knüpfen digitale Freundschaften, veröffentlichen Fotos, Kommen-

tare und viele andere Inhalte. Linked Data ist wie dafür geschaffen, diese Beziehungen abzubilden. Im Grunde genommen erzeugt jeder Klick auf einen „Gefällt mir“-Button ein RDF-Tripel. Dessen „Subjekt“ ist der klickende Nutzer, „gefällt“ ist das Prädikat und die dem Nutzer gefallende Seite ist das Objekt (siehe Abbildung 1).

Freunde von Freunden

Die Linked-Data-Prinzipien von Tim Berners-Lee [1] verlangen zunächst, dass „Dinge“ – nicht nur Dokumente – mittels URI identifiziert werden. Auch Personen können dabei durch einen URI identifiziert werden. Sobald dies geschehen ist, können sie als Subjekt oder Objekt in einem RDF-Tripel verwendet werden. Mein URI lautet etwa „<http://me.desone.org/person/aveltens#me>“ und mit folgendem RDF-Tripel sage ich aus, dass ich Tim Berners-Lee kenne (siehe Listing 1).

In dem Beispiel wird die FOAF-Ontologie [2] verwendet, um die Bekanntschafts-

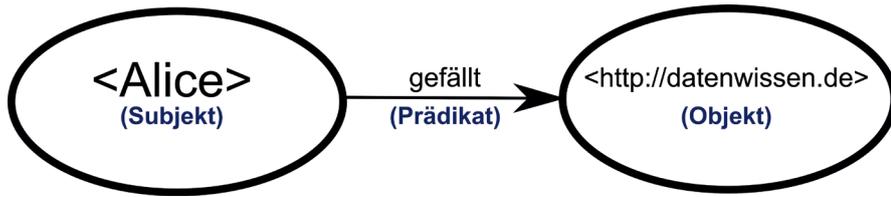


Abbildung 1: „Gefällt mir“-Beziehung als RDF-Graph

```
<http://me.desone.org/person/aveltens#me>
  foaf:knows
    <http://www.w3.org/People/Berners-Lee/card#i>.
```

Listing 1

```
<http://me.desone.org/person/aveltens#me>
  a foaf:Person;
  foaf:familyName "Veltens";
  foaf:givenName "Angelo";
  foaf:homepage <http://datenwissen.de/>;
  foaf:basedNear <http://sws.geonames.org/2945024/>.
```

Listing 2

beziehung auszudrücken. FOAF steht für „Friend of a Friend“ und ist eine weit verbreitete Ontologie, die strukturierte Aussagen über Personen und deren Beziehungen trifft. In Listing 2 ist ein Auszug aus meinem FOAF-Profil in der RDF-Syntax „Turtle“ zu sehen:

In der ersten Zeile steht der URI des Subjekts, über das Aussagen getroffen werden. Es folgen Prädikat-Objekt-Paare, durch Semikolon getrennt. Aus den Daten geht hervor, dass es sich um eine Person handelt („a foaf:Person“). Auch Vor- und Nachname sind im Profil hinterlegt. Da es sich bei der Homepage um eine Web-Ressource handelt, wird sie über ihren URI verlinkt und nicht wie die Namen als String-Literal hinterlegt. Die große Stärke von Linked Data liegt auch hier in der Verlinkung von beliebigen Ressourcen im Web of Data. „foaf:basedNear“ verweist daher auf den URI der Stadt Braunschweig beim Dienst „geonames.org“. Folgt man diesem Link, erhält man weitere Informationen zur Stadt, beispielsweise deren Einwohnerzahl.

Natürlich ist man in einem Profil nicht auf FOAF beschränkt, sondern kann alle im Web verfügbaren Ontologien nutzen und bei Bedarf eine eigene entwickeln. Für die Bedürfnisse von Anwendungen im Social Web ist in Ergänzung zu FOAF die SIOC-Ontologie [3] geeignet. Ruft man „http://

me.desone.org/person/aveltens#me“ im Browser auf, erhält man eine menschenlesbare HTML-Seite. Verlangt man im HTTP-Accept-Header jedoch „application/rdf+xml“, antwortet der Server mit maschinenlesbarem RDF/XML. Das RDF-Dokument hat auch einen eigenen URI und lässt sich über „http://me.desone.org/person/aveltens.rdf“ direkt aufrufen.

Im Web of Data werden Dinge von Dokumenten unterschieden. Eine Person ist etwas anderes als deren Profilseite im Web. Daher benötigen beide einen eigenen URI. Während ich als Person durch „http://me.desone.org/person/aveltens#me“ (Man beachte den Fragment-Identifizier „#me“, der Teil des URI ist) identifiziert werde, lautet der generische URI meiner Profilseite „http://me.desone.org/person/aveltens“

Abbildung 2: Dank Content-Negotiation kann unter dem Profil-URI sowohl ein menschen- (HTML, links) als auch maschinenlesbares Profil (RDF/XML, rechts) ausgeliefert werden

(ohne „#me“). Letzterer identifiziert ein Dokument, der erste eine Person (siehe Abbildung 2). Über „http://me.desone.org/person/aveltens.rdf“ und „http://me.desone.org/person/aveltens.html“ werden zudem zwei verschiedene Repräsentationen des generischen Dokuments identifiziert (RDF/XML und HTML).

Wie man sieht, ist es leicht, ein Social-Network-Profil mithilfe von Linked Data zu erzeugen und mit anderen Profilen und Web-Ressourcen zu verlinken. Aber das allein reicht noch nicht, um soziale Interaktionen im WWW zu ermöglichen. Mit bewährten, asynchronen Verschlüsselungsmechanismen und dem WebID-Protokoll eignet sich ein FOAF-Profil auch zu Authentisierung im Web.

Das WebID-Protokoll

Authentifizierung im Social Web bedeutet, nachweisen zu können, dass einem Nutzer ein bestimmtes Profil gehört. Bei zentral organisierten Social-Network-Plattformen kann dies durch die Eingabe des entsprechenden Benutzernamens und des zugehörigen Passworts erfolgen. In einem dezentralen Social Web aus verlinkten FOAF-Profilen soll eine Nutzerin nachweisen können, dass das unter einem bestimmten URI abrufbare Profil das ihre ist. Dieser Nachweis soll zudem gegenüber beliebigen Diensten im Web möglich sein.

Das WebID-Protokoll erreicht dies durch geschickte Kombination bewährter Web-Standards. Neben einem persönlichen URI und einem darunter verfügbaren FOAF-Profil benötigt ein Nutzer zusätzlich ein X.509-Zertifikat und den dazu gehörigen privaten Schlüssel. Alle heutigen Browser enthalten eine Zertifikatsverwaltung, in der diese bequem hinterlegt werden können. Auch die Authentifizierung mit

solchen Zertifikaten gegenüber einem Web-Server ist keine neue Erfindung, sondern eine seit Langem funktionierende Technologie. Der neue Weg, den WebID einschlägt, ist die Verknüpfung dieser Zertifikate mit FOAF-Profilen.

Zum einen wird im Zertifikat der URI des Nutzers als „Subject Alternative Name“ hinterlegt. Ein Server, der die Authentifizierung durchführt, kann den URI auslesen und die dort hinterlegten Daten abrufen. Um zu beweisen, dass dieses Profil tatsächlich dem Nutzer gehört, fügt dieser neben den gewünschten Profildaten auch seinen öffentlichen Schlüssel dort ein. Die dazu nötigen Prädikate stellt die Cert-Ontologie [4] zur Verfügung (siehe Listing 3).

Der öffentliche Schlüssel ist vom Typ „cert:RSAPublicKey“ und wird über das Prädikat „cert:key“ mit der entsprechenden Person verlinkt. „cert:exponent“ und „cert:modulus“ beinhalten die entsprechenden Daten des Schlüssels. Optional kann der Schlüssel auch über ein „rdfs:label“ benannt werden. Die Zusammenhänge sind in Abbildung 3 schematisch dargestellt.

Authentisiert sich nun eine Nutzerin über ihr Zertifikat an einem Web-Server, kann dieser das Profil abrufen und die digitale Signatur mithilfe des dort hinterlegten öffentlichen Schlüssels prüfen. Das WebID-Protokoll setzt hier also auf bewährte Public-Key-Authentifizierung, um die Zugehörigkeit des FOAF-Profiles zum Nutzer nachzuweisen.

Diese Vorgehensweise ist nicht nur sicher, sondern auch benutzerfreundlich. Ein Klick auf einen Link genügt, um sich gegenüber beliebigen Diensten im Web zu authentisieren. Weder müssen sich Nutzer komplizierte Passwörter merken, noch müssen sie ihren URI eintippen. Denn letzterer ist bereits im Zertifikat enthalten.

Digitale Freundschaften schließen

Nachdem nun Nutzerdaten im FOAF-Profil

```
<http://me.desone.org/person/aveltens#me>
cert:key <#publickey>.

<#publickey>
a cert:RSAPublicKey;
rdfs:label "Angelos Public Key";
cert:exponent 65537;
cert:modulus "ACE0[...]^^<http://www.w3.org/2001/XMLSchema#hexBinary>."
```

Listing 3

bereitstehen und die Authentifizierung über das WebID-Protokoll gewährleistet ist, bleibt noch die Frage offen, wie die Kommunikation zwischen WebID-Nutzern ablaufen kann. Wie kann ich beispielsweise jemandem mitteilen, dass ich ihn oder sie als Freund hinzufügen möchte. Natürlich kann man einfach ein entsprechendes RDF-Tripel im eigenen Profil ergänzen, doch möchte man sein Gegenüber vielleicht über diesen Link informieren, ihr die Gelegenheit geben, die Freundschaft zu bestätigen und einen Backlink zu setzen.

Dazu gibt es einen „Pingback“-Mechanismus, ähnlich wie man ihn schon von Blogs kennt. Verlinkt eine Ressource auf eine andere, kann sie die Information darüber über HTTP-POST an einen bestimmten URI senden. Welcher URI das ist, kann die verlinkte Ressource wiederum selbst bekannt geben. Auch dafür gibt es eine passende Ontologie, das Semantic Pingback Vocabulary [5]. Das folgende Listing gibt über das Prädikat „ping:to“ an, dass sich der Pingback-Dienst von „my-profile.eu“ um eingehende „Pings“ zu meinem Profil kümmert (siehe Listing 4).

Logge ich mich mit meiner WebID bei „my-profile.eu“ ein, bekomme ich eingehende „Pings“ in der Web-Oberfläche angezeigt. Im Artikel „Friending on the Social Web“ haben Henry Story, Andrei Sambra und Sebastian Tramp semantische Pingbacks näher beschrieben [6].

Daten-Silos überwinden

Mit Linked Data und WebID kann es gelingen, die abgeschotteten Daten-Silos des Web 2.0 zu öffnen und ein echtes Social Web zu spinnen. Da sowohl Personen als auch Inhalte über ihre URIs global verlinkbar sind, entfällt die Notwendigkeit eines zentralen Social-Network-Dienstes. Wie auch im dokumentenbasierten WWW Links zu beliebigen anderen Seiten möglich sind, können auf Linked Data basie-

```
<http://me.desone.org/person/aveltens#me>
ping:to <https://my-profile.eu/pingback.php>.
```

Listing 4

rende Social-Web-Anwendungen Links zu Daten anderer Dienste setzen. Abbildung 4 zeigt, wie das Profil der WebID-Nutzerin „Alice“ mit verschiedenen Inhalten im Social Web verlinkt ist. In diesem fiktiven Beispiel hostet Alice ihr FOAF-Profil selbst und hat sich mit ihrer WebID bei einem Bilderdienst eingeloggt. Nach einem Klick auf den dortigen „Gefällt mir“-Button wird der in Alices Profil angegebene Pingback-Dienst benachrichtigt, der daraufhin ein RDF-Tripel im FOAF-Profil ergänzt. Alice hat außerdem im Blog eines Freundes einen Kommentar hinterlassen. Auch dieser wurde mit ihrem Profil verlinkt, ebenso mit dem Blogartikel, auf den er sich bezieht.

Eine Vielzahl von kleinen, auf eine Kernaufgabe spezialisierten Diensten kann sich so zu einem Social Web of Data vernetzen. In der Summe bieten diese Dienste ihren Nutzern weit mehr Möglichkeiten als zentralisierte Social-Network-Plattformen und dennoch entfällt die Notwendigkeit, für diese Vielfalt mehrere Accounts zu führen. Die Herausforderung ist jedoch noch, dass viele Dienste WebID unterstützen müssen und die Handhabung ähnlich einfach werden muss wie die von Facebook & Co.

Semantische Zugriffskontrolle

Ein FOAF-Profil muss keinesfalls komplett öffentlich sein. Über semantische Access Control Lists (ACL) ist es möglich, genau zu definieren, wer welche Informationen sehen darf. Diese ACL funktionieren ähnlich, wie man es bereits von Dateisystemen kennt, nur dass nun Personen und Gruppen über URIs identifiziert werden. Angenommen, Alice wird durch „http://me.example.org/person/alice#me“ identifiziert und stellt unter diesem URI einige unkritische Informationen über sich öffentlich. Gleichzeitig will sie aber engen Freunden auch Zugriff auf ihr Geburtsdatum und ihre Telefonnummer geben. Dazu hinterlegt sie diese Daten im Dokument „http://me.example.org/person/alice/private“ und verlinkt es in ihrem eigentlichen Profil (siehe Listing 5). Man beachte, dass in diesem Beispiel der relative URI „private“

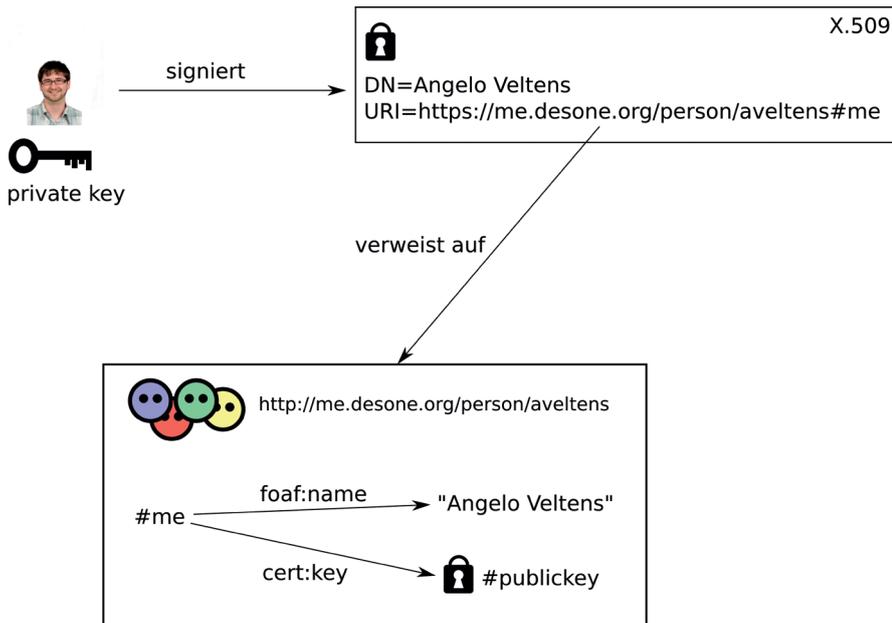


Abbildung 3: Schematische Darstellung der Beziehungen zwischen Nutzer, Zertifikat und FOAF-Profil

genügt, da das Dokument in der Pfadhierarchie direkt unterhalb des öffentlichen Profil-Dokuments liegt. Nun gilt es noch, den Zugriff auf die privaten Informationen zu schützen. Dazu hinterlegt Alice eine Autorisierung in der ACL-Ontologie [7] in ihrem privaten Dokument (siehe Listing 6).

Das Beispiel nutzt nicht mehr die Turtle-Syntax, sondern die ähnliche, aber mächtigere Notation 3. Da die ACL selbst keinen eigenen URI benötigt, ist sie in den ersten fünf Zeilen in eckigen Klammern als sogenannter „blank node“ hinterlegt. Das Prädikat „acl:accessTo“ teilt mit, welche Ressource durch die ACL geschützt wird. Das leere Klammernpaar „<>“ stellt im Grunde wieder einen relativen URI dar, verweist also auf das aktuelle Dokument „http://me.example.org/person/alice/private“. Über „acl:mode acl:Read“ wird ausgesagt, dass es um den Lesezugriff geht.

„acl:agentClass“ schließlich legt fest, wer zugriffsberechtigt ist. Das Prädikat verweist auf einen Typ, dem eine Ressource angehören muss, um Zugriff zu erlangen. Über „acl:agentClass foaf:Person“ könnte man beispielsweise allen Personen Zugriff auf eine Ressource gestatten. Alice will ihre Daten jedoch nur engen Freunden zugänglich machen und gruppiert diese daher in einem eigenen Typ, „#closeFriend“. Ein Web-Server oder eine Web-Anwendung, die das Profil von Alice beherbergt, kann nun bei versuchtem Zugriff auf die private Ressource den Nutzer über das

WebID-Protokoll authentifizieren, die ACL auswerten und anschließend den Zugriff verweigern oder gestatten.

Wer WebID ausprobieren möchte, kann sich leicht bei „http://my-profile.eu“ ein Profil anlegen. Dabei wird ein Zertifikat im Browser erzeugt und installiert. Das zugehörige FOAF-Profil wird bei „my-profile.eu“ gehostet. Anschließend kann man sich mit seiner WebID beispielsweise bei „http://picserv.desone.org“ anmelden und dort ein Bild hochladen. Die nächste Ausgabe wird den praktischen Einsatz von WebID intensiver behandeln und zeigen, wie eine Anwendung Nutzer über WebID authentifizieren kann.

Linked Data und WebID im Unternehmensnetz

WebID kann auch für den Einsatz im Unternehmens-Intranet eine Option sein. Zum einen eignet es sich als Single-Sign-on-

Verfahren (SSO). Selbst wenn Mitarbeiterdaten über mehrere Systeme verteilt sind, können diese als Grundlage für ein unternehmensweites SSO dienen. Die Systeme müssen dazu lediglich die Daten, wie oben beschrieben, als RDF bereitstellen. Ein zentraler Authentifizierungsdienst mit Kenntnis aller Legitimationsdaten ist jedoch nicht notwendig.

Insbesondere, wenn konsequent auf Linked Data gesetzt wird, können Unternehmen darüber hinaus auch von verbessertem Wissensmanagement profitieren. Mit Linked Data und semantischen Technologien lässt sich Wissen organisieren und besser auffinden. Oft steht ein solches Wissensmanagement jedoch vor dem Problem, dass Informationen nicht in den geeigneten Formaten vorliegen.

Identifiziert man Mitarbeiter über WebID, können RDF-Tripel als Nebenprodukt der täglichen Arbeit entstehen. Angenommen, eine Mitarbeiterin liest im Unternehmens-Wiki die Dokumentation über ein Web-Framework. Sie bekommt dort einen Button „kenne mich aus“ zur Verfügung gestellt. Ein Klick darauf verlinkt ihre WebID entsprechend mit dem Web-Framework. Projekte, die gerade planen, das Framework einzusetzen, sehen diese Beziehung und können das Fachwissen der Mitarbeiterin bei Bedarf zu Rate ziehen. Die dazu nötigen Kontaktdaten sind im FOAF-Profil ersichtlich.



Abbildung 4: Verlinkung des WebID-Profiles von Alice mit Inhalten im Social Web

```
# In http://me.example.org/person/alice/private
<http://me.example.org/person/alice#me>
foaf:birthDay "11-30";
foaf:phone <tel:+49-123-456-789>.
```

```
# In http://me.example.org/person/alice
<http://me.example.org/person/alice#me>
foaf:givenName "Alice";
rdfs:seeAlso <private>.
```

Listing 5

```
[
acl:accessTo <>;
acl:mode acl:Read;
acl:agentClass <#closeFriend>
].

<#closeFriend> is rdf:type of
<http://example.net/profile/bob#i>,
<http://trudy.example/foaf#i>.
```

Listing 6

Fazit

Während Linked Data das Web aus Dokumenten zu einem Web aus Daten erweitert,

bringt WebID die soziale Komponente in dieses Web. Nicht nur Dinge werden durch URIs identifiziert, sondern auch Personen. Durch Links lassen sich die URIs dieser Personen mit den Dingen im Web in eine Beziehung setzen. Die sozialen Beziehungen der Nutzer untereinander lassen sich ebenso abbilden wie die Verbindungen zu den Inhalten, die sie im Web erzeugen und teilen. Die Möglichkeiten, die uns heute auf Social-Network-Plattformen geboten werden, lassen sich so auf das gesamte offene Web ausweiten. Es entsteht eine direkte Interaktion zwischen den Profilen der Nutzer, die sie bei Bedarf selbst hosten können, und der Vielzahl an Diensten im Web. Die Notwendigkeit für eine zentrale Plattform entfällt ebenso wie das ständige Anlegen von Accounts bei neuen Diensten.

Referenzen

- [1] <http://www.w3.org/DesignIssues/LinkedData.html>
- [2] <http://xmlns.com/foaf/spec>
- [3] <http://sioc-project.org>
- [4] <http://www.w3.org/ns/auth/cert#>

- [5] <http://purl.org/net/pingback>
- [6] <http://bbfish.net/tmp/2011/05/09>
- [7] <http://www.w3.org/ns/auth/acl>

Angelo Veltens
angelo.veltens@online.de
<http://datenwissen.de>



Angelo Veltens studierte Angewandte Informatik an der Dualen Hochschule Baden-Württemberg Karlsruhe und befasste sich in Studienarbeiten und Abschlussarbeit mit Linked Data und semantischem Wissensmanagement. Heute ist er als Software-Entwickler in Braunschweig tätig, arbeitet in seiner Freizeit am „Social Web of Data“ und referiert auf Konferenzen zu diesem Thema.

Berliner Expertenseminare

- Wissensvertiefung für Oracle-Anwender
- Mit ausgewählten Schulungspartnern
- Von Experten für Experten

10./11. September 2013

Buchen Sie 2 Tage englischsprachige Intensiv-Schulung zum Thema JavaServer Faces (JSF 2.2), dem Standard Web-Applikations-Framework für Java EE.
 mit **Ed Burns**





www.ijug.eu

**JETZT
ABO
BESTELLEN**

Sichern Sie sich 4 Ausgaben für 18 EUR

Für Oracle-Anwender und Interessierte gibt es das Java aktuell Abonnement auch mit zusätzlich sechs Ausgaben im Jahr der Fachzeitschrift *DOAG News* und vier Ausgaben im Jahr *Business News* zusammen für 70 EUR. Weitere Informationen unter www.doag.org/shop/

FAXEN SIE DAS AUSGEFÜLLTE FORMULAR AN

0700 11 36 24 39

ODER BESTELLEN SIE ONLINE

go.ijug.eu/go/abo



Interessenverbund der Java User Groups e.V.
Tempelhofer Weg 64
12347 Berlin

Java aktuell

+++ AUSFÜLLEN +++ AUSSCHNEIDEN +++ ABSCHICKEN +++ AUSFÜLLEN +++ AUSSCHNEIDEN +++ ABSCHICKEN +++ AUSFÜLLEN

Ja, ich bestelle das Abo Java aktuell – das IJUG-Magazin: 4 Ausgaben zu 18 EUR/Jahr

Ja, ich bestelle den kostenfreien Newsletter: Java aktuell – der iJUG-Newsletter

ANSCHRIFT

Name, Vorname

Firma

Abteilung

Straße, Hausnummer

PLZ, Ort

GGF. ABWEICHENDE RECHNUNGSANSCHRIFT

Straße, Hausnummer

PLZ, Ort

E-Mail

Telefonnummer



Die allgemeinen Geschäftsbedingungen* erkenne ich an, Datum, Unterschrift

*Allgemeine Geschäftsbedingungen:

Zum Preis von 18 Euro (inkl. MwSt.) pro Kalenderjahr erhalten Sie vier Ausgaben der Zeitschrift "Java aktuell - das iJUG-Magazin" direkt nach Erscheinen per Post zugeschickt. Die Abonnementgebühr wird jeweils im Januar für ein Jahr fällig. Sie erhalten eine entsprechende Rechnung. Abonnementverträge, die während eines Jahres beginnen, werden mit 4,90 Euro (inkl. MwSt.) je volles Quartal berechnet. Das Abonnement verlängert sich automatisch um ein weiteres Jahr, wenn es nicht bis zum 31. Oktober eines Jahres schriftlich gekündigt wird. Die Wiederrufsfrist beträgt 14 Tage ab Vertragserklärung in Textform ohne Angabe von Gründen.

