Zentrale und dezentrale Informationsplattformen

Hans-Gert Gräbe

Version vom 28. Mai 2015

1 Vorbemerkungen

1.1 Plattformen, Datenmodell und Datenstrukturen

Für die Darstellung einer größeren Menge gut durchsuchbarer Daten muss im allgemeinen Fall ein *Datenmodell* entwickelt und dieses auf entsprechende *Datenstrukturen* (etwa ein Datenbankschema) heruntergebrochen werden. Für die öffentliche Zugänglichkeit und den Austausch der Daten (oder eines als öffentlich zu definierenden Teils) über entsprechende Schnittstellen (API) sind neben diesen Schnittstellen auch die Austauschformate für die Daten als *Bindings* zu definieren. Hier kommen heute im Webbereich üblicherweise selbst definierte Formate auf der Basis von XML oder JSON zum Einsatz.

1.2 Zentrale Informationsplattformen

Die meisten Datensilos sind zentral konzipierte Anwendungen, die Nutzer über verschiedene Nutzerszenarien dazu animieren, ihre Daten im Silo abzugeben. Konzeptionelles Ziel solcher Plattformen ist es, dass sich aus einer solchen Zusammenführung großer Datenmengen einer Vielzahl von Nutzern Mehrwertdienste organisieren lassen, die den Nutzeraufwand des Einstellens eigener Daten überwiegen und auf diese Weise eine selbsttragende Dynamik der Plattform zustandekommt. Große Plattformen wie Facebook, Amazon usw. zeigen, dass ein solches Konzept aufgehen kann, allerdings erinnern entsprechende Erwartungen neuer Startups auf diesem Gebiet oft an Geschichten „vom Tellerwäscher zum Millionär“. Der Weg in die digitale Gesellschaft ist mit Datenfriedhöfen gepflastert ....

Schauen wir uns mit `http://www.edutags.de` ein solches Beispiel genauer an. Das Grundkonzept der Plattform sieht vor, dass Nutzer hier Metainformationen (Tags) zu Lehrmaterialien (Edu) ablegen können.

Erläuterungen weiter ausbauen.

1.3 Entwicklung und Betrieb zentraler Informationsplattformen

Zur Entwicklung der Datenschicht einer solchen Applikation sind die Spezifikation des Datenmodells und der Datenstrukturen unerlässlich. Beides ist im Rahmen des jeweiligen Entwicklerteams möglich. Die Spezifikation des Datenmodells beschreibt zugleich die Restriktionen,

mit denen die Nutzer konfrontiert sind – es kann nur solche Information und in solchen Strukturen gesammelt werden, wie es die Entwickler vorgesehen haben.

Änderungen und Erweiterungen der Spezifikation müssen vom Betreiber gesammelt und entsprechend als neue Features umgesetzt werden. Dies setzt eine intensive Weiterentwicklung der Plattform und damit ein klares und finanziertes Betreiberkonzept voraus, was im Fall von `http://www.edutags.de` durch die Einbindung in den *Deutschen Bildungsserver* als Gemeinschaftsservice von Bund und Ländern sowie (in der technischen Umsetzung) an das *Deutsche Institut für Internationale Pädagogische Forschung* (DIPF) `http://www.dipf.de` als rechtsfähige Stiftung des Öffentlichen Rechts und Institut der Leibniz-Gemeinschaft gewährleistet ist, wie ein Blick in die entsprechenden Impressumsseiten zeigt.

Im Gegensatz hierzu steht die Herausgabe der Daten über entsprechende maschinenlesbare Schnittstellen meist nicht im Fokus der Betreiber einer solchen zentralen Plattform. Entsprechende Such- und Filterfunktionen sind wichtig für die Attraktivität der Plattform auf der Ebene der direkten Nutzerinteraktionen im Sinne des Web 1.0, die maschinenlesbare Auslieferung entsprechender Daten wird aber – wenigstens noch heute – als eher gefährlich für das Betreiberkonzept angesehen. Ein derartiger Zugang hat entwicklungstechnisch denselben Vorteil wie oben bereits beschrieben – durch Einsatz eines angemessenen internen Entwicklungspotenzials können derartige Such- und Filterfunktionen auf dem wachsenden Datenbestand nutzerfreundlich entwickelt und weiterentwickelt werden.

Diese Bestandsaufnahme legt nahe, dass es für derartige zentrale Informationsplattformen eine kritische Masse an potenziellen und wirklichen Nutzern geben muss, ohne die ein nachhaltiger Betrieb einer solchen Plattform und insbesondere die Allokation der für den initialen Aufbau erforderlichen Mittel nicht möglich ist.

1.4 Lokale Informationsplattformen

Diese Voraussetzungen sind für weitgehend ehrenamtlich betriebene lokale Informationsplattformen mit sehr engem Budget in der Regel nicht gegeben. In den meisten Fällen sind die entsprechenden Akteure und Multiplikatoren nahe an realen Entwicklungsprozessen im jeweiligen Gebiet, die Informationsplattform hat hierbei vor allem unterstützenden Charakter.

Oft wird eine solche Plattform über entsprechende Projektgelder von einem lokalen IT-Partner in einem überschaubaren Zeitraum entworfen und aufgesetzt und danach von den Multiplikatoren mit deutlich geringerem Personaleinsatz – oft durch technisch versierte Ehrenamtler – betrieben.

In vielen Fällen fehlt eine klare Dokumentation der Plattform bzw. haben sich die Datenstrukturierungsprinzipien über die Zeit weiterentwickelt und einen „Zoo“ von Strutkuren hinterlassen, der sich mit den verfügbaren geringen Ressourcen nur schwer einfangen lässt. Eine Analyse des Quellcodes und vor allem der entsprechenden Datenbankschemata lässt eine vorsichtige Rekonstruktion des Datenmodells zu, sofern ein solches in expliziter Form in der Entwicklung seitens des lokalen IT-Partners überhaupt eine Rolle gespielt hat.

1.5 Datenaustausch zwischen lokalen Informationsplattformen

Im Gegensatz zum Singleton-Charakter zentralen Informationsplattformen haben wir es bei lokalen Informationsplattformen mit einem regionalen Ökosystem von Websites zu tun, die

zunächst als Schaufenster der jeweiligen Betreiber entworfen wurden und deren technisch gestützte Kommunikation untereinander als neue Herausforderung zu realisieren ist. Im Sinne des Web 2.0 Paradigmas geht es dabei darum, das regionale Ökosystem von Websites als Ansammlung verlinkter, aus je lokalen Datenbankinhalten generierter Webseiten weiterzuentwickeln hin zu einem System von Websites, das zusätzlich eine gemeinsame verteilte Datenbank öffentlich zugänglicher Inhalte pflegt und verwendet.

Hierzu müssen

- entsprechende öffentlich interessante Inhalte der einzelnen Knoten des regionalen Website-Ökosystems indentifiziert werden,
- diese Inhalte lokal in einem gemeinsamen Austauschformat für interessierte Akteure zur Verfügung gestellt werden sowie
- relevante Inhalte von jenen Akteuren abgerufen, verarbeitet und die angereicherten Inhalte auf ähnliche Weise verfügbar gemacht werden.

Für eine solche Vernetzung ist eine gewisse Homogenisierung auf der Ebene der (oftmals nicht explizit spezifizierten) Datenmodelle der einzelnen Betreiber erforderlich. Eine solche Homogenisierung sollte nicht auf der (konzeptionellen) Ebene der Datenmodelle stehenbleiben, sondern Eingang in die jeweils konkret verwendeten Datenstrukturen finden, um den Datenaustausch zu vereinfachen. Eine solche Homogenisierung ist ein diffiziler zunächst sozialer und später auch technisch umzusetzender *längerer Prozess*.

Für eine solche Vernetzung auf Datenebene sollen möglichst standardisierte Konzepte zum Einsatz kommen. Das zentrale Framework wird durch die Konzepte der *Linked Open Data Cloud* (LOD) `http://lod-cloud.net` vorgegeben, in der mit RDF als generalem Datenstrukturierungskonzept und HTTP als Web-Protokoll die technische Kommunikation auf der Basis weit verbreiteter Standards organisiert wird.

2 Linked Open Data und RDF

2.1 Grundlagen

RDF steht für *Resource Description Framework*, ein Konzept zum Aufschreiben von Geschichten über „die Welt“ als Mengen von Drei-Wort-Sätzen.

```
<Subjekt> <Prädikat> <Objekt> .
```

Ein zentrales Konzept ist dabei der *Unique Resource Identifier* (URI) als textuelle Repräsentation realweltlicher Objekte (und Subjekte) im digitalen Universum. Er spielt eine ähnliche Rolle wie der Primärschlüssel eines Datensatzes in einer relationalen Datenbank, wobei allerdings die Unterscheidung zwischen Primär- und Fremdschlüsseln verloren geht, da ein RDF-Satz dem einzelnen Eintrag

```
<Zeilennummer> <Spaltennummer> <Wert> .
```

in einer relationalen Tabelle entspricht und die Rekonstruktion eines Datensatzes bereits der Zusammenführung aller RDF-Sätze mit demselben Subjekt `<Zeilennummer>` entspricht.

Ein solcher Identifier ist in der Regel strukturiert aufgebaut aus einem *Namenspräfix* und einem Suffix, wobei verschiedene Namenspräfixe die Überlappungsfreiheit der URIs aus verschiedenen Kontexten garantieren.

Weitere wichtige Designelemente von RDF:

- Subjekt und Prädikat müssen URIs sein, als Objekt kann eine URI oder ein Literal (Typ rdf:Literal) stehen.
- Literale können Typ- und Sprachmarkierungen tragen.
- Es gibt verschiedene Notationen für dieselbe Menge von RDF-Sätzen (Turtle, rdf/xml, json, ntriples) und Werkzeuge, diese Notationen ineinander umzuwandeln.

Zusammenfassung der zentralen Konzepte

1. *Zentrale Idee:* Speichere textuelle Beschreibungen auf uniforme Weise als Tripel und verwende Standardkonzepte und -werkzeuge zur Verwaltung dieser Daten.
2. *Ressourcen:* Textuelle Repräsentation im digitalen Universum als URI, HTTP als Protokoll, mit dem auf solche weltweit verteilten URIs auf einheitliche Weise zugegriffen werden kann.
3. *Resource Descriptions:* Gib auf Anfrage an die HTTP-Adresse ein nützliches Stück Information im RDF-Format zurück, das mit anderen solchen Informationseinheiten zu neuen RDF-Sätzen kombiniert werden kann.
4. Betreibe *RDF Triple Stores* als Teil einer weltweiten verteilten Datenspeicher-Infrastruktur, etwa `http://leipzig-data.de/Data`.
5. (Verteilte) Anfragesprache SPARQL: Stelle *SPARQL Endpunkte* auf RDF Triple Stores bereit, etwa `http://leipzig-data.de:8890/sparql`.

Wichtig ist die Umsetzung der Punkte 1.–3., also

- die Entwicklung einer Strategie der Vergabe von URIs, die im Rahmen des eigenen Webservers als URL aufgelöst werden können,
- die Ausgabe von nützlicher Information über die durch die URI gekennzeichnete Ressource im RDF-Format bei einer HTTP-Anfrage (hierzu kann durch URI-Rewriting ein größerer Bereich von Adressen auf eine überschaubare Anzahl von RDF-Dateien weitergeleitet werden)

sowie die Diskussion der Ergebnisse im Kreis interessierter Betreiber von Websites.