

Vom RESTful Webservice für Pflanzenschutzmittelregistrierungsdaten zur anwendungsunabhängigen Ontologie

Öffentliche Daten auf die nächste Stufe heben

Katharina Albrecht ¹, Kristoffer Janis Schneider¹ und Daniel Martini¹

Abstract: Bei der Ausbringung von Pflanzenschutzmitteln müssen Abstandsauflagen eingehalten werden. Allerdings müssen diese händisch ermittelt und berücksichtigt werden. Um den Landwirt dabei zu unterstützen, wurde im PAM-Projekt ein Entscheidungshilfesystem entwickelt. Ein Teil dieses Systems ist der Abstandsauflagendienst, den das KTBL bereitstellt. Für diesen Dienst wurde ein Prozess entwickelt, welcher die Pflanzenschutzmittelregistrierungsdaten aus einer REST-Schnittstelle ausliest, die ausgelesenen Informationen mit Hilfe eines RML-Mappings in eine Ontologie überführt und diese über einen SPARQL-Endpoint anwendungsunabhängig abrufbar macht.

Keywords: Linked Open Data, Ontologie, Semantic Web, RML, SPARQL

1 Einleitung

Die maschinelle Ausbringung von Pflanzenschutzmitteln (PSM) ist Stand der Technik; Schläge werden GPS-geführt befahren, mit Applikationskarten können Abstände zu Saumstrukturen, Oberflächengewässern oder urbanen Strukturen definiert werden und über die Teilbreitenabschaltung können die PSM gezielt appliziert werden. Dies geschieht jedoch nicht automatisch. Noch immer obliegt es dem Landwirt, diese einzuhaltenden Abstände manuell aus den Auflagentexten in der vom Bundesamt für Verbraucherschutz und Lebensmittelsicherheit (BVL) bereitgestellten Zulassungsdatenbank, im Folgenden PSM-DB genannt, zu ermitteln und sie entweder in die Applikationskarte einzupflegen oder während der Ausbringung einzuhalten. Des Weiteren müssen Aktualisierungen regelmäßig überprüft und vorgenommen werden. Dieses Vorgehen ist nicht nur zeitintensiv, sondern auch fehleranfällig.

Um den Landwirt dabei zu unterstützen, wurde in den PAM-Projekten ein Entscheidungshilfesystem [Sc16] entwickelt. Während der Projektlaufzeiten ist am KTBL ein Prozess entstanden, welcher die Auflagendaten mit Hilfe von Linked Open Data

¹ Kuratorium für Technik und Bauwesen in der Landwirtschaft e.V., Digitale Technologien, Bartmingstraße 49, 64289 Darmstadt, k.albrecht@ktbl.de,  <https://orcid.org/0000-0002-3400-8102>, k.schneider@ktbl.de, d.martini@ktbl.de

(LOD) maschinenlesbar und anwendungsunabhängig bereitstellt. Dabei werden die Zulassungsdaten mit Hilfe eines deklarativen Mappings aus der vorhandenen Datenbank in als Resource Description Framework (RDF) [BG04] triples repräsentierte Aussagen überführt. Die triples wiederum werden in einem Triplestore bereitgestellt und mit Hilfe von SPARQL Queries abgefragt. Das BVL hat im Jahr 2020 an der Initiative tech4germany² teilgenommen und stellt seitdem die PSM-DB über eine RESTful Schnittstelle bereit. So ist es möglich, mit geeigneten technischen Werkzeugen auf die verschiedenen Tabellen zuzugreifen. Dies ist ein bedeutender Schritt hinsichtlich der Maschinenlesbarkeit der Zulassungsdaten.

2 Stand des Wissens

In den vergangenen PAM-Projekten wurde der Abstandsauflagendienst entwickelt und fortwährend an neue Entwicklungen angepasst und verbessert. [Ma15] stellt die initiale Entwicklung eines Mappingprozesses dar und [Ma18] beschreibt den PSM Mappingprozess, der während des ersten PAM-Projektes entstanden ist. Im Folgeprojekt PAMrobust wurde besonderer Wert auf die Nutzung von Standards und auf die Reduktion der Medienbrüche gelegt. Dieser Mappingprozess ist in [AMS19] zusammengefasst. Die Datenstruktur und inhaltliche Darstellung wurden in PAM3D verfeinert und bilden somit die Basis für die nachfolgend beschriebenen Arbeiten, welche während der Laufzeit von PAM-M durchgeführt wurden.

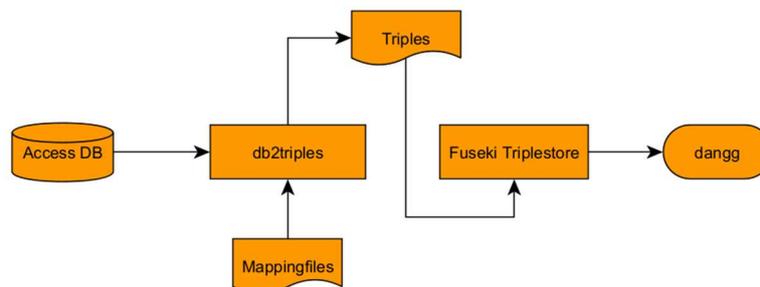


Abb. 1: PSM-Daten liegen in einer Access DB vor, werden mit db2triples gemappt und über den Fuseki Triplestore via Webdienst dangg bereitgestellt

In Abb. 1 **Fehler! Verweisquelle konnte nicht gefunden werden.** ist der bisherige Mappingprozess dargestellt. Zunächst bildete eine relationale Datenbank die Datengrundlage. Mit Hilfe des Mapping-Tools db2triples, welches R2RML [DSC12] implementiert, wurden die relationalen Daten in RDF triples übersetzt. Diese triples wurden über den Apache Jena Fuseki³ Triplestore publiziert, sodass über SPARQL-

² <https://tech.4germany.org/project/psm/>

³ <https://jena.apache.org/documentation/fuseki2/>

Abfragen [HS13] auf die Daten zugegriffen werden konnte. Zusätzlich wurden die Informationen über einen Webservice des KTBL veröffentlicht.

Bisher bestand hinsichtlich der Aktualität der bereitgestellten Ontologie eine starke Abhängigkeit gegenüber der Access DB, welche manuell bereitgestellt, gespeichert und verarbeitet wurde. Dank der Bereitstellung der PSM-DB als REST-Service ist diese Abhängigkeit nicht mehr gegeben. Die RESTful-Bereitstellung ermöglicht zukünftig ein vollständig automatisiertes Auslesen der Daten. Für den PAM-Prozess bedeutete dies zunächst eine Überarbeitung des bestehenden Vorgehens. Es musste ein neuer Prozess zur Abfrage der REST API implementiert werden, und das darauf aufbauende Mapping sowie die Bereitstellung der Daten mussten den neuen Ansprüchen genügen. Zudem wurde das Quellcode-Repository für db2triples auf GitHub geschlossen und das Werkzeug offenbar nicht mehr weiterentwickelt oder aktualisiert. Daher musste für das Mapping eine Alternative gefunden und eingesetzt werden.

3 Der PAM-Prozess

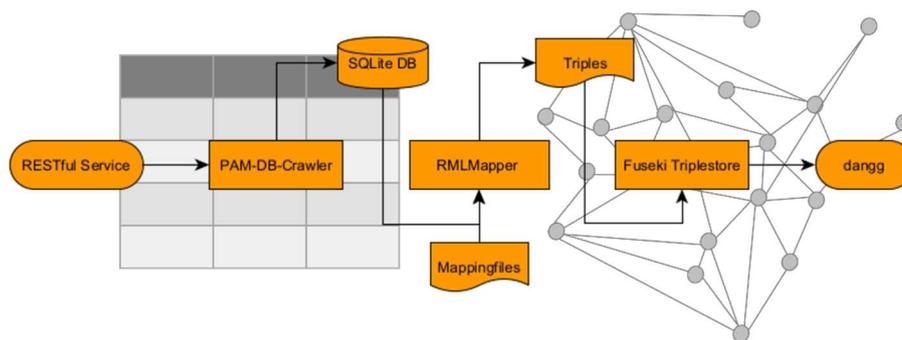


Abb. 2: PSM-Daten werden als RESTful-Service bereitgestellt, mit dem PAM-DB-Crawler abgefragt und in einer SQLite DB gespeichert. Das Mapping wird mit dem RMLMapper prozessiert und die Bereitstellung erfolgt über einen Fuseki Triplestore und via Webservice dangg

Abb. 2 zeigt den weiterentwickelten PAM-Prozess. Die PSM-DB, welche über den RESTful Webservice bereitgestellt wird, wird vom PAM-DB-Crawler⁴ abgerufen und in eine SQLite-Datenbank gespeichert. Dies wird in Abschnitt 3.1 erläutert. Im nachfolgenden Abschnitt 3.2 wird das Mapping erläutert. Dabei werden die Informationen aus der SQLite-DB mit Hilfe von Mapping-Dateien und des Mapping tools RMLMapper⁵ von rml.io in RDF triples übertragen. In Abschnitt 3.3 folgt dann die Erläuterung der Bereitstellung per Fuseki Triplestore.

⁴ <https://github.com/KTBL/PAM-DB-Crawler>

⁵ <https://github.com/RMLio/rmlmapper-java>

3.1 PAM-DB-Crawler

Die Inhalte der PSM-DB werden vom BVL bereitgestellt und können dort via REST-API abgefragt werden. Der REST-Service⁶ liefert die Daten im JSON-Datenformat zurück. Jede Tabelle aus der PSM-DB wird als eigene Schnittstelle bereitgestellt und kann per HTTP-GET abgefragt werden. Da die meisten Schnittstellen sehr große Datenmengen zurückliefern (deutlich über 100 Datenelemente), wird ein Paging-Mechanismus genutzt. Um weitere Elemente abzufragen, kann der „offset“ Parameter URL-kodiert mitgegeben werden. Bei Bedarf kann die Page-Größe via des „limit“ Parameter verändert werden.

Um das bereits entwickelte Mapping weiter nutzen zu können, werden die PSM-Daten in einer relationalen Datenbank benötigt. Mithilfe des REST-Service bietet sich die Möglichkeit, alle Daten abzufragen und in einer lokalen Datenbank zu speichern. Ein Weg, dies umzusetzen, wäre die Datenmodelldefinitionen zu nutzen, um händisch passende Datenstrukturen zu erstellen, diese mittels API-Abfragen zu befüllen und die abgefragten Daten in einer SQLite-Datenbank zu speichern. Bei diesem Vorgehen ist jedoch viel Boilerplate-Code notwendig. Zur Reduktion des Boilerplate-Codes bietet es sich an, einen Generator im Sinne der Meta-Programmierung zu verwenden. Ein solcher Generator erzeugt auf Basis von Definitionsinformationen den Code automatisch.

Ein großer Vorteil dieses Vorgehens ist, dass Änderungen am Datenmodell oder an der Adresse der Schnittstellen schnell eingepflegt und auf verschiedenen Wegen angepasst werden können.

3.2 PSM-Mapping

Die nun in einer SQLite-DB vorliegenden PSM-Daten müssen mit Hilfe eines Mappingverfahrens aus der relationalen Darstellung in RDF triples überführt werden. Dabei reicht es nicht, nur die einzelne Tabelle zu mappen, sondern es müssen auch Beziehungen zwischen Tabellen berücksichtigt werden, wie in nachfolgendem Beispiel zu sehen ist.

AWG_ID	...
034028-60/07-004	...
...	...

AWG_ID	KULTUR	...
034028-60/07-004	RUBID	...
...

Abb. 3: links: ein Auszug aus Tabelle AWG, rechts: ein Auszug aus Tabelle AWG_KULTUR

Abb. 3 zeigt Auszüge aus zwei Tabellen der PSM-DB. Die Tabelle AWG beinhaltet übergreifende Informationen zu den jeweiligen Indikationen. Dabei ist die AWG_ID die ID der Indikation und verbindet somit die jeweiligen Informationen miteinander. Die zweite Tabelle zeigt einen Auszug aus AWG_KULTUR, also jene Kultur, auf die das Pflanzenschutzmittel mit der jeweiligen Indikation angewandt wird. Um aus diesen zwei

⁶ <https://psm-api.bvl.bund.de>

Tabellen eine Aussage, also ein RDF triple zu erstellen, müssen diese gemeinsam gemappt werden. Dafür werden in der Mappingdatei mit Hilfe von Triples Maps die Aussagen definiert. Diese bestehen immer aus einem Subjekt und ein oder mehreren Prädikaten und Objekten. Triples Maps werden dann über sogenannte Join-Conditions verknüpft. Abbildung 4 zeigt die Umsetzung des oben genannten Beispiels.

```
psmr:ind_034028-60-07-004
  rdf:type          psm:Indication ;
  psm:appliedOnCrop psm:crop_RUBID .
```

Abb. 4: Mapping-Ergebnis für die Beispielindikation 034028-60/07-004

3.3 Triplestore und Bereitstellung

Um die generierte Ontologie abrufbar zu machen, müssen die Triple in einen Triplestore geladen werden. Dieser Triplestore wird über einen SPARQL-Endpoint verfügbar gemacht und kann dann mittels der standardisierten Abfragesprache SPARQL abgefragt werden. Zur Publikation der Daten wird der Fuseki-Server genutzt. SPARQL Queries können entweder direkt in der Benutzeroberfläche des Fuseki oder per HTTP-POST abgesetzt werden. Über eine DESCRIBE Query können alle Informationen abgerufen werden, die mit der gewählten Indikation verknüpft sind. Weiterhin können SELECT Queries, welche einfache Abfragen mit tabellarischer Ausgabe ähnlich SQL ermöglichen, oder CONSTRUCT Queries, mit deren Hilfe selektiv Untergraphen erstellt werden, definiert werden. Sofern bei der Implementierung des Endpoints vorgesehen, sind auch UPDATE Queries möglich. Als Rückgabeformat stehen eine Reihe von Formaten zur Verfügung, darunter Turtle, JSON-LD, N-Triples oder XML.

4 Ergebnisse und Diskussion

Der am KTBL entwickelte Abstandsauflagendienst gibt alle Abstandsauflagen, welche für eine bestimmte Indikation berücksichtigt werden müssen, im JSON-LD⁷ Format zurück. Die Bereitstellung der PSM-DB über eine RESTful-Schnittstelle vereinfacht den PAM-Prozess deutlich. Durch die Nutzung des PAM-DB-Crawlers können die Daten flexibel in die SQLite-DB gespeichert und gemappt werden. Der RML-Mapper ist ein adäquater Ersatz für db2triples und bietet Raum für weitere Verbesserungen, z.B. das Einbeziehen weiterer Datenquellen in anderen Datenformaten wie JSON oder CSV. Die Nutzung von Fuseki hat sich ebenfalls bewährt. Insgesamt bietet der PAM-Prozess nun einen zuverlässigen Weg, um insbesondere die Abstandsauflagen der PSM-DB anwendungsunabhängig und maschinenlesbar bereitzustellen.

⁷ <https://json-ld.org/>

5 Ausblick

Die BVL REST-API ermöglicht die vollständige Automatisierung des PAM-Prozesses. Die Entwicklung des PAM-DB-Crawlers ist dabei ein essenzieller Schritt hinsichtlich dieser Automatisierung. Um Redundanzen und Fehlinformationen bei Aktualisierungen zu vermeiden, muss hier jedoch noch ein geeignetes Vorgehen entwickelt werden. Des Weiteren ist es mit dem RML-Mapper möglich, das erstellte Mapping automatisch auf einen SPARQL Endpoint via SPARQL-Update zu publizieren. Hierfür sind noch Überlegungen zum Aufbau einer stabilen Infrastruktur notwendig, um sowohl eine stabile Bereitstellung als auch einen sicheren Betrieb gewährleisten zu können.

Förderhinweis: Die Förderung des Vorhabens FKZ 321-06.01-28-RZ-5-105 erfolgt aus Mitteln der Landwirtschaftlichen Rentenbank. Die Projektträgerschaft erfolgt über die Landwirtschaftliche Rentenbank im Förderprogramm Innovationsförderung.

Literaturverzeichnis

- [AMS19] Albrecht, K.; Martini, D.; Schmitz, M.: Linked Open Data im Pflanzenschutz – Maschinenlesbare Bereitstellung der notwendigen Informationen zu Indikationen und Abstandsaufgaben. In: Meyer-Aurich, A.; Gandorfer, M.; Barta, N.; Gronauer, A.; Kantelhardt, J.; Floto, H. (Hrsg.): 39. GIL-Jahrestagung, Digitalisierung für landwirtschaftliche Betriebe in kleinstrukturierten Regionen – ein Widerspruch in sich? Bonn: Gesellschaft für Informatik e.V. S. 19-24, 2019.
- [DSC12] Das, S.; Sundara S.; Cyganiak, R.: R2RML: RDB to RDF Mapping Language, <https://www.w3.org/TR/r2rml/>, Stand: 12.10.2022.
- [HS13] Harris, S.; Seaborne, A.: SPARQL 1.1 Query Language. World Wide Web Consortium, 2013. <http://www.w3.org/TR/sparql11-query/>, Stand: 12.10.2022.
- [Ma15] Martini, D. et al.: KTBL-Planungsdaten auf dem Weg in die Zukunft – Bereitstellung über Linked Open Data. In: Ruckelshausen, A.; Schwarz, H.-P.; Theuvsen, B. (Hrsg.), Informatik in der Land-, Forst- und Ernährungswirtschaft 2015, Geisenheim. GI-Edition Lecture Notes in Informatics, Bd. 238, Bonn, S. 105-108, 2015.
- [Ma18] Martini, D.: Webservices auf heterogenen Datenbeständen – Methoden der Umsetzung am Beispiel der KTBL-Planungsdaten. In: Ruckelshausen, A.; Meyer-Aurich, A.; Borchard, K.; Hofacker, C.; Loy, J.-P.; Schwerdtfeger, R.; Sundermeier, H.-H. F.; Theuvsen, B. (Hrsg.): 38. GIL-Jahrestagung, Digitale Marktplätze und Plattformen. Bonn: Gesellschaft für Informatik e.V. S. 155-158, 2018.
- [PC13] Prud'hommeaux, E.; Carothers, G.: Turtle – Terse RDF Triple Language. World Wide Web Consortium, 25 February 2014. <http://www.w3.org/TR/turtle/>, Stand: 12.10.2022.
- [Sc16] Scheiber, M. et al.: Pflanzenschutz-Anwendungs-Manager (PAM): Automatisierte Berücksichtigung von Abstandsaufgaben. Praktische Vorführung und Feldtestergebnisse. In: Ruckelshausen, A.; Meyer-Aurich, A.; Rath, T.; Recke, G.; Theuvsen, B. (Hrsg.): Informatik in der Land-, Forst- und Ernährungswirtschaft 2016. Bonn: Gesellschaft für Informatik e.V. S. 177-180, 2016.