

SODA – Die Datenbank als Document Store

Rainer Willems

Oracle Deutschland B.V. & Co. KG

Dreieich

Schlüsselworte

SODA, Simple Oracle Document Access, Document Store, Schemaless, JSON, Collections

Einleitung

Oracle hat mit der Version 12.1.0.2 SODA eingeführt. SODA steht für "Simple Oracle Document Access" und macht die Oracle-Datenbank zum Document Store. Diese erfreuen sich wegen ihrer Einfachheit und Flexibilität großer Beliebtheit bei den Entwicklern, die mit einer schemalosen Entwicklung agil arbeiten können. Man lässt Daten im JSON-Format einfach per REST-Schnittstelle in den Document Store "fallen" und fertig.

Allerdings müssen auch diese Document Stores sich mit Fragestellungen wie Betrieb, Verfügbarkeit und Sicherheit auseinandersetzen und man bekommt ein Datensilo, welches früher oder später doch mit der relationalen Welt integriert sein will. SODA deckt beide Seiten ab. Die einfache, agile und schemalose Entwicklung wird ohne SQL-Kenntnisse ermöglicht und die Daten liegen doch in der gewohnten Oracle-Datenbank. Betriebsthemen sind damit bereits geklärt und die Integration mit der relationalen Welt wird durch die JSON-Funktionalität der Datenbank gewährleistet. Alles was SQL hergibt kann damit auch auf JSON-Dokumenten verwendet werden.

Schemalose Applikations-Entwicklung

Die primäre Eigenschaft von schemaloser Entwicklung liegt darin, dass kein klassisches Datenmodell in einer Datenbank verwendet wird. Die Daten liegen in Dokumenten mit einer Struktur und sind bezüglich der Attribute unabhängig von starren Vorgaben. JSON (JavaScript Notation) ist das Format, welches hier in der Regel verwendet wird. Im folgenden ein Beispiel für solch ein JSON-Dokument.

```
{
  "name": "Willems",
  "vorname": "Rainer",
  "firma": "Oracle",
  "adresse": {
    "strasse": "Robert-Bosch-Strasse",
    "ort": "Dreieich",
    "plz": "63303"
  }
}
```

Kommt nun beispielsweise die Anforderung nach einem neuen Attribut, muss kein Datenmodell angepasst werden, bevor der Entwickler dies nutzen kann, das neue Attribut kommt einfach in das JSON-Dokument. Struktur und Daten liegen eben zusammen.

Oft verwenden Entwickler NoSQL-Datenbanken für solche Dokumente. Diese müssen allerdings am Ende des Tages nicht nur die einfache Abspeicherung und Abfrage solcher Dokumente ermöglichen – was sie zweifelsfrei machen – sondern idealerweise 2 weitere Eigenschaften mit sich bringen: den reibungslosen und möglichst hochverfügbaren Betrieb und die Integration mit anderen – im Zweifel

dann relationalen – Daten. Und genau hier kommt SODA ins Spiel. SODA ermöglicht genau diese schemalose Entwicklung von Anwendungen auf Basis von JSON-Dokumenten in der Oracle-Datenbank. Auch ohne die Kenntnis von SQL können Entwickler “NoSQL-like” arbeiten, können dann aber alle Vorteile der Oracle-Datenbank nutzen.

Der DBA freut sich, da er kein neues Produkt managen muss oder gar Hochverfügbarkeitskonzepte umsetzen muss; das hat er schon alles mit der Oracle-Datenbank.

Der Entwickler freut sich, da er so entwickeln kann, wie er das möchte, aber zusätzliche Funktionen aus der Oracle-Datenbank nutzen kann, die keine NoSQL-Datenbank so liefern kann. Dazu aber später mehr.

NoSQL Document Stores

Der primäre Zugriff auf JSON-Dokumente erfolgt über den Key/Value-Ansatz. Zu einem eindeutigen Schlüssel, existiert ein JSON-Dokument und die JSON-Dokumente müssen nicht zwanghaft die gleiche Struktur haben (was zwar flexibel ist, aber auch Nachteile haben kann).

Die Dokumente liegen in sogenannten Collections, welche Geschäftsobjekte oder Applikationen repräsentieren. Typische Funktionalitäten, die ein NoSQL Document Store nun bietet sind hauptsächlich:

- Erstellen und Löschen von Collections
- Erstellen, Abfragen, Ändern oder Löschen von Dokumenten aufgrund des Schlüssels
- Auflistung der Dokumente (inkl. Filtermöglichkeiten)
- Erstellen und Löschen von Indizes

JSON in der Oracle Datenbank

Mit dem Release 12.1.0.2 hat Oracle die Unterstützung von JSON in die Datenbank implementiert. Hierfür wurde kein neuer Datentyp eingeführt, sondern die JSON-Dokumente werden in Standard-Datentypen (CLOB, BLOB oder auch VARCHAR) abgelegt. SQL wurde nun so erweitert, damit auch mit diesen Objekten gearbeitet werden kann (siehe SQL & JSON unten).

SODA

Ebenfalls mit 12.1.0.2 wurden 2 Implementierungen von SODA eingeführt, die es ermöglichen NoSQL-Like mit diesen JSON-Dokumenten zu arbeiten. Zum einen ist dies *SODA for Java*, ein Interface für Java-Entwickler welches SQL*Net zur Datenbank nutzt. Die hier angebotenen JAVA-Klassen und Methoden ermöglichen die oben aufgeführten Funktionalitäten analog zu einem NoSQL Document Store.

Bei der zweiten Implementierung handelt es sich um *SODA for REST*, eine REST-basierte Schnittstelle, die als Java Servlet in den Oracle Rest Data Services (ORDS) integriert ist und per HTTP verwendbar ist.

Der Hauptunterschied, der bei der Verengung der beiden Implementierungen zu beachten ist, liegt in der Transaktionalität. Während SODA for Java auf einer Standard-Datenbankverbindung basiert und dementsprechend transaktional verwendet werden kann, ist REST (mit der Grundlage HTTP) eben statuslos und jede Operation steht für sich. Dies haben aber REST-based Schnittstellen so an sich.

Folgende Operationen bietet diese Schnittstelle:

List of all Collections	GET	/ords/schema/soda/version
Create Collection	PUT	/ords/schema/soda/version/collection
Add Document	POST	/ords/schema/soda/version/collection
Add Documents	POST	/ords/schema/soda/version/collection?action=insert
Query Collection	GET	/ords/schema/soda/version/collection
Call Document	GET	/ords/schema/soda/version/collection/id
Change Document	PUT	/ords/schema/soda/version/collection/id
Remove Document	DELETE	/ords/schema/soda/version/collection/id
Remove Documents	POST	/ords/schema/soda/version/collection?action=delete
Remove all Documents	POST	/ords/schema/soda/version/collection?action=truncate
Query by Example	POST	/ords/schema/soda/version/collection?action=query
Create/Drop Index	POST	/ords/schema/soda/version/collection?action=unindex
Remove Collection	DELETE	/ords/schema/soda/version/collection

Abb. 1: REST-based Interface

Diese Pfade der sogenannten SODA Collection API können wie folgt interpretiert werden:

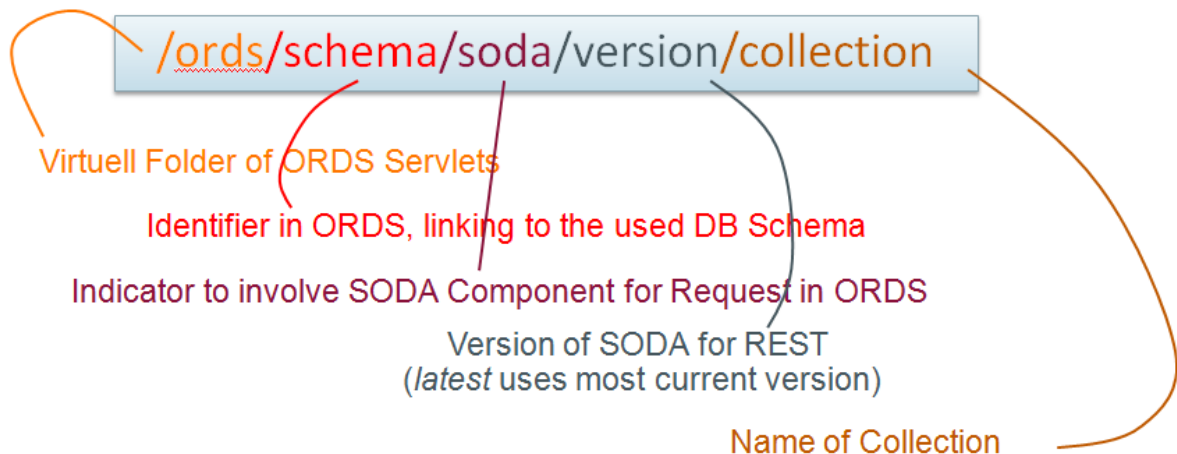


Abb. 2: Pfade der SODA Collection API für REST

Die Abfrage der bestehenden Collections sieht also wie folgt aus (bei gegebener

GET-Request: `http://<host>:<port>/ords/<schema>/soda/latest`

Eine neue Collection wird angelegt mit

PUT-Request: `http://<host>:<port>/ords/<schema>/soda/latest/mycollection`

und eine neues Dokument dort eingefügt per

POST-Request: <http://<host>:<port>/ords/<schema>/soda/latest/mycollection>

unter der Mitgabe des JSON-Dokumentes.

Die Architektur innerhabl von ORDS sieht wie folgt aus:

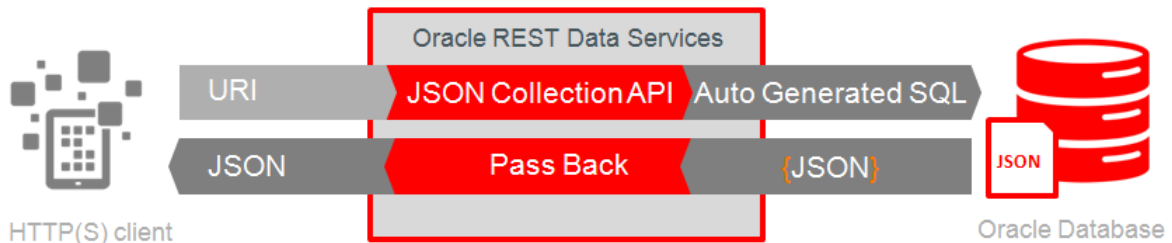


Abb. 3: REST-based Interface

In der Oracle-Datenbank wird letztendlich eine Tabelle mit dem Namen der Collection (Achtung, Groß/Kleinschreibung ist relevant), angelegt, welche ein BLOB Spalte für die Aufnahme der JSON-Dokumente beinhaltet. Dies ist aber vollkommen transparent aus Sicht der REST-Schnittstelle.

SQL & JSON

Ein riesiger Vorteil der Verwendung der Oracle Datenbank besteht – neben den ausgereiften und etablierten Betriebsprozessen im Rechenzentrum – in der möglichen Verknüpfung der JSON-Dokumenten mit den relationalen Inhalten. Ist bspw. obiges JSON-Dokument in der Tabelle *demo* in der Spalte *json_document* abgelegt (was bei der Verwendung der Collection *demo* der Fall wäre), könnte man dieses mit folgendem SQL auslesen.

```
select d.json_document.vorname,
       d.json_document.name,
       d.json_document.adresse.ort,
       d.json_document.firma
from "demo" d;
```

Dies ist die Kurznotation, es gibt auch eine erweiterte Variante, die insbesondere bei Listen in Dokumenten hilfreich ist:

```
select vorname, name, ort, firma
from "demo", json_table(json_document,
                        '$'
                        columns (vorname varchar2(30) path '$.vorname[0]',
                                name varchar2(30) path '$.name[0]',
                                ort varchar2(30) path '$.adresse.ort[0]',
                                firma varchar2(30) path '$.firma[0]'
                        )
)
```

Natürlich kann auch in beiden Notationen gefiltert werden und viele Möglichkeiten der Datenbank können direkt auf den JSON-Dokumenten angewendet werden. Hier seien in erster Linie Joins genannt. Die Verknüpfung der relationalen Daten mit den JSON-Dokumenten verhindert Informationsinseln und bietet einzigartige Möglichkeiten. Eine weitere sehr spannende Anwendungsmöglichkeit sind SQL Pattern Matching Abfragen auf JSON-Dokumenten. Hier

entstehen Möglichkeiten, die dem NoSQL-Verfechter leuchtende Augen bereiten. Und mit der Version 12.2 kommt noch die direkte Integration der In-Memory Option mit JSON.

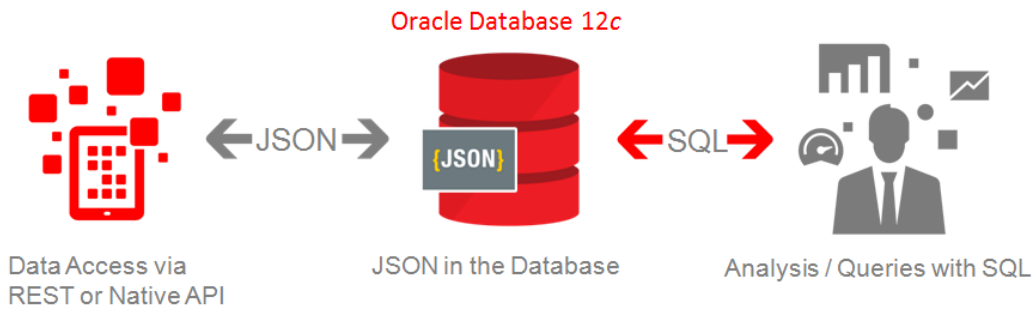


Abb. 4: JSON und SQL wachsen zusammen

SODA im SQLcl

Das neue Frontend von Oracle, SQLcl – ein Kommandozeilentool mit allen Möglichkeiten von SQL*Plus durch komfortable Möglichkeiten aus dem SQL Developer erweitert -, hat eine SODA-Schnittstelle implementiert, die die oben beschriebenen Möglichkeiten aus der Kommandozeile ermöglicht. So wird beispielsweise durch den Befehl

```
soda create mycollection
```

eine neue Collection in der Datenbank angelegt. Sie finden SQLcl im OTN auf den Seiten des SQL Developer. Es lohnt sich !

ORDS – Oracle REST Data Services

Neben der beschriebenen Schnittstelle ermöglicht ORDS auch eine analoge Verwendung der Oracle NoSQL-Datenbank und eine Schnittstelle, die es ermöglicht ein Mapping von JSON-Dokumenten in relationale Strukturen und zurück anzubieten. Auch können mit PL/SQL eigene Micro-Services definiert werden.

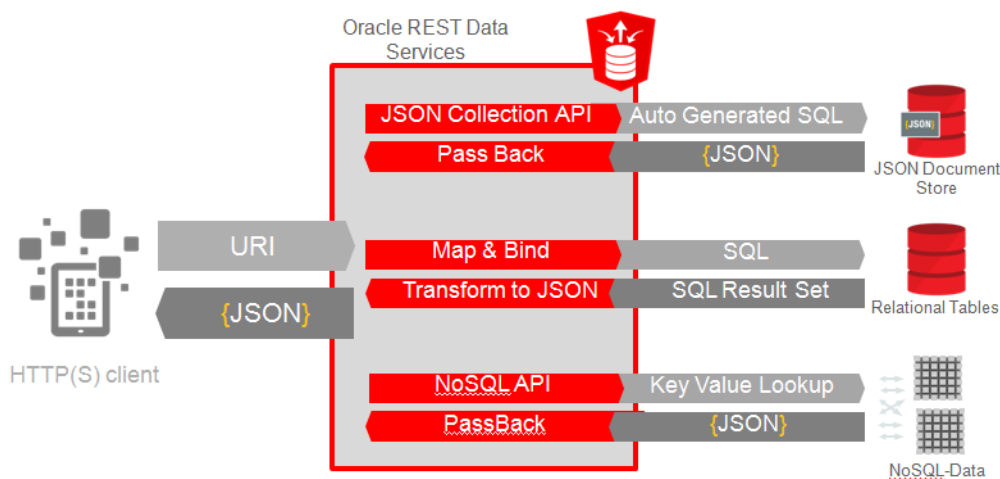


Abb. 5: Oracle REST Data Services

Hinweis

Die Verwendung von SODA setzt das Release 12.1.0.2 voraus und benötigt den Patch 20885778.
Mögliche Java-Container für SODA for Rest sind der Oracle WebLogic Server ($\geq 11gR1$), GlassFish Server (\geq Release 3.1.2) und Apache Tomcat ($\geq 7.0.56$).

Kontaktadresse:

Rainer Willems
Oracle Deutschland B.V. & Co. KG
Robert-Bosch-Strasse 5
D-63303 Dreieich

Telefon: +49 (0) 6103-397-290
E-Mail rainer.willems@oracle.com
Internet: www.oracle.de