Oracle NoSQL Database

• 18.3 New Features

• Oct, 2018
Oracle NoSQL Database – 18.3 New Features

- Querying GeoJSON Data
- Identity Column
- Namespaces
- Admin Web Service
Oracle NoSQL Database – 18.3 New Features

Querying GeoJSON Data
Oracle NoSQL Database – GeoJSON Query Function

Customer Benefit: Index and Query Spatial Data

Find NATURE PARKS in northern California

SELECT t.poi as park
FROM PointsOfInterest t
WHERE t.poi.kind = "nature park" and
geo_intersect(t.poi.location,
{ “type” : “polygon”,
 “coordinates” : [
 [121.94, 36.28],
 [117.52, 37.38],
 [119.99, 39.00],
 [120.00, 41.97],
 [124.21, 41.97],
 [124.39, 40.42],
 [121.94, 36.28]
 ]
})
Identity Column
Oracle NoSQL Database – IDENTITY column

- Identity column
- Primary Keys automatically generated
- Increment or decrement a counter
- Works with both API and SQL

```
CREATE Table USERS (id INTEGER GENERATED ALWAYS AS IDENTITY (START WITH 1 INCREMENT BY 1),
name STRING,
PRIMARY KEY (id))
```

TableAPI api = store.getTableAPI();
Table table = api.getTable("foo");
Row row = table.createRow();
row.put("name", "joe");
api.put(row, null, null);
System.out.println("Value :") + row.get("id")); //Prints 1
row = table.createRow();
row.put("name", "smith");
api.put(row, null, null);
System.out.println("Value :") + row.get("id")); //Prints 2
Namespace
Oracle NoSQL Database - Namespace

- Create custom containers of tables
- All objects within namespaces must be uniquely identified.
- Manage authorization at container level
- Import/export at container level

Customer Benefit: Allow easy organization of tables

Table Namespace: Sales
Table: customer

<table>
<thead>
<tr>
<th>id</th>
<th>customername</th>
<th>salevalue</th>
<th>saledate</th>
</tr>
</thead>
</table>

Table Namespace: Service
Table: customer

<table>
<thead>
<tr>
<th>id</th>
<th>customername</th>
<th>quantity_sold</th>
<th>Sale_id</th>
</tr>
</thead>
</table>

SELECT * from Sales: customer

SELECT * from Service: customer
Oracle NoSQL Database – 18.3 New Features

Admin Web Service
Oracle NoSQL Database – Admin Web Service

- Admin operations available via REST APIs and CLI
- HTTP Based
  - GET
  - PUT
  - DELETE
  - POST
- Payload
  - JSON
- SSL Encrypted

Customer Benefit: Automate Ops with REST API

Request URI: "https://NoSQL:20001/V0/nosql/admin/topology" -d '{"command":"show"}'

METHOD: POST

Response JSON:

```
{
  "operation": "show topology",
  "returnCode": 5000,
  "description": "Operation ends successfully",
  "returnValue": {
    "storeName": "mystore",
    "numPartitions": 20,
    "sequenceNumber": 24,
    ... (topology related JSON information)
  }
}
```