

Db2 for z/OS: Native REST

NEDb2UG

December 7, 2023

Tom Toomire

toomire@us.ibm.com

Db2 for z/OS Development



Db2 for z/OS REST objectives

Using REST and JSON to invoke one SQL statement or stored procedure

Enabling new business value for your enterprise data

Modernizing using the power of SQL

Unleashing Db2 data for the API economy

What is a RESTful API?

REST stands for Representational State Transfer architecture. (It is sometimes spelled "ReST".)

- stateless,
- client-server,
- cacheable communications protocol
 - ✓ HTTP protocol is used

A RESTful API is an application programming interface (API) that uses HTTP requests to GET, PUT, POST, and DELETE data.

Uniform resource identifier (URI) and JSON name-value pairs

`http://www.myhost.com:port/account/update`

```
{  
  "account": "12345",  
  "lastName": "Bloggs",  
  "action": "Deposit",  
  "amount": "$1000.00"  
}
```

HTTP request method examples: RESTful

Using the URL: <https://myhost.com/customer/235>

GET = Record for customer #235. (in SQL terms - **SELECT**)

PUT + Info = Updated record for customer #235. (in SQL terms - **UPDATE**)

POST + Info = New record for customer #235. (in SQL terms - **INSERT**)

DELETE = Customer #235 Deleted. (in SQL terms - **DELETE**)

HTTP Request Method Examples: *Db2 native REST*

Using the URL: <https://myhost.com/services/collection/servicename>

POST + [SELECT]

“customer”: “235”

POST + [UPDATE]

“customer”: “235” ,
“telephone”: “987-654-3210”

POST + [INSERT]

= “customer”: “235”
(all required name value pairs)

POST + [DELETE]

= “customer”: “235” .

NOTE

Db2 native REST can only use POST for applications, however this can be paired with respective **SQL** terms such as SELECT, INSERT, UPDATE, DELETE, CALL, TRUNCATE, and WITH.

*For example, you can use the POST method with the **SQL** issuing a DELETE.*



Db2 REST service properties

Db2 REST service invocation

- Direct Db2 DDF REST access

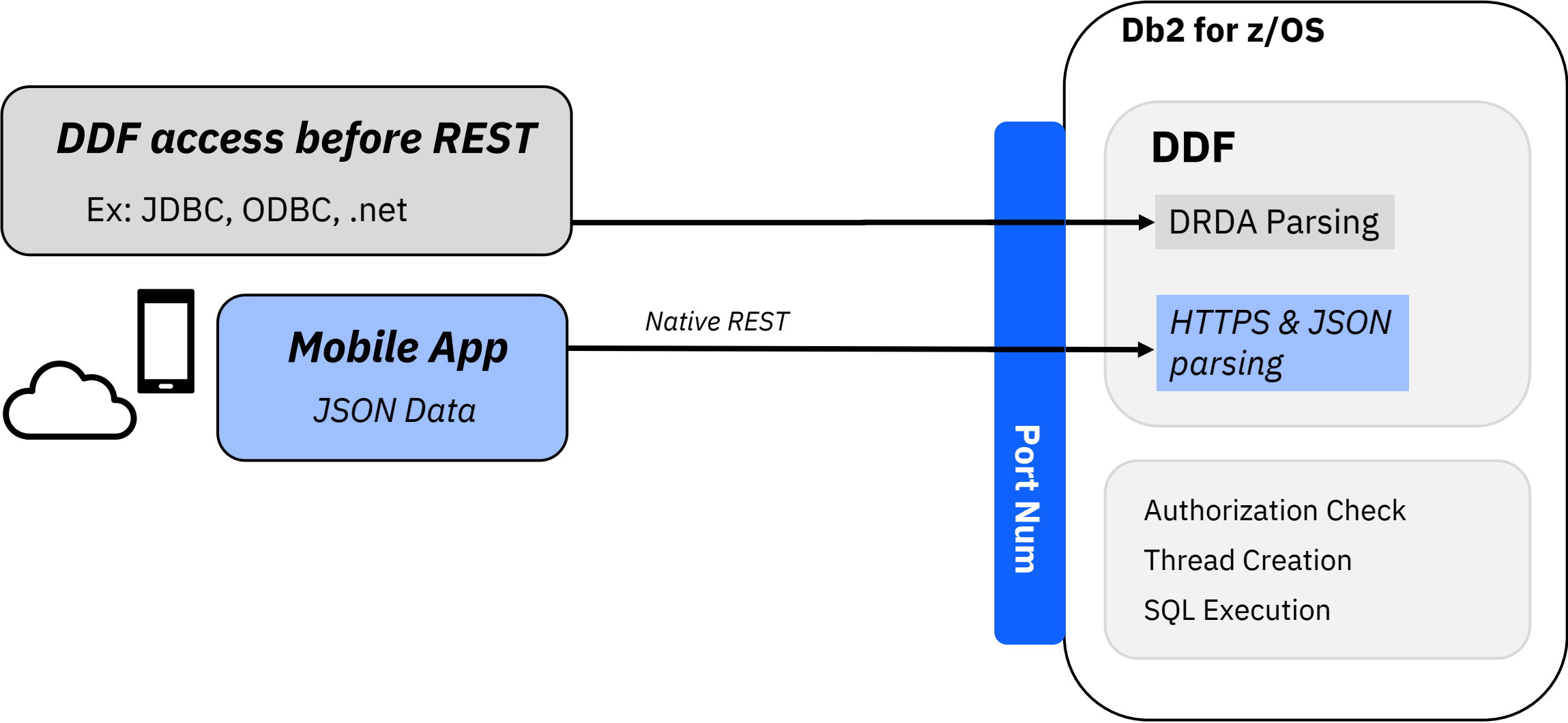
Service details

- One SQL statement or stored procedure call per service
 - Service is a statically bound package in Db2
- CALL, DELETE, INSERT, SELECT, TRUNCATE, UPDATE and WITH
 - MERGE can be in a service comprised of a stored procedure, not in a service comprised of a single SQL statement

All of the various Db2 base SQL data types

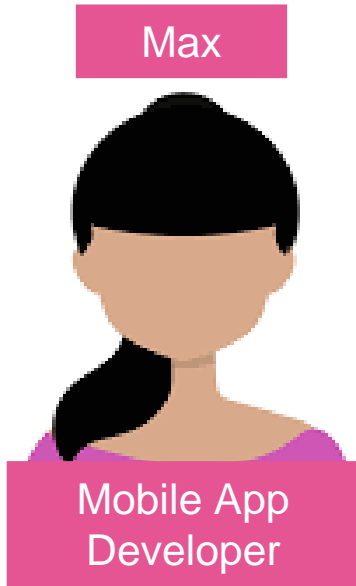
- Including BLOB, CLOB, DBCLOB and XML

Db2 REST architecture diagram



Appendix

When a developer goes to retrieve data

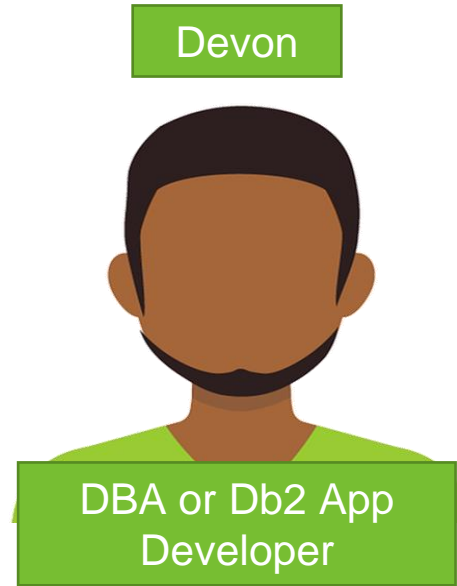


Invokes a Db2 REST service

- Service consists of stored procedure or SQL statement

Output returns in JSON format

Doesn't need to know SQL, nor that the data came from Db2



Creates a Db2 REST service

- Service consists of a stored procedure or SQL statement

Creates or reuses the stored procedure or SQL statement used in the REST call

Doesn't need to know JSON

Managing Db2 REST services

REST client in browser

- Typically a plug-in

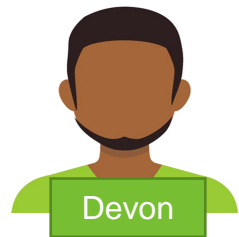
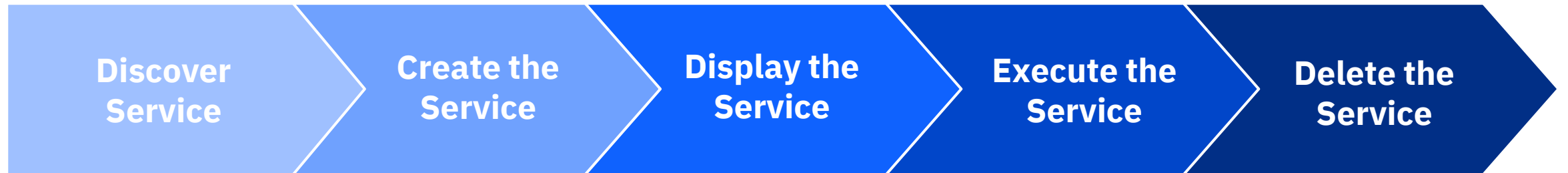
REST app

- Browser look and feel

BIND subcommand

- Standard Db2 interaction

Db2 REST service progression



URI format with versioning support

Original

`/services[/<collection id>]/<service name>`

Example: `/services/SYSIBMServices/displayEmployee`

Versioning

`/services/<collection id>/<service name>[/<version>]`

Example: `/services/SYSIBMServices/selectByEmpNum/V1`

Db2 REST Services with z/OS Connect

Db2 user created native REST services are invoked by the *POST method only*

Mobile and cloud programmers following the RESTful API design model use the *HTTP Methods (Verbs): POST, GET, PUT and DELETE*

z/OS Connect's "API Editor" can map a Db2 POST method SQL statement to the appropriate RESTful method for a given behavior

Architecture Diagram

