

Mainframe Db2 APIs – Creating the Future

Using z/OS Connect APIs to access Mainframe Data through Db2

Andy Johnson

Randy Black

Mainframe Db2 DBAs

Who are we?

Andy Johnson – JohnsoaU@Nationwide.com

- Application DBA for the past 8 years
- Specializing in IDAA and Db2 REST services/zOS Connect
- Mainframe Developer for 12 years prior
- Worked with DB2 and IMS

Randy Black – blackr3@Nationwide.com

- Application DBA for the past 25 years
- Specializing in IDAA/Db2 REST services/zOS Connect & SQL Performance
- Mainframe Developer for 10 years prior
- Worked with COBOL and DB2

What is an API and what is z/OS Connect EE?

API – Application Programming Interface

An API is an interface between different parts of a computer program intended to simplify the implementation and maintenance of software.

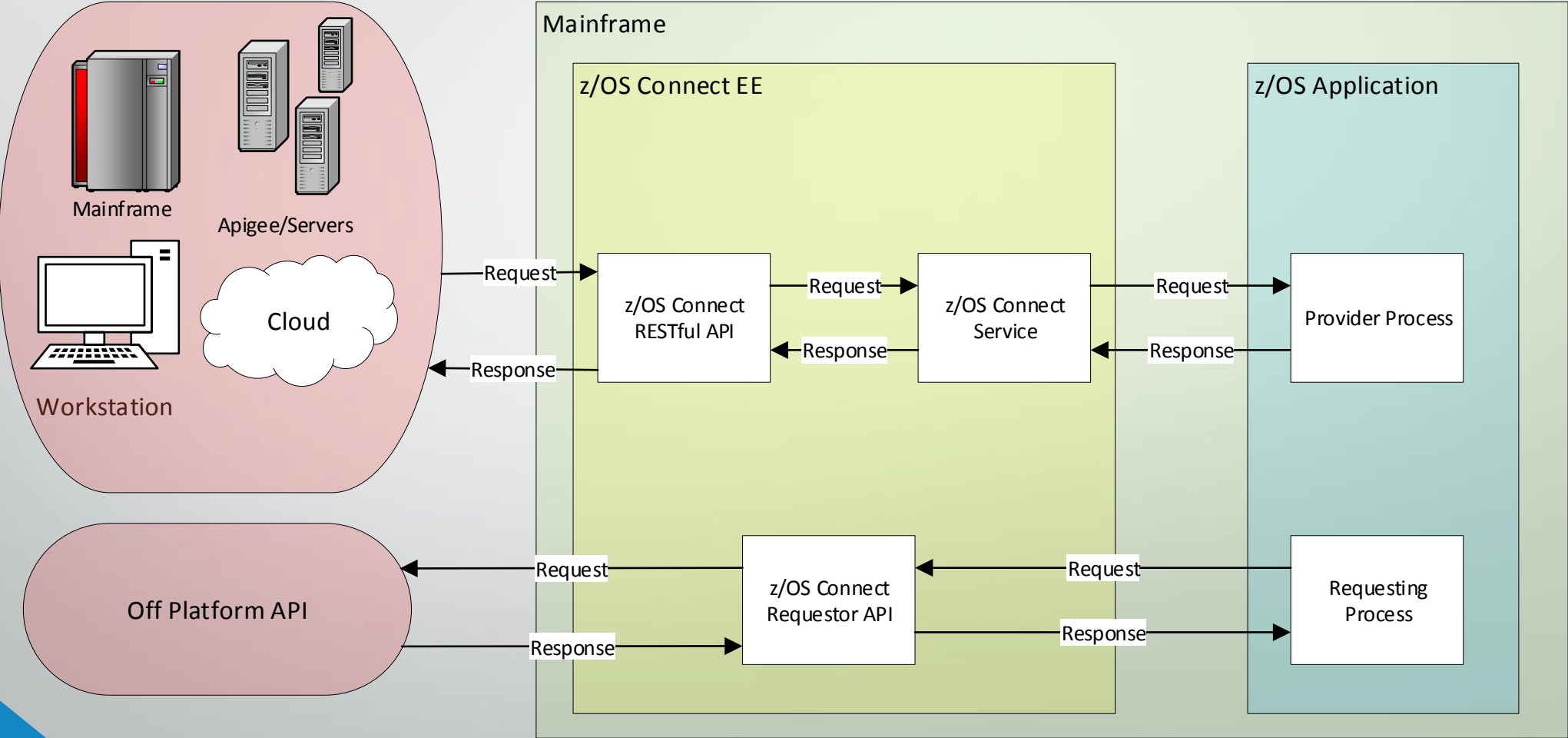
z/OS Connect Enterprise Edition (version V3.0.32)

- The tool used to create and manage APIs that access mainframe data and utilize APIs going on and off the mainframe.
- Creates RESTful APIs supporting the standard API methods: PUT, POST, DELETE and GET as well as PATCH and HEAD.
- API enables Batch, IMS, CICS, MQ and Db2/IDAA as well as enabling a mainframe process to use an off platform API.
- API Toolkit version - V3.0.8.1

What can an API do?

- Provider
 - A provider API is where the mainframe application is creating an API to provide data for a consumer
 - z/OS Connect enables the mainframe application to develop the API and the consumer to execute the API
- Requestor
 - A requestor API is where the mainframe application is requesting data from an off-platform API
 - z/OS Connect provides the off-platform link that the mainframe application can consume
- Security
 - RACF – Standard access to anything mainframe
 - z/OS Connect supports RACF passtickets
 - OAuth – Standard access for internet applications
 - Supports JWT, APIKey and others

How does z/OS Connect handle Providers and Requestors?



What in Db2 can be exposed as a Provider API?

Native Db2 REST service

- Native Db2 REST must be turned on within Db2
 - Requires V11 NFM and a few APARs applied
- A Db2 REST service supports a single SQL statement
- Db2 REST is not RESTful...Db2 only supports POST for execution and GET for discovery
- z/OS Connect turns a Db2 REST service into a RESTful API

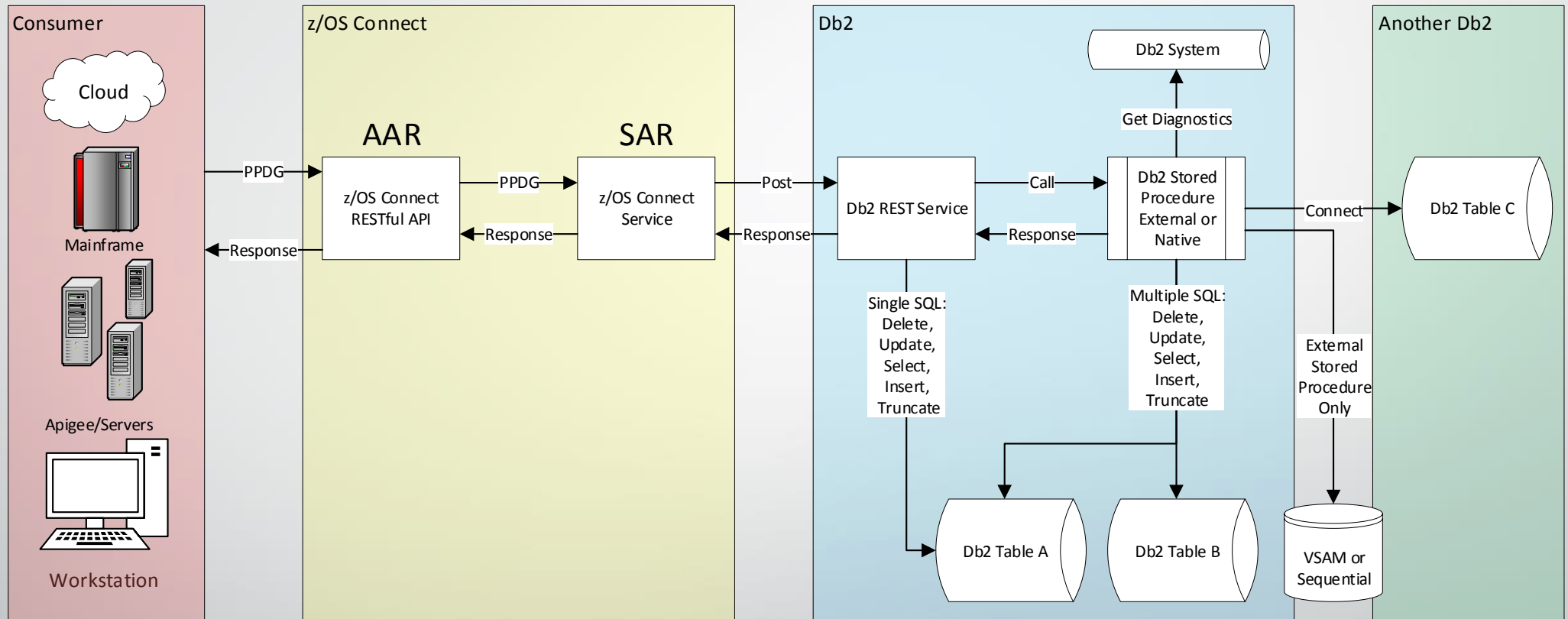
What can my Single SQL statement be?

- INSERT, UPDATE, DELETE, TRUNCATE or SELECT
- CALL statement for a native or external stored procedure – Recommended

What data can I go after?

- Db2 data from Db2 or IDAA
- VSAM or Sequential data within an external stored procedures WLM
- Off Platform data through another API
 - Mainframe to Distributed
 - Mainframe to Mainframe

How does Db2 and z/OS Connect work together?



PPDG = Put, Post, Delete or GET

Using a Stored Procedure allows you to build in business logic and error handling that a single SQL statement Db2 REST Service does not support

Who is a Candidate for zOS Connect APIs?

- Cloud

A mainframe application migrating to the cloud but leaving their data on the mainframe

Any application on the cloud that needs data from the mainframe

- Mobile/Web

A mainframe application wanting to expose their data for mobile/browser consumption

- Modernization

A mainframe application looking to enhance their interfaces. Existing interfaces should be easy to convert into APIs.

A mainframe application that needs data from an off-platform API

- Cost Conscious

APIs are zIIP eligible

What are we Doing Today?

- Install Db2 Connect on your sever. Stay current with upgrades.
- Create an ODBC/JDBC Connection profile
- Open the connection in the code
- Run SQL or call a stored procedure
- Close the connection

Why would we want to change?

- APIs are the industry standard
- Mainframes may be labeled as taboo or obsolete and an API hides the underlying technology
- Easier integration without the technology specific drivers and software

What is a better way?

To Provide Data via Db2

- Create and deploy an API and all related artifacts.
- Call the API
 - Via the zOS Connect URL
 - Via an 3rd party Provider that uses the zOS Connect URL

To Request Data via COBOL

- Configure the requestor API within zOS Connect.
- Call the API via the URL

Become Technology Agnostic!!!!

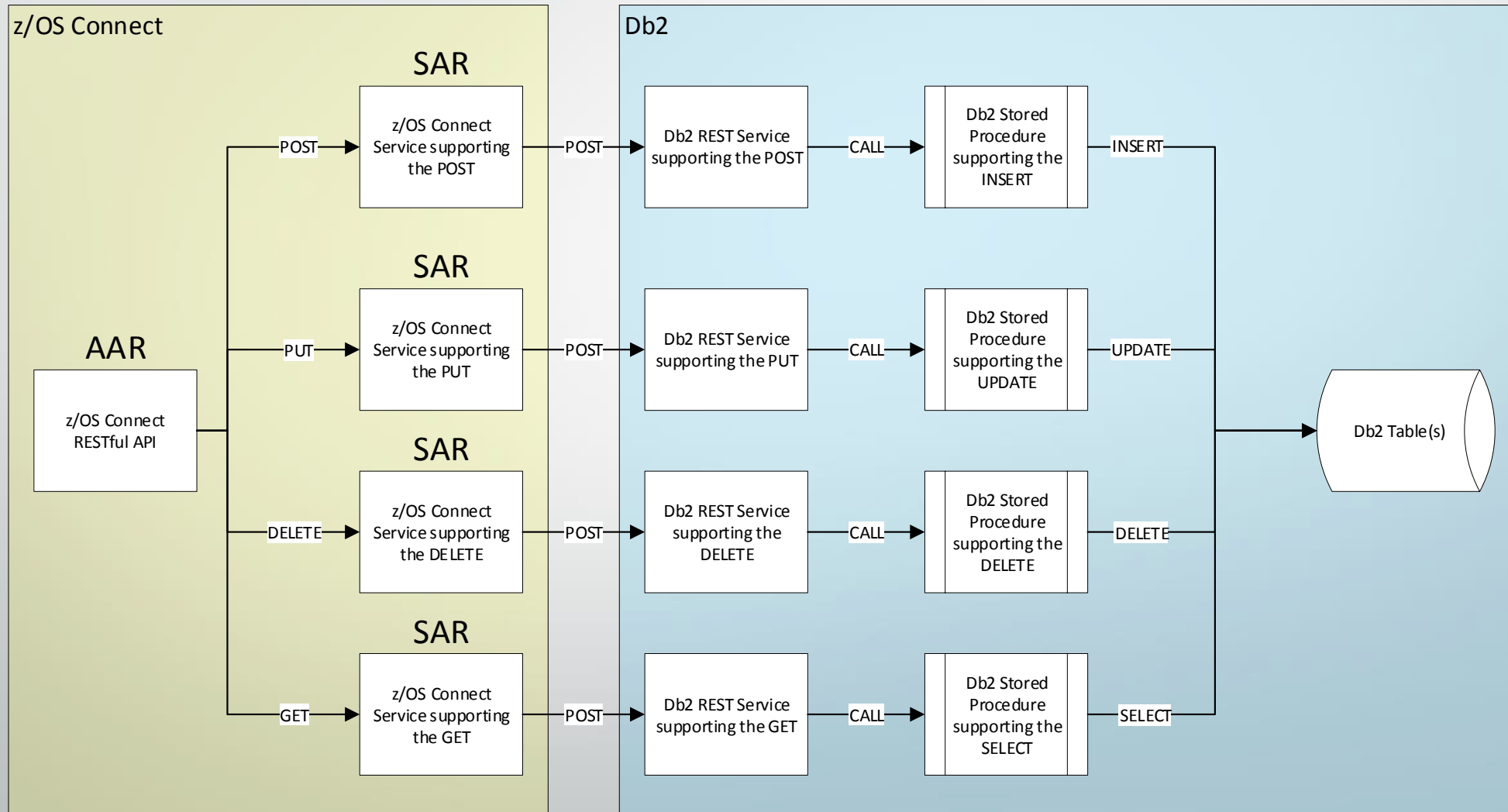
The design and architecture of the consuming application is not dependent on the platform of the providing application.

What does better look like?

Demo – Db2 SQL - Browser

Demo – Full Application – Stored Procedure to Db2 - Browser

What was that Full Application Doing?



What is a Db2 Provider API made of?

- Db2 REST service – The single SQL service that z/OS Connect will invoke
- z/OS Connect Db2 Service Project – the design location to build your zOS Connect service archive (SAR)
 - SAR – service archive – mapping of the z/OS Connect service to the Db2 REST service
 - Inside the SAR is the REQUEST(input) and RESPONSE(output) JSON files and a YAML that holds the configuration to the Db2 REST service
- z/OS Connect API Project – The design location to build your API and methods (AAR)
 - AAR – API archive – mapping of the API to the Service Archive (SAR)
- Open API Specification (OAS) – YAML/JSON Documentation on the API and it's features

Many of the API artifacts can be generated from the tool kits!!

How do I make a Db2 REST service?

- Through an API tool like Postman
 - When you activate Db2 REST services you will have access to IBMs built in Service Manager
 - `http://your.db2.url:port/services/DB2ServiceManager`
 - Run a POST to invoke the service manager providing the parms in your request body
- In Batch using a BIND SERVICE command
 - Put the SQL in a file and reference it through the SQLDDNAME parm
 - Sample can be found via the IBM support site
 - https://www.ibm.com/support/knowledgecenter/SSEPEK_12.0.0/comref/srvc/tpc/db2z_cmd_bindservice.html

Both options will create a package in SYSPACKAGE for the service.

Don't forget that you will need EXECUTE authority on the Db2 REST package for anyone that wants to use your service.

What Does a Db2 REST Service Look Like?

Create a REST Service

```
{
  "requestType": "createService",
  "serviceName": "GetProfileData",
  "description": "Get a profile and all it's attributes.",
  "collectionID": "MyRestCollection",
  "version": "V1",
  "owner": "MyOwner",
  "qualifier": "MySchema",
  "sqlStmt": "CALL GetProfileData
(:Connection, :AuthID, :PlanName, :CollID, :PkgName, :Location, :KeyWords,
:SQLState, :SQLCode, :Message, :LineNbr, :ReturnedSQLCode, :TokenCount, :O
rdinalToken1, :OrdinalToken2, :OrdinalToken3, :OrdinalToken4, :OrdinalTo
ken5, :MessageID, :MessageText, :Module, :ReturnedSQLState) "
}
```

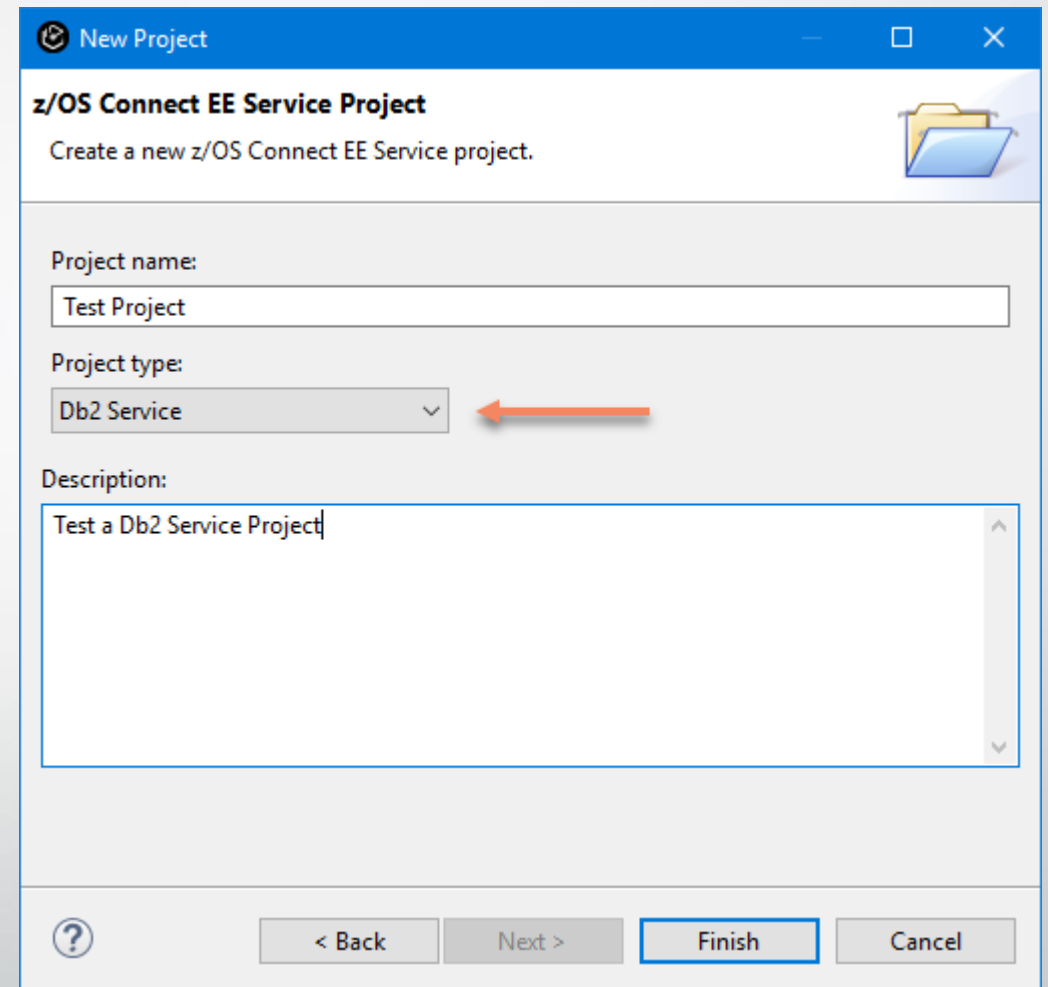
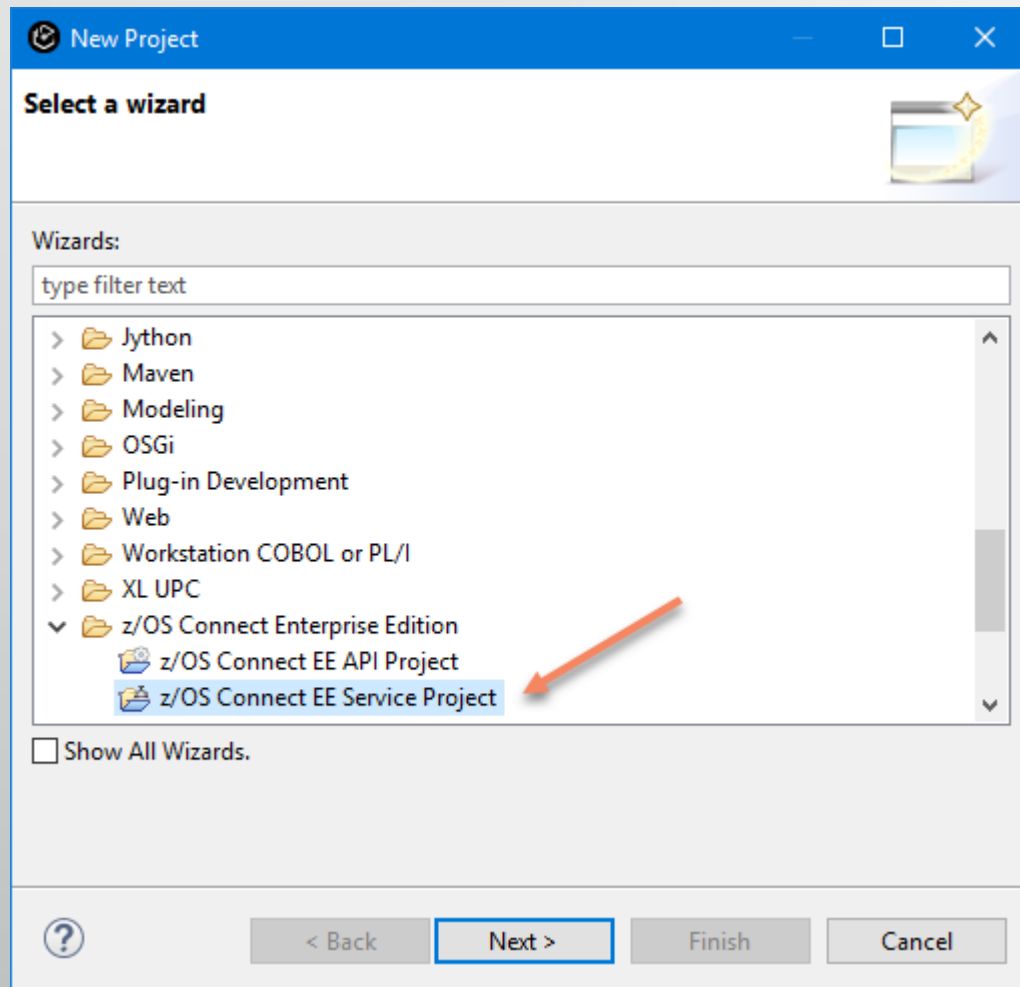
Drop a REST Service

```
{
  "requestType": "dropService",
  "serviceName": "GetProfileData",
  "collectionID": "MyRestCollection",
  "owner": "MyOwner"
}
```

How do I create and deploy a SAR?

- Create a new Db2 Service project (SAR) using the API toolkit plugin in IDz
 - The name of your Service project will become the name of the service in zOS Connect
- Select Import from Db2 service manager... to link to your Db2 REST service
 - Identify the connection you want to use to display the available Db2 REST services
 - Pick your Db2 REST Service from the list of available Db2 REST Services
- Switch to the configuration tab to select the z/OS Connect connection reference into Db2
 - A connection will need defined within the z/OS Connect configuration
- Save the project to accept the generated REQUEST and RESPONSE documents and your connection reference
- Click the Deploy button to generate the SAR and move it into the zOS Connect Service list
 - The deploy will ask you to choose the zOS Connect sever to use with the deploy

How do I create a Db2 Service Project (SAR)?



How do I link to my Db2 REST Service?

General Information

Edit or update the general information of the service.

Type: Db2 Service
Version: 1.0.0
Description: Test a Db2 Service Project

Actions

Steps to create a service:

1. Input service version.
2. Import JSON schemas from a Db2 service manager or your local machine.
3. [Complete the configuration for the service.](#)
4. [Deploy the service.](#)

Define Db2 service

Import a Db2 native REST service from a Db2 service manager. Alternatively, you can import a REST service from a local file.

Import from Db2 service manager... **Step 1**

Collection ID: SYSIBMSERVICE
Db2 native REST service name: myService
Db2 native REST service version: V1
Request JSON schema:
Response JSON schema:

Import Db2 service from service manager

Db2 service manager connection: DBT2 **Step 2**

Type to search...

Service Name	Version	Collection ID	Description	URI
DeleteAttribut...	V1	DBDSGC2_RE...	Delete a singl...	/services/DB...
DeleteProfileD...	V1	DBDSGC2_RE...	Delete a profi...	/services/DB...
GetOncallList	V1	DBDSGC2_RE...	Get Oncall List	/services/DB...
GetOncallList...	V1	DBDSGC2_RE...	Get Primary o...	/services/DB...
GetProfileData	V1	DBDSGC2_RE...	Get a profile ...	/services/DB...

Step 3 **Step 4**

Import Cancel

Step 1 – Click Import from Db2 service manager...

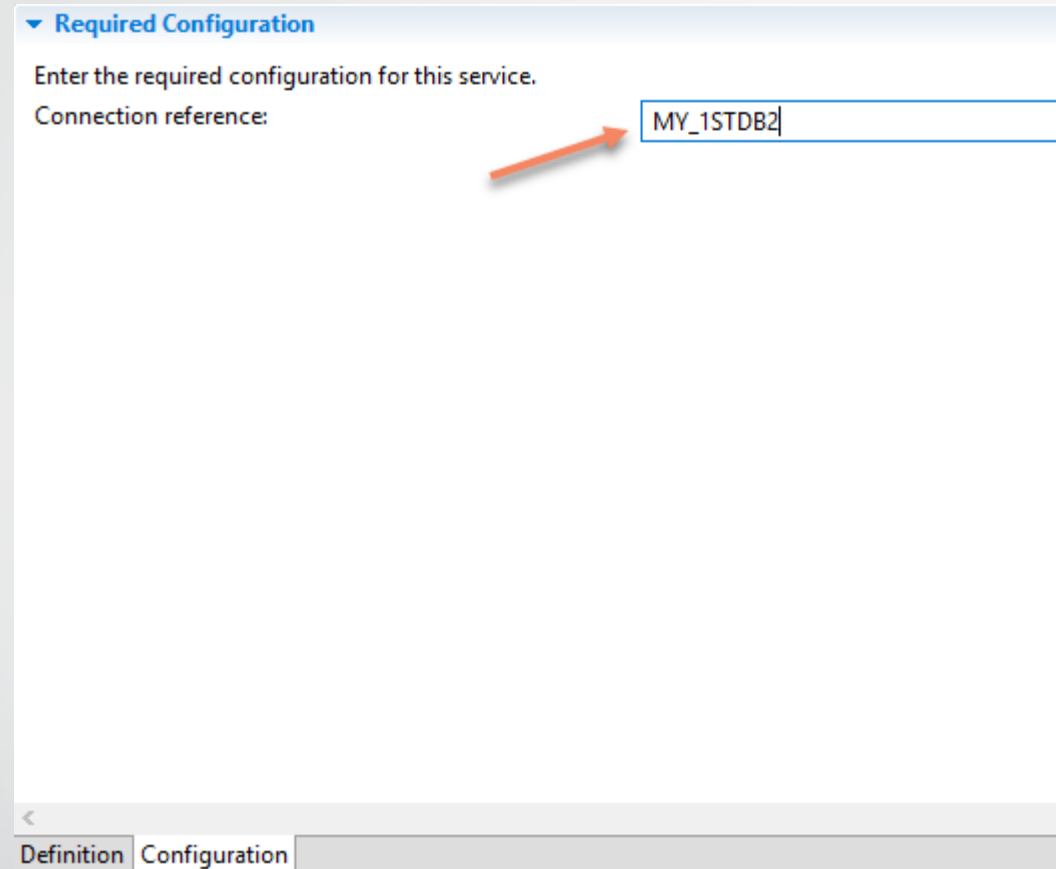
Step 2 – Select the connection where your Db2 REST service lives

Step 3 – Highlight the Db2 REST service you want to transform into a z/OS Connect Service

Step 4 – Click Import

Be sure to save the project after you import your Db2 REST service

How Does the Service Connect to Db2?



▼ Required Configuration

Enter the required configuration for this service.

Connection reference:

<

Definition Configuration

This is the connection that was set up within the zOS Connect configuration

Where is the Connection String Stored?

➤ Server.XML

- XML based file that contains all the settings for z/OS Connect
- Supports INCLUDE of additional files incase you want to break apart your settings
- Things found in this file related to Db2:
 - Db2 Connections
 - RACF Passticket setup
 - Trace settings
 - Configuration Overrides
- At Nationwide we have Db2, trace and override settings in their own XML files that are included into the master Server.XML

```
<zosconnect_zosConnectServiceRestClientBasicAuth
  id="RACF_passticket"
  applName="RACFDDF"/>
```

```
<zosconnect_zosConnectServiceRestClientBasicAuth
  id="HardCoded"
  userName="MYID"
  password="{xor}Hi8vMzoYDWc="/>
```

Notice that the password is encrypted using WebSphere {xor}

```
<zosconnect_zosConnectServiceRestClientConnection
  id="MY_1STDB2"
  receiveTimeout="60s"
  host="http://MY.1STDB2.URL"
  port="9999"
  connectionTimeout="30s"
  basicAuthRef="RACF_passticket"/>
```

```
<zosconnect_zosConnectServiceRestClientConnection
  id="MY_2NDDB2"
  receiveTimeout="60s"
  host="http://MY.2NDDB2.URL"
  port="9999"
  connectionTimeout="30s"
  basicAuthRef="HardCoded"/>
```

Now that config is done...how do I Deploy?

Make sure you SAVE your project before you deploy!

▼ Actions

Steps to create a service:

1. Input service version.
2. Import JSON schemas from a Db2 service manag
3. Complete the configuration for the service.
4. Deploy the service.
5. Export the service.

z/OS Connect EE Server: ATE_SYC

The following services will be created on the server:

Service name	Version	Type
Test Project	1.0.0	Db2 Service

OK Cancel

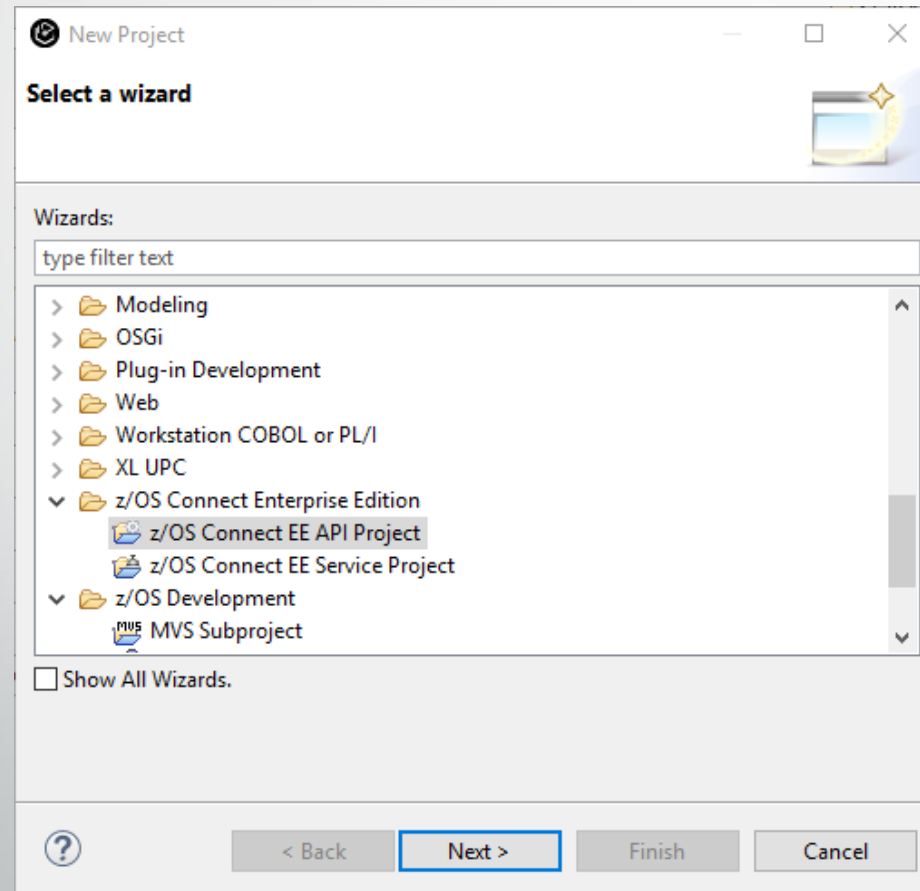
Are there any configuration overrides?

- At the service name level, z/OS Connect can be configured to override two values
 - Collection ID of your Db2 REST service
 - Connection Reference that links your service to Db2
- The overrides are processed from server.xml
 - Same place that the connection reference are stored
 - The override is applied at execution time
- One set of override pairs per zOS Connect service per z/OS Connect server

```
<zoscconnect_services>
  <service name="exampleDb2a" >
    <property name="collectionId" value= "AppAcollid" />
    <property name="connectionRef" value="MY_1STDB2" />
  </service>
  <service name="exampleDb2b" >
    <property name="collectionId" value= "AppBcollid" />
    <property name="connectionRef" value="MY_2NDDB2" />
  </service>
</zoscconnect_services>
```

How do I create and deploy an AAR?

- Create a new z/OS Connect API project using the API toolkit plugin in IDz
- Under your new project, open the packages.xml file to map your methods
- Associate a SAR to each of your methods and map the Request/Response variables
- Save package.xml
- Right Click on your project and select z/OS Connect EE then Deploy API to z/OS Connect EE Server



How does the AAR become RESTful?

Package.xml – Sample Method to Service Mapping with PATH extensions

db2-profile-table-maintenance API

API Editor

Describe your API

Name: db2-profile-table-maintenance Description: Maintain data on the Db2 Profile Table

Base path: /db2-profile-table-maintenance

Version: 1.0.0

+ Path ↑ ↓ ✕

/Development/Profile

+ Methods (4) ▼

▶	POST	DBDSGCX_REST.PostProfileData	Service...	Mapping...	↑	↓	✕
▶	GET	DBDSGCX_REST.GetProfileData	Service...	Mapping...	↑	↓	✕
▶	PUT	DBDSGCX_REST.PutProfileData	Service...	Mapping...	↑	↓	✕
▶	DELETE	DBDSGCX_REST.DeleteProfileData	Service...	Mapping...	↑	↓	✕

+ Path ↑ ↓ ✕

/Development/Attribute

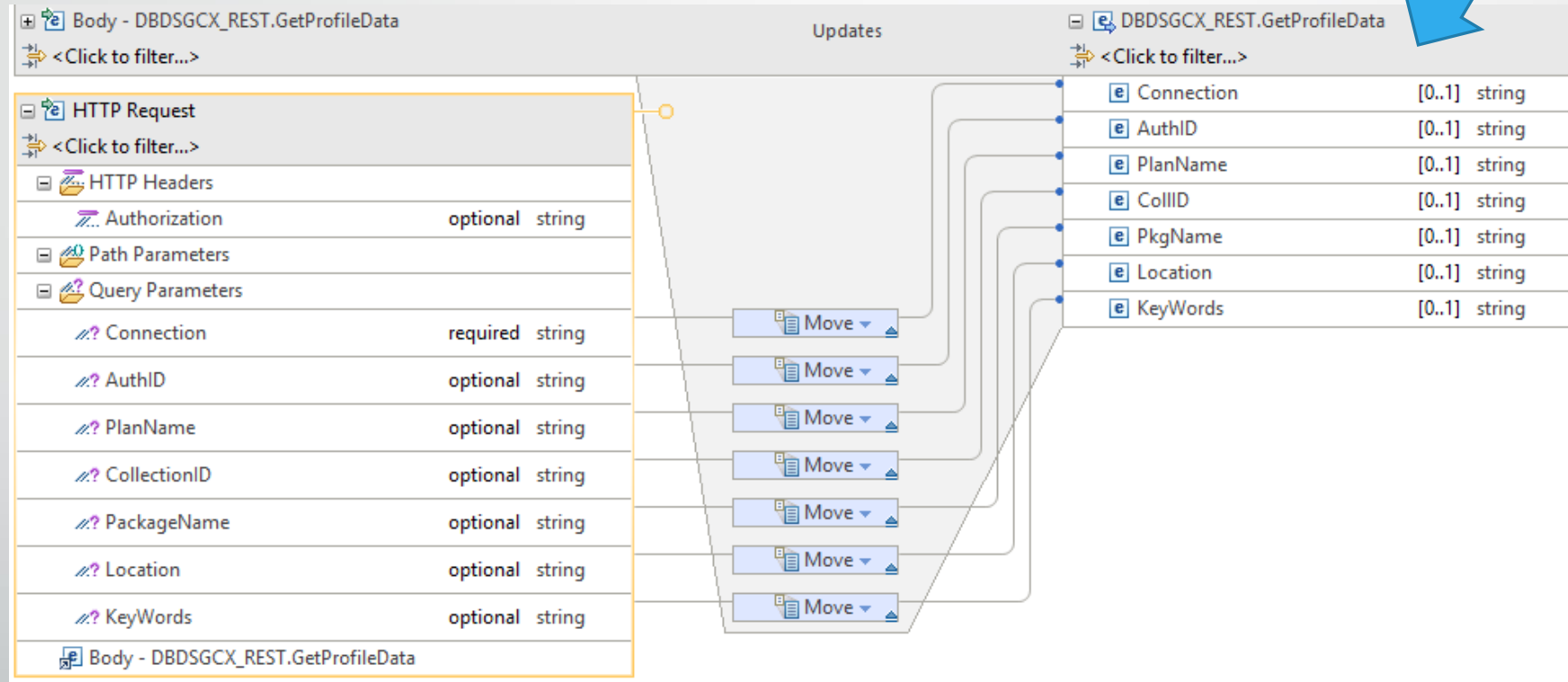
+ Methods (3) ▼

▶	POST	DBDSGCX_REST.PostAttributeData	Service...	Mapping...	↑	↓	✕
▶	PUT	DBDSGCX_REST.PutAttributeData	Service...	Mapping...	↑	↓	✕
▶	DELETE	DBDSGCX_REST.DeleteAttributeData	Service...	Mapping...	↑	↓	✕

How do the AAR parms map to the SAR parms?

Package.xml – Sample REQUEST mapping – Input Variables

This is from the SAR file REQUEST JSON



Be descriptive in your Query Parameter Names – This is what your consumers will see when they use your API.

How do I Deploy my API?

The image shows a screenshot of an IDE's menu system. The 'z/OS Connect EE' menu item is selected, and its sub-menu is open, showing options like 'Deploy API to z/OS Connect EE Server', 'Export z/OS Connect EE API Archive', and 'Import z/OS...'. In the background, a 'Deploy API' dialog box is visible. The dialog shows the target server as 'z/OS Connect EE Server: ATE_SYC'. It lists existing APIs on the server in a table:

API name	Version	Base path
db2-profile-table-history	1.0.0	/db2-profile-table-history

Below the table, there is a checkbox for 'Update existing APIs' which is unchecked. A red error icon and text state: 'One or more APIs already exist on the server.' The dialog has 'OK' and 'Cancel' buttons at the bottom right.

What should I call my APIs?

- Be descriptive – clientPreferences not cPref
- Use camelCase
- Make sure it's unique
 - Don't overlay an existing API
 - A DEPLOY will warn you if the API already exists
- Don't be afraid to use versioning
- Use a PATH to extend your URL
 - Make sure the PATH is unique within your API

How do I Call an API?

Javascript - Provider

```
var xmlhttp = new XMLHttpRequest();
var xmlhttpURL = 'my.api.url/path/parms';
var myuser = 'RACFID';
var mypw = 'RACFIDPW';
xmlhttp.onerror = function()
{
    .....
}
xmlhttp.onreadystatechange = function()
{
    if (xmlhttp.readyState == 4)
    {
        if (xmlhttp.status == 200)
        {
            var myObj = JSON.parse(xmlhttp.responseText);
            var OP = myObj['Output Parameters'];
            var RS1 = myObj['ResultSet 1 Output'];
            var RS2 = myObj['ResultSet 2 Output'];
            var holdSQLCODE = OP.SQLCode;
            var holdMessage = OP.Message;
            for (x in RS1)
            {
                Field1 = rs1[x].Field1
                Field2 = rs1[x].Field2
                .....
            }
            .....
        }
    }
}
xmlhttp.open("GET",xmlhttpURL, true);
xmlhttp.withCredentials = true;
xmlhttp.setRequestHeader('Accept','application/json');
xmlhttp.setRequestHeader('Content-type','application/json');
xmlhttp.setRequestHeader('Authorization','Basic ' + btoa(myuser+':'+mypw));
xmlhttp.setRequestHeader('Access-Control-Allow-Origin','*');
xmlhttp.send();
```

How do I Call an API?

COBOL - Requestor

```
01 API-INFO-OPER1.
  03 BAQ-APINAME          PIC X(255)
    VALUE 'Enterprise-Phone-Validation_1.0'.
  03 BAQ-APINAME-LEN      PIC S9(9) COMP-5 SYNC VALUE 31.
  03 BAQ-APIPATH         PIC X(255)
    VALUE '%2Fcustomer-information-management%2Fenterprise-pho
-   'ne-validation%2Fv1%2FvendorParameters'.
  03 BAQ-APIPATH-LEN     PIC S9(9) COMP-5 SYNC VALUE 88.
  03 BAQ-APIMETHOD     PIC X(255) VALUE 'GET'.
  03 BAQ-APIMETHOD-LEN PIC S9(9) COMP-5 SYNC VALUE 3.
01 BAQ-REQUEST-INFO.
.....
01 BAQ-RESPONSE-INFO.
.....
  03 BAQ-RETURN-CODE     PIC S9(9) COMP-5 SYNC.
    88 BAQ-SUCCESS     VALUE 0.
.....
01 BAQ-REQUEST-PTR      USAGE POINTER.
01 BAQ-REQUEST-LEN     PIC S9(9) COMP-5 SYNC.
01 BAQ-RESPONSE-PTR    USAGE POINTER.
01 BAQ-RESPONSE-LEN    PIC S9(9) COMP-5 SYNC.
01 COMM-STUB-PGM-NAME  PIC X(08) VALUE 'BAQCSTUB'.
.....
CALL COMM-STUB-PGM-NAME USING
    BY REFERENCE API-INFO-OPER1
    BY REFERENCE BAQ-REQUEST-INFO
    BY REFERENCE BAQ-REQUEST-PTR
    BY REFERENCE BAQ-REQUEST-LEN
    BY REFERENCE BAQ-RESPONSE-INFO
    BY REFERENCE BAQ-RESPONSE-PTR
    BY REFERENCE BAQ-RESPONSE-LEN.
IF BAQ-SUCCESS THEN
  loop thru output
ELSE
  error handling
END-IF.
```

What tools are Nationwide using?

- IDz w/ the z/OS Connect API toolkit plug-in
 - Build artifacts used to create an API
 - Deploy Artifacts
 - API Toolkit version - V3.0.8.1
 - IDz version - V14.2.2
- Git
 - Source Code repository for API artifacts
- Postman
 - Execute/Test all API artifacts
- Browser
 - Additional method to execute/test APIs

Wanna build an API?

Let's use IDz and postman to build and test a Db2 provider API!

Questions?

