PUGCHALLENGE AMERICAS

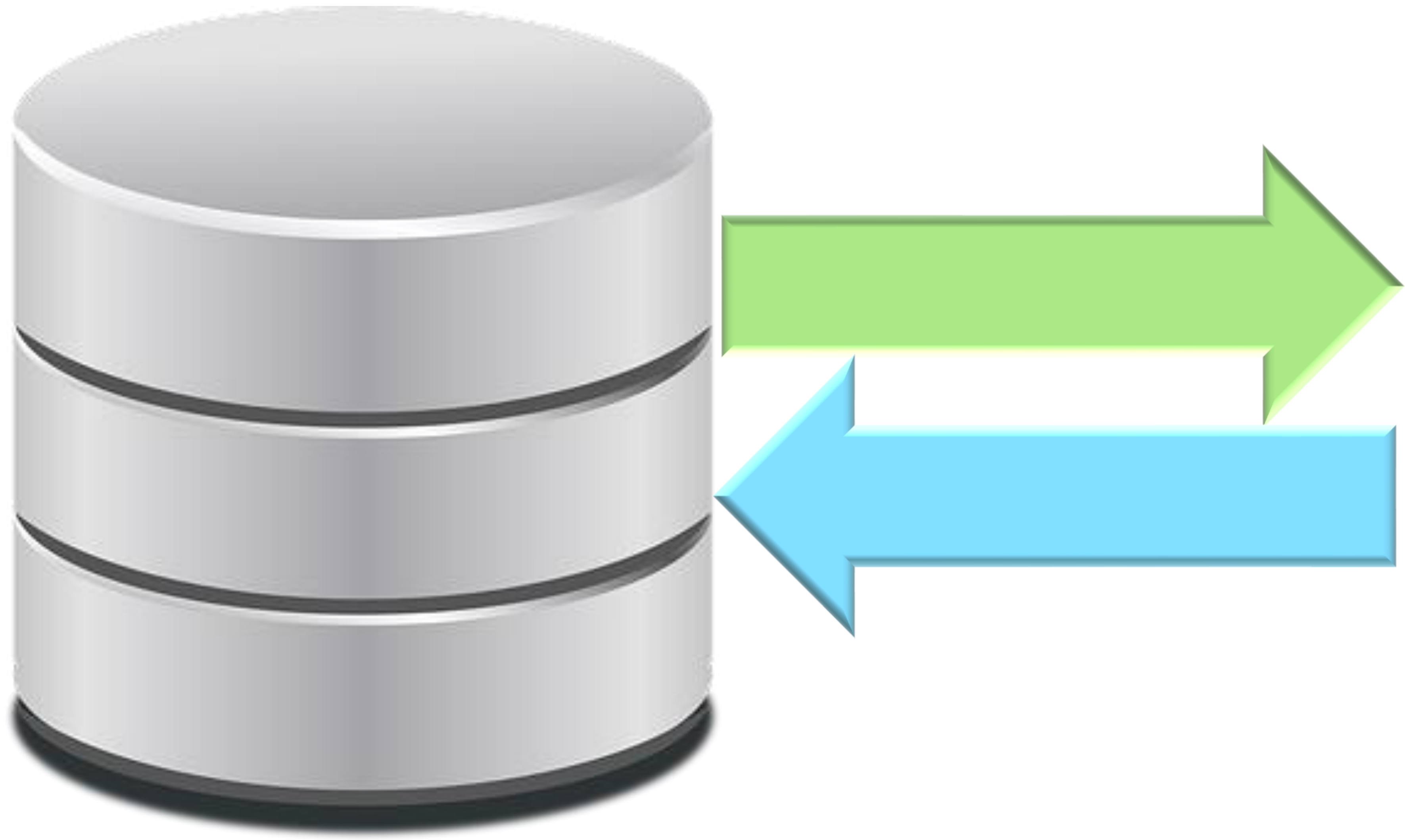
Data Integration: The *REST* of the Story

RESTful Interface Design for Data Integration

Tony Lavinio
Senior Software Architect
October 9, 2019

What are we doing here?!

- Why are we talking about REST?
  - You are providing data! (or want to)
  - You are consuming data! (or want to)
- APIs are everywhere!
  - External APIs
  - Internal APIs



"Use the API" sounds easy – but sometimes it's harder than it has to be.
What can *you* do to be successful?

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

The Story Arc*

Background	*REST Basics.*
Internal APIs vs. Documented APIs	*Dealing with incomplete informa…*
Tools & Debugging	*Postman, Fiddler, AutoREST*
Complex Data & Relationships	*Nested & repeating structures. Joins.*
Designing for Change	*Schema evolution.*
Designing for Performance	*Paging, Filtering, Caching.*
Authentication & Security	*Basic, OAuth2, etc.*
Error Handling	*Bad JSON. HTTP Status.*

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Background

Why Choose REST?

- Fast infrastructure available
  - It leverages browser technologies
  - Compression/Caching
- Simple enough for an intern to implement
  - Protocol simple
  - Incremental value
- All the cool kids are doing it
  - Almost every application has a REST-in or REST-out option
  - Tools widely available and free

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

REST Basics

- Protocol: **HTTP** *or* **HTTPS**
- Actions: **Verbs**
    - GET
    - POST
    - PUT
    - PATCH
    - DELETE
- Payload: **JSON**
    - Less often, **XML**

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

REST Basics – GET

- GET
  - FIND or FOR EACH (ABL)　　　SELECT (SQL)
- GET /customer
  - FOR EACH customer:　　　SELECT * FROM customer;
- GET /customer/key
  - FIND customer WHERE id = key.　　　SELECT customer WHERE id = key;
- GET /customer?name=UFO%20Frisbee
  - FOR EACH customer
    WHERE name = "UFO Frisbee":　　　SELECT customer
    WHERE name = 'UFO Frisbee';

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
PUGCHALLENGE AMERICAS

REST Basics – POST

- POST
  - CREATE (ABL) INSERT (SQL)
  - Requires a JSON payload, like
    { "id":27, "name":"UFO Frisbee", "salesrep":"DKP" }
- POST /customer
  - CREATE customer INSERT INTO customer

```
CREATE customer
ASSIGN
    id = 27
    name = "UFO Frisbee"
    salesrep = "DKP".
```

```
INSERT INTO customer
    (id, name, salesrep)
    VALUES
    (27, 'UFO Frisbee', 'DKP');
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
PUGCHALLENGE AMERICAS

REST Basics – PUT and PATCH

- PUT and PATCH
  - Technically, PUT should reset any unspecified fields to their default
    - A lot like deleting and recreating, with just the values specified
  - Most sites mean PATCH when they implement PUT
    - So we'll pretend PUT means PATCH for the rest of this slide
  - Assignments (ABL) UPDATE (SQL)
  - Requires a JSON payload, like
    { "salesrep":"SLS" }
- PUT /customer/27
  - ABL:

    ```
    FIND customer
        WHERE id = 27.
    customer.salesrep = "SLS".
    ```

    SQL:

    ```
    UPDATE customer
        SET salesrep = 'SLS'
        WHERE id = 27;
    ```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

REST Basics – DELETE

- DELETE
  - DELETE (ABL) DELETE (SQL)
- DELETE /customer/27
  - FIND customer WHERE id = 27. DELETE customer. DELETE customer WHERE id = 27;

```
FIND customer                         DELETE customer
    WHERE id = 27.                        WHERE id = 27;
DELETE customer.
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Internal APIs vs. Documented APIs

What’s what

- Some applications do a really good job of documenting their APIs
- Some tell a good story, but the docs don’t match reality
- Some endpoints, especially those built on internal systems, have no reference material at all, other than “Ask Rita; I think she wrote that.”

```
/** @author Rita */
public class Result
  /**
   * @return
   */
  public Object getValue() {
   return object.calc();
  }

  /**
   * @param object
   */
  public void setValue(Object o) {
   object = o.clone();
  }
}
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
PUGCHALLENGE AMERICAS

Strategies

- Reading documentation
  - And hoping it’s accurate

```
This sentence is false.
```

- Reading source code
  - If it’s available, and if it’s not written in APL

```
{ω /⍨ ~{ω∨≠\ω}ω∊'<>'} txt
```

- Finding Rita
  - “Oh, She resigned last week.”
    “Um, Anyone know if IT has wiped her computer yet?”
- Poking it with a stick
  - Hitting the endpoint with `POSTMAN` and seeing what comes out
- Sampling
  - Using ARC to sample and analyze the structure

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Things to look for in documentation

- Authentication
- Paging
- Parameter semantics
- JSON response format (results aren't always at the root)

```
[
   {"id":8, "name":"Butternut Squash"},
   {"id":23, "name":"Sub Par Golf"},
   {"id":27, "name":"UFO Frisbee"}
]
```

```
{
   "customers":[
      {"id":8, "name":"Butternut Squash"},
      {"id":23, "name":"Sub Par Golf"},
      {"id":27, "name":"UFO Frisbee"}
   ]
}
```

```
{
   "offset":0,
   "count":3,
   "total":3,
   "page":1,
   "pages":1,
   "results":[
      {"id":8, "name":"Butternut Squash"},
      {"id":23, "name":"Sub Par Golf"},
      {"id":27, "name":"UFO Frisbee"}
   ]
}
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

PUGCHALLENGE AMERICAS

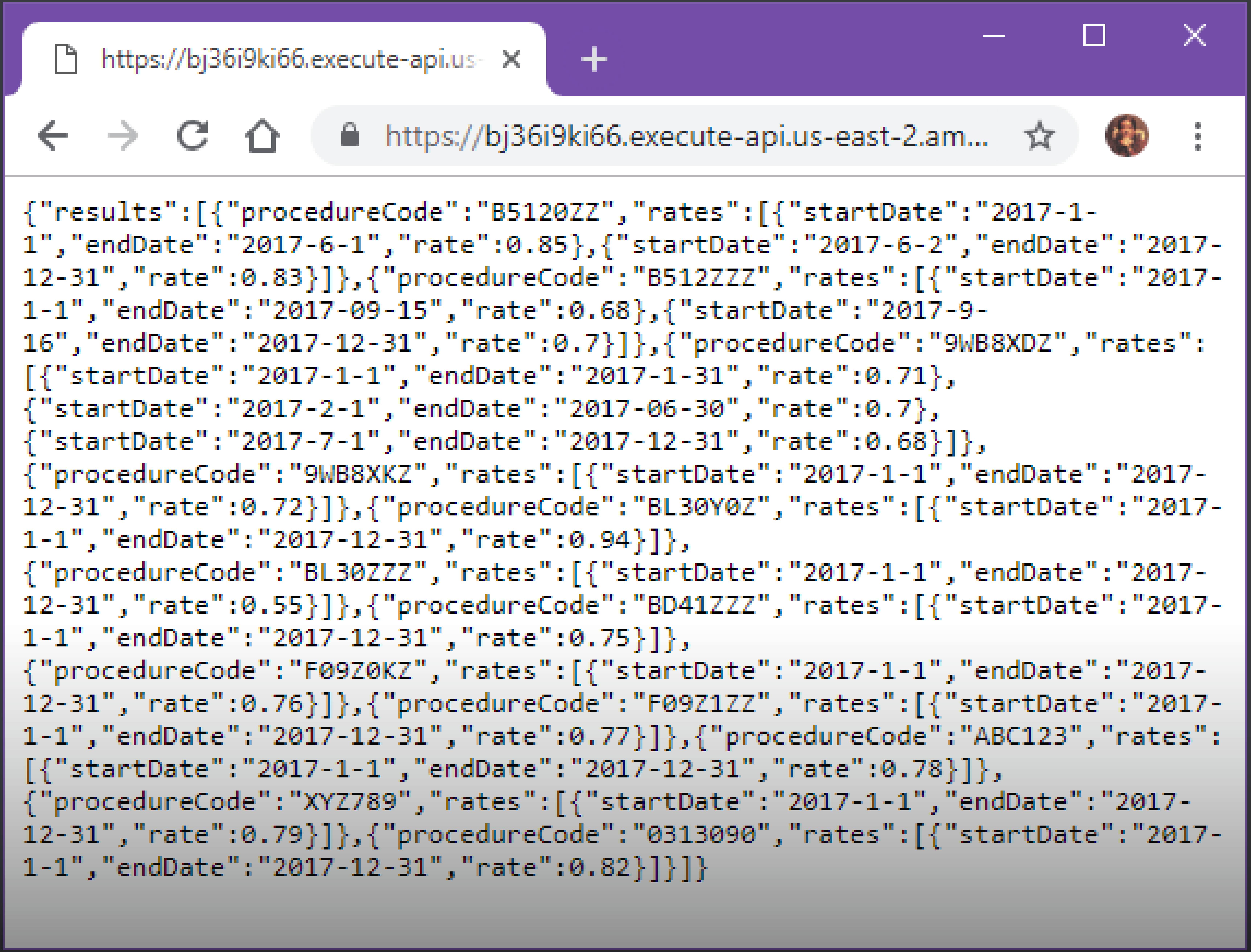
Tools & Debugging

Browser

- Quick way to test GET requests when no special headers or authentication is required
  - But that’s about it

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
PUGCHALLENGE AMERICAS

Inspecting with a Browser (Chrome)


https://bj36i9ki66.execute-api.us-
https://bj36i9ki66.execute-api.us-east-2.am...

```
{"results":[{"procedureCode":"B5120ZZ","rates":[{"startDate":"2017-1-1","endDate":"2017-6-1","rate":0.85},{"startDate":"2017-6-2","endDate":"2017-12-31","rate":0.83}]},{"procedureCode":"B512ZZZ","rates":[{"startDate":"2017-1-1","endDate":"2017-09-15","rate":0.68},{"startDate":"2017-9-16","endDate":"2017-12-31","rate":0.7}]},{"procedureCode":"9WB8XDZ","rates":[{"startDate":"2017-1-1","endDate":"2017-1-31","rate":0.71},{"startDate":"2017-2-1","endDate":"2017-06-30","rate":0.7},{"startDate":"2017-7-1","endDate":"2017-12-31","rate":0.68}]},{"procedureCode":"9WB8XKZ","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.72}]},{"procedureCode":"BL30Y0Z","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.94}]},{"procedureCode":"BL30ZZZ","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.55}]},{"procedureCode":"BD41ZZZ","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.75}]},{"procedureCode":"F09Z0KZ","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.76}]},{"procedureCode":"F09Z1ZZ","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.77}]},{"procedureCode":"ABC123","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.78}]},{"procedureCode":"XYZ789","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.79}]},{"procedureCode":"0313090","rates":[{"startDate":"2017-1-1","endDate":"2017-12-31","rate":0.82}]}]}
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

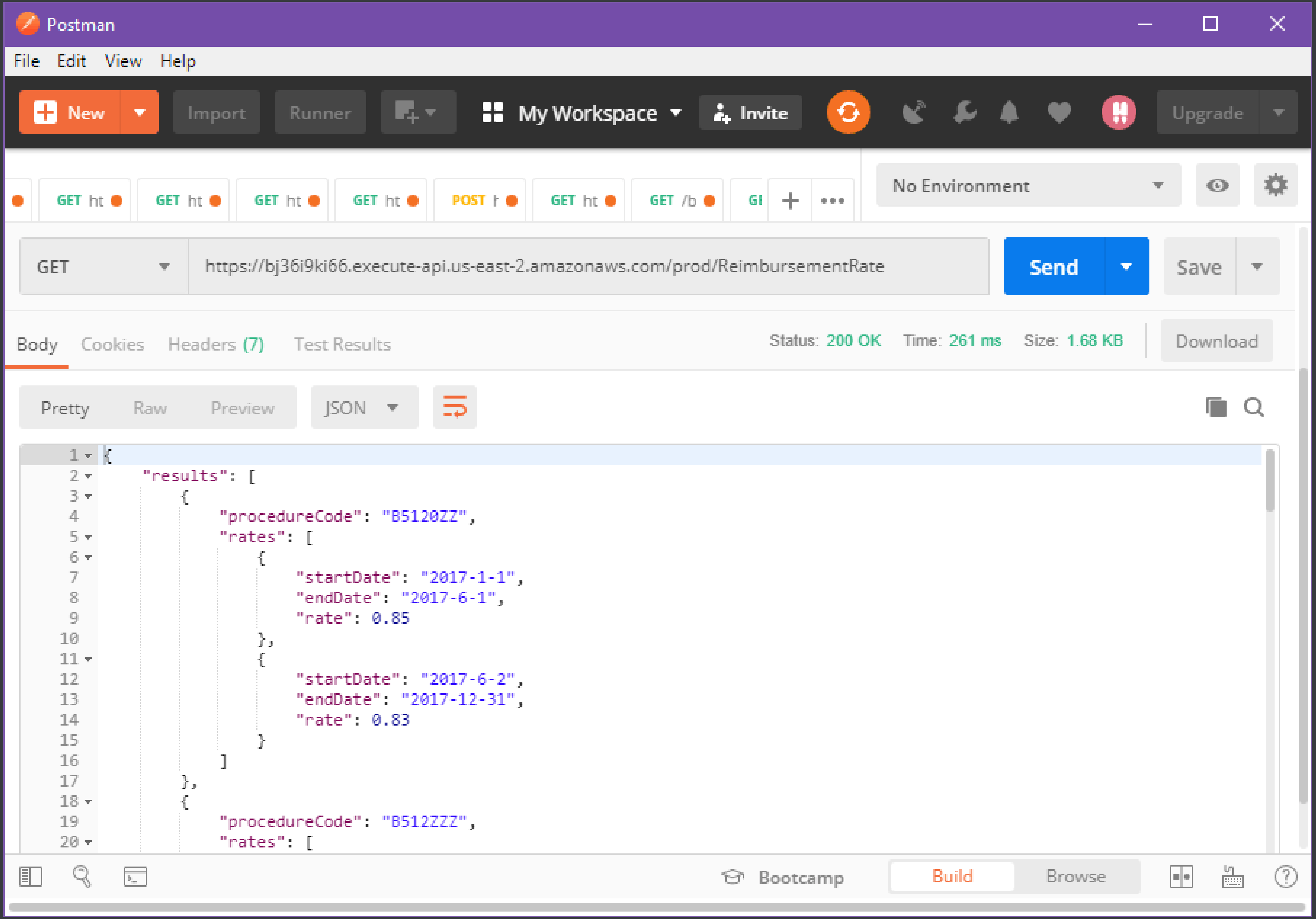
Postman

- Issue request
- Inspect results
- Modify headers
- Experiment with authentication

POSTMAN

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Inspecting with Postman (showing body)



GET https://bj36i9ki66.execute-api.us-east-2.amazonaws.com/prod/ReimbursementRate

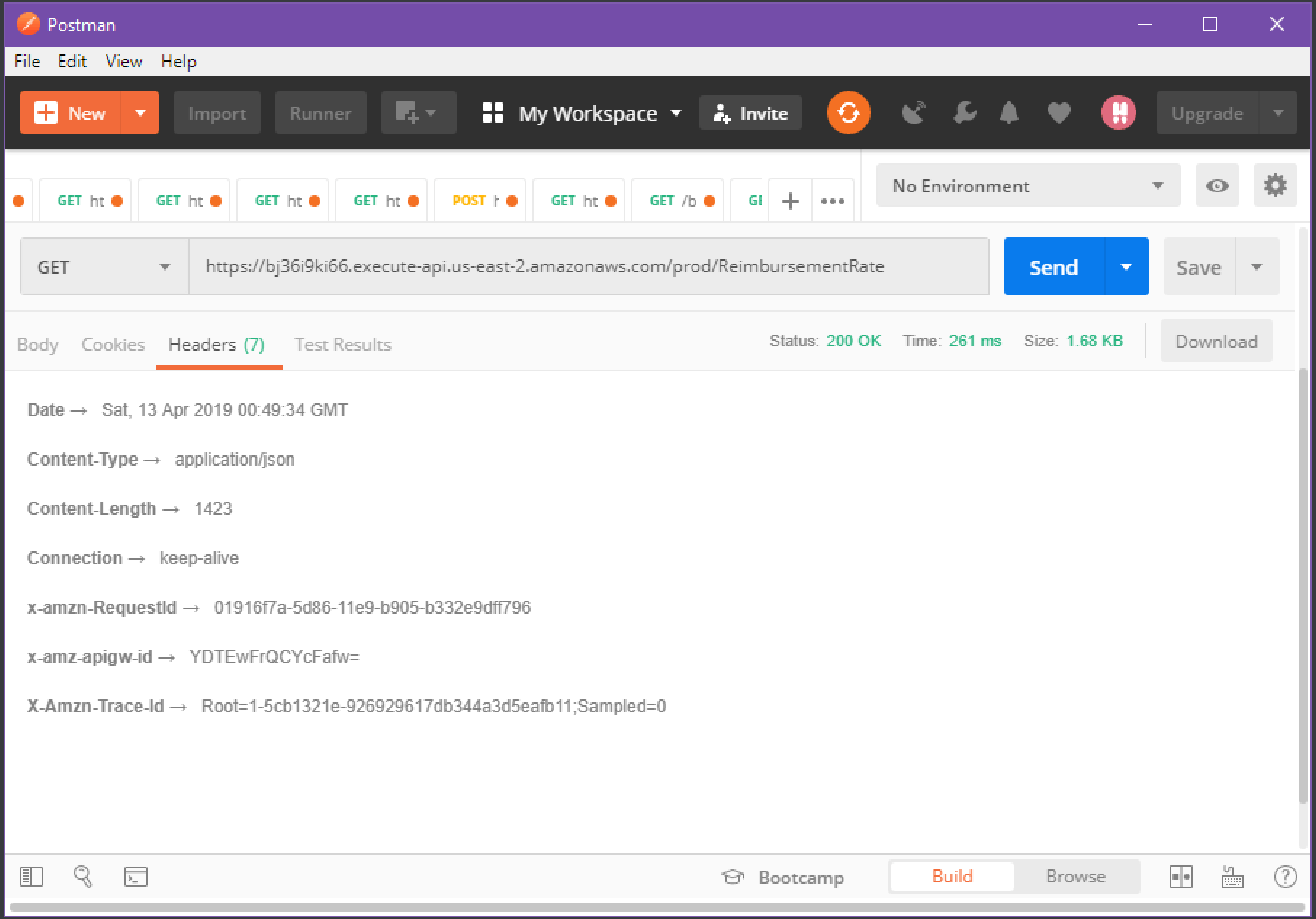
Status: 200 OK Time: 261 ms Size: 1.68 KB

```
{
    "results": [
        {
            "procedureCode": "B5120ZZ",
            "rates": [
                {
                    "startDate": "2017-1-1",
                    "endDate": "2017-6-1",
                    "rate": 0.85
                },
                {
                    "startDate": "2017-6-2",
                    "endDate": "2017-12-31",
                    "rate": 0.83
                }
            ]
        },
        {
            "procedureCode": "B512ZZZ",
            "rates": [
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Inspecting with Postman (showing headers)



GET https://bj36i9ki66.execute-api.us-east-2.amazonaws.com/prod/ReimbursementRate

Status: 200 OK Time: 261 ms Size: 1.68 KB

```
Date → Sat, 13 Apr 2019 00:49:34 GMT
Content-Type → application/json
Content-Length → 1423
Connection → keep-alive
x-amzn-RequestId → 01916f7a-5d86-11e9-b905-b332e9dff796
x-amz-apigw-id → YDTEwFrQCYcFafw=
X-Amzn-Trace-Id → Root=1-5cb1321e-926929617db344a3d5eafb11;Sampled=0
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Fiddler

Fiddler

- Use as a proxy to watch stuff happen
  - Even **HTTPS** interception
- Intercept (and tweak) data in stream

https://www.telerik.com/download/fiddler

https://docs.telerik.com/fiddler/Configure-Fiddler/Tasks/ConfigureFiddler

https://docs.telerik.com/fiddler/Configure-Fiddler/Tasks/ConfigureJavaApp

- If using HTTPS, here's how to configure:
  https://stackoverflow.com/questions/8549749/how-to-capture-https-with-fiddler-in-java

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Import certificate from Fiddler; Inform JVM

- <*java*>\bin\keytool.exe -importcert -trustcacerts
  -file **<*desktop*>\FiddlerRoot.cer**
  -keystore **<*directory*>/FiddlerKeystore** -alias **Fiddler**
- Enter keystore password: **password**
- Trust this certificate? [no] **yes**

- -Dhttps.proxyHost=**127.0.0.1**
- -Dhttps.proxyPort=**8888**
- -Djavax.net.ssl.trustStore=**<*directory*>/FiddlerKeystore**
- -Djavax.net.ssl.trustStorePassword=**password**

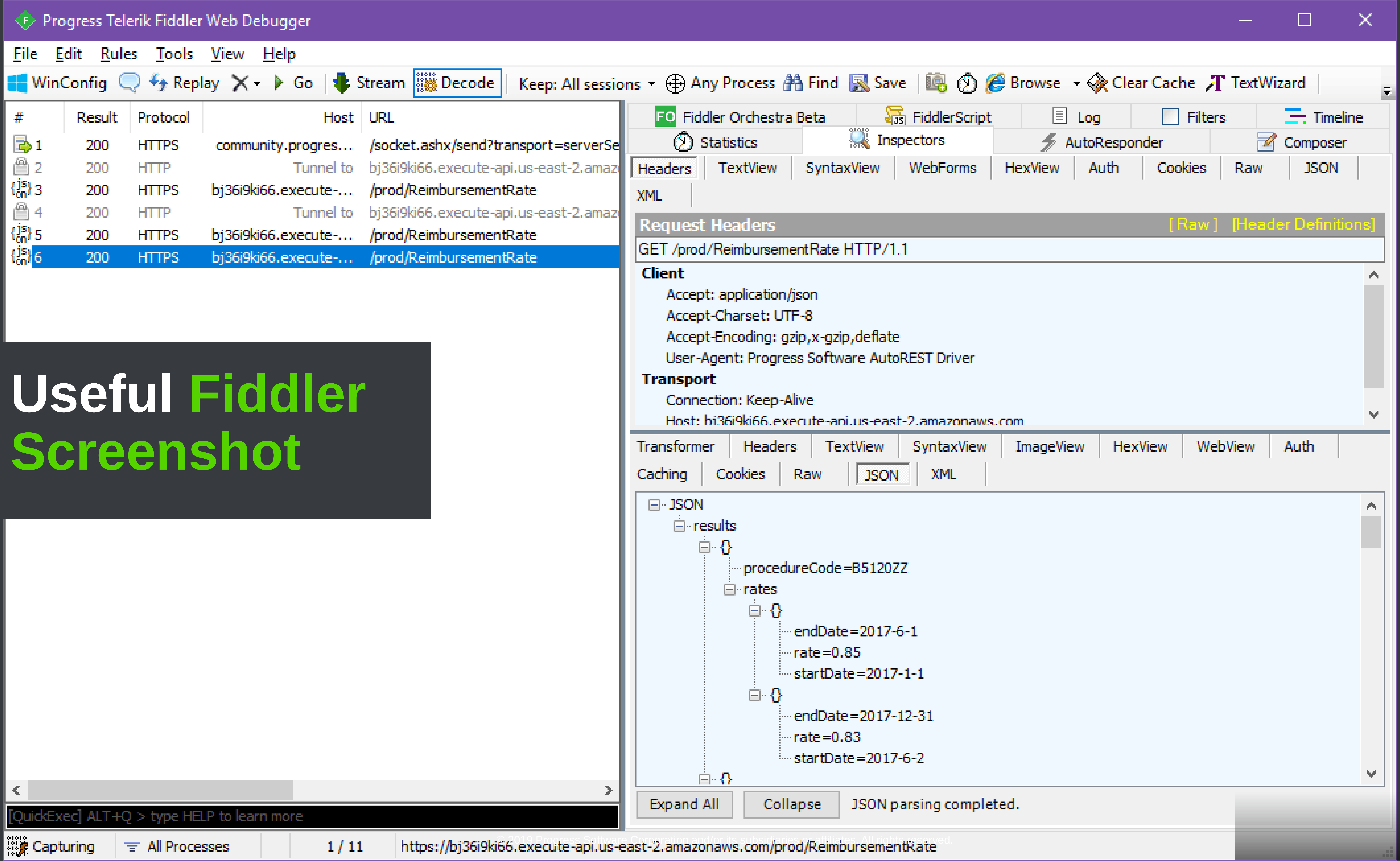
© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Fiddler Screenshot

Fiddler on the Roof, 1971—8 Academy Awards,
4 Golden Globes, 3 BAFTAs, etc. Not software.

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Useful Fiddler Screenshot


Progress Telerik Fiddler Web Debugger
File Edit Rules Tools View Help
WinConfig Replay Go Stream Decode Keep: All sessions Any Process Find Save Browse Clear Cache TextWizard
Fiddler Orchestra Beta FiddlerScript Log Filters Timeline
Statistics Inspectors AutoResponder Composer
Headers TextView SyntaxView WebForms HexView Auth Cookies Raw JSON XML
Transformer Headers TextView SyntaxView ImageView HexView WebView Auth
Caching Cookies Raw JSON XML
Expand All Collapse

#	Result	Protocol	Host	URL
1	200	HTTPS	community.progres…	/socket.ashx/send?transport=serverSe
2	200	HTTP	Tunnel to	bj36i9ki66.execute-api.us-east-2.amaz
3	200	HTTPS	bj36i9ki66.execute-…	/prod/ReimbursementRate
4	200	HTTP	Tunnel to	bj36i9ki66.execute-api.us-east-2.amaz
5	200	HTTPS	bj36i9ki66.execute-…	/prod/ReimbursementRate
6	200	HTTPS	bj36i9ki66.execute-…	/prod/ReimbursementRate

```
Request Headers                                [ Raw ]  [Header Definitions]
GET /prod/ReimbursementRate HTTP/1.1
Client
    Accept: application/json
    Accept-Charset: UTF-8
    Accept-Encoding: gzip,x-gzip,deflate
    User-Agent: Progress Software AutoREST Driver
Transport
    Connection: Keep-Alive
    Host: bj36i9ki66.execute-api.us-east-2.amazonaws.com
```

```
JSON
  results
    {}
      procedureCode=B5120ZZ
      rates
        {}
          endDate=2017-6-1
          rate=0.85
          startDate=2017-1-1
        {}
          endDate=2017-12-31
          rate=0.83
          startDate=2017-6-2
    {}
```

JSON parsing completed.

[QuickExec] ALT+Q > type HELP to learn more

Capturing   All Processes   1 / 11   https://bj36i9ki66.execute-api.us-east-2.amazonaws.com/prod/ReimbursementRate

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Complex Data & Relationships

One of these things is not like the others

Surface	Length	Width	Legs
Oak	42"	72"	5
Maple	36"	60"	4
Teak	30"	54"	2
Ebony	80cm	160cm	2

Muppet	Voice	Image
Grover	Frank Oz	
Kermit	Jim Henson	

Team	Zone	Standing
Brazil	CONMEBOL	3
Chile	CONMEBOL	14
Colombia	CONMEBOL	8
Germany	UEFA	2
Mexico	CONCACAF	20
USA	CONCACAF	13

```
{
  _id: 1,
  tournament: "world cup",
  years:[
    {
      year: 2014,
      location: "Brazil",
      teams:[
        "Brazil", "Chile", "Colombia",
        "Germany", "Mexico", "USA"
      ]
    },
    …
  ]
}
```

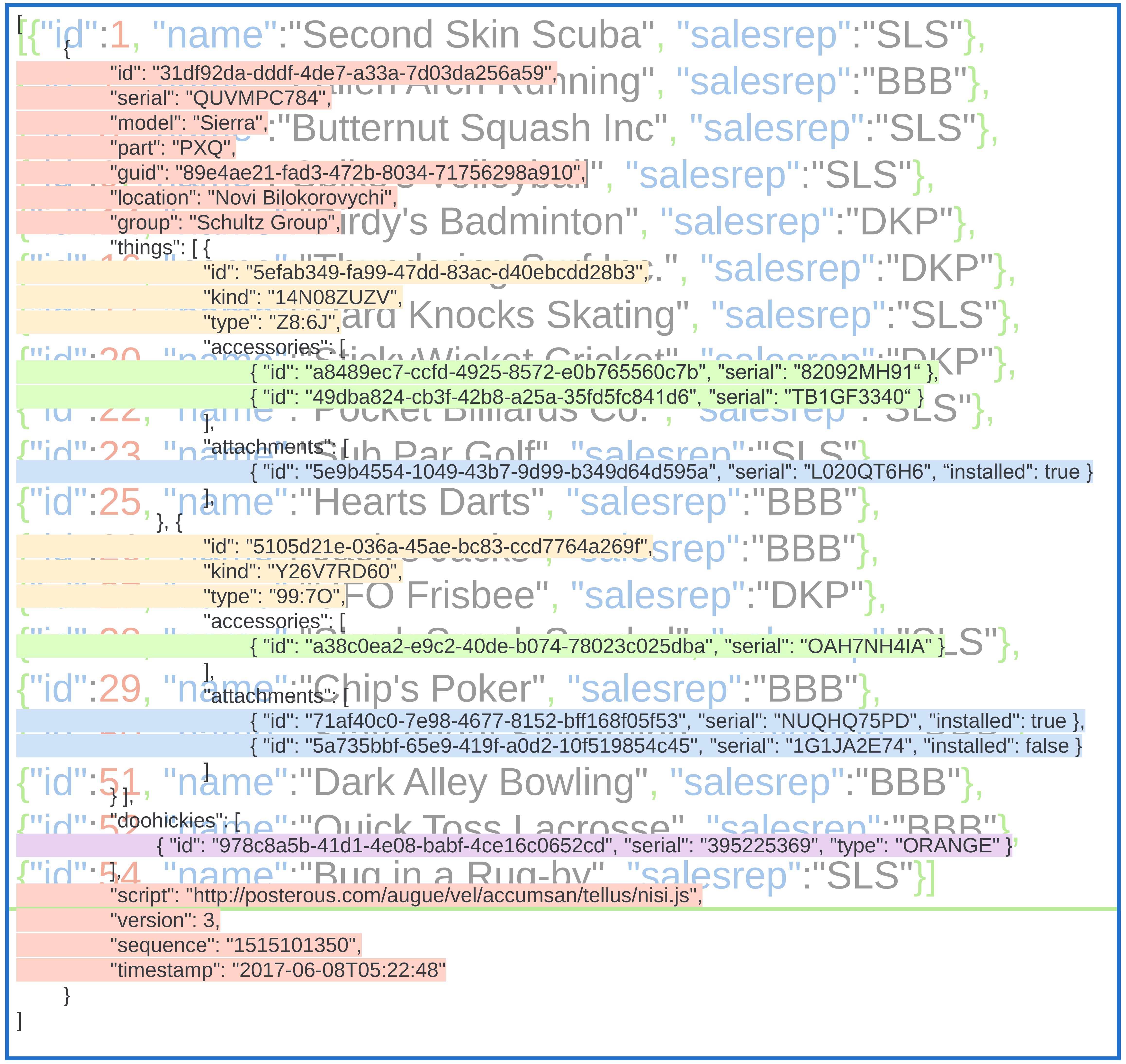
© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

For many APIs, you’ll get back rectangles of data

- Often for APIs that simply front some other reporting engine

But for other APIs, you’ll get really complex structures



```
[{"id":1, "name":"Second Skin Scuba", "salesrep":"SLS"},
[illegible]nning", "salesrep":"BBB"},
[illegible]:"Butternut Squash Inc", "salesrep":"SLS"},
[illegible]", "salesrep":"SLS"},
[illegible]rdy's Badminton", "salesrep":"DKP"},
[illegible]", "salesrep":"DKP"},
[illegible]ard Knocks Skating", "salesrep":"SLS"},
{"id":20, "name":"Sticky Wicket Cricket", "salesrep":"DKP"},
{"id":22, "name":"Pocket Billiards Co.", "salesrep":"SLS"},
{"id":23, "name":"[illegible]b Par Golf", "salesrep":"SLS"},
{"id":25, "name":"Hearts Darts", "salesrep":"BBB"},
[illegible]srep":"BBB"},
[illegible]FO Frisbee", "salesrep":"DKP"},
[illegible]"SLS"},
{"id":29, "name":"Chip's Poker", "salesrep":"BBB"},
{"id":51, "name":"Dark Alley Bowling", "salesrep":"BBB"},
{"id":52, "name":"Quick Toss Lacrosse", "salesrep":"BBB"},
{"id":54, "name":"Bug in a Rug-by", "salesrep":"SLS"}]
```

```
[
    {
        "id": "31df92da-dddf-4de7-a33a-7d03da256a59",
        "serial": "QUVMPC784",
        "model": "Sierra",
        "part": "PXQ",
        "guid": "89e4ae21-fad3-472b-8034-71756298a910",
        "location": "Novi Bilokorovychi",
        "group": "Schultz Group",
        "things": [ {
            "id": "5efab349-fa99-47dd-83ac-d40ebcdd28b3",
            "kind": "14N08ZUZV",
            "type": "Z8:6J",
            "accessories": [
                { "id": "a8489ec7-ccfd-4925-8572-e0b765560c7b", "serial": "82092MH91“ },
                { "id": "49dba824-cb3f-42b8-a25a-35fd5fc841d6", "serial": "TB1GF3340“ }
            ],
            "attachments": [
                { "id": "5e9b4554-1049-43b7-9d99-b349d64d595a", "serial": "L020QT6H6", “installed": true }
            ]
        }, {
            "id": "5105d21e-036a-45ae-bc83-ccd7764a269f",
            "kind": "Y26V7RD60",
            "type": "99:7O",
            "accessories": [
                { "id": "a38c0ea2-e9c2-40de-b074-78023c025dba", "serial": "OAH7NH4IA" }
            ],
            "attachments": [
                { "id": "71af40c0-7e98-4677-8152-bff168f05f53", "serial": "NUQHQ75PD", "installed": true },
                { "id": "5a735bbf-65e9-419f-a0d2-10f519854c45", "serial": "1G1JA2E74", "installed": false }
            ]
        } ],
        "doohickies": [
            { "id": "978c8a5b-41d1-4e08-babf-4ce16c0652cd", "serial": "395225369", "type": "ORANGE" }
        ],
        "script": "http://posterous.com/augue/vel/accumsan/tellus/nisi.js",
        "version": 3,
        "sequence": "1515101350",
        "timestamp": "2017-06-08T05:22:48"
    }
]
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Complex Data Models – The Why

- Model relationship semantics
  - An order has items, after all
- Keep related data together
  - Prevents multiple round trips
  - Helps data consistency
- Report on query/execution metadata
  - Did the query succeed?
  - Is this a partial result?

```
{    "meta":{
         "Status":"success",
         "exectime":873,
         "records":5937984
     },
     "pagenumber":4,
     "morePages" :true,
     "results": [
          {
               "Address":"11 Perkins St",
               "City":"Boston",
               "Cust-num":4,
               "Order-num":2,
               "Name":"Pedal Power Cycles",
               "items": [{
                         "Item-num":3,
                         "Price":2.55,
                         "Qty":4,
                    }, {
                         "Item-num":9,
                         "Price":75,
                         "Qty":2,
                    }
               ]
          }
     ]
}
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Normalization

```
https://myservice/orders/2
{
    "Order-num":2,
    "Odate":"1990-09-06",
    "customer": {
        "Cust-num":4,
        "Name":"Pedal Power Cycles",
        "Address":"11 Perkins St",
        "City":"Boston"
    },
    "orderlines": [
        {
            "Item-num":3,
            "Price":2.55,
            "Qty":4
        }, {
            "Item-num":9,
            "Price":75,
            "Qty":2
        }, {
            "Item-num":19,
            "Price":19.95,
            "Qty":17
        }
    ]
}
```

```
TABLE: orders
Order-num: integer, key
Odate: Date
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Normalization→Flattening (Objects)

```
https://myservice/orders/2
{
    "Order-num":2,
    "Odate":"1990-09-06",
    "customer": {
        "Cust-num":4,
        "Name":"Pedal Power Cycles",
        "Address":"11 Perkins St",
        "City":"Boston"
    },
    "orderlines": [
        {
            "Item-num":3,
            "Price":2.55,
            "Qty":4
        }, {
            "Item-num":9,
            "Price":75,
            "Qty":2
        }, {
            "Item-num":19,
            "Price":19.95,
            "Qty":17
        }
    ]
}
```

```
TABLE: orders
Order-num: integer, key
Odate: Date
Cust-num: integer
Name: varchar(64)
Address: varchar(64)
City: varchar(64)
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Normalization→Arrays (Lists)

```
https://myservice/orders/2
{
    "Order-num":2,
    "Odate":"1990-09-06",
    "customer": {
        "Cust-num":4,
        "Name":"Pedal Power Cycles",
        "Address":"11 Perkins St",
        "City":"Boston"
    },
    "orderlines": [
        {
            "Item-num":3,
            "Price":2.55,
            "Qty":4
        }, {
            "Item-num":9,
            "Price":75,
            "Qty":2
        }, {
            "Item-num":19,
            "Price":19.95,
            "Qty":17
        }
    ]
}
```

TABLE: orders

Order-num (KEY)	Odate	Cust-num	Name	Address	City
2	1990-09-06	4	Pedal Power Cycles	11 Perkins St	Boston

TABLE: orderlines

Order-num (KEY)	Position (KEY)	Item-num	Price	Qty
2	1	3	2.55	4
2	2	9	75.00	2
2	3	19	19.95	17

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Relationships – JOIN in a single endpoint

```
https://myservice/orders/
{
    "Order-num":2,
    "Odate":"1990-09-06",
    "customer": {
        "Cust-num":4,
        "Name":"Pedal Power Cycles",
        "Address":"11 Perkins St",
        "City":"Boston"
    },
    "orderlines": [
        {
            "Item-num":3,
            "Price":2.55,
            "Qty":4
        }, {
            "Item-num":9,
            "Price":75,
            "Qty":2
        }, {
            "Item-num":19,
            "Price":19.95,
            "Qty":17
        }
    ]
}
```

TABLE: orders

Order-num (KEY)	Odate	Cust-num	Name	Address	City
2	1990-09-06	4	Pedal Power Cycles	11 Perkins St	Boston

TABLE: orderlines

Order-num (KEY)	Position (KEY)	Item-num	Price	Qty
2	1	3	2.55	4
2	2	9	75.00	2
2	3	19	19.95	17

SELECT orders.Name, orderlines.Price, orderlines.Qty

FROM orders INNER JOIN orderlines

ON orders.Order-num = orderlines.order-num

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Relationships – JOIN across endpoints

```
https://myservice/orders/
{
    "Order-num":2,
    "Odate":"1990-09-06",
    "customer": {
        "Cust-num":4,
        "Name":"Pedal Power Cycles",
        "Address":"11 Perkins St",
        "City":"Boston"
    },
    "orderlines": [
        {
            "Item-num":3,
            "Price":2.55,
            "Qty":4
        }, {
            "Item-num":9,
            "Price":75,
            "Qty":2
        }, {
            "Item-num":19,
            "Price":19.95,
            "Qty":17
        }
    ]
}
```

```
https://myservice/customers

{
    "Contact":"Alicia Primes",
    "Curr-bal":520.77,
    "Cust-num":4,
    "Discount":2,
    "Max-credit":416,
    "Name":"Pedal Power Cycles",
    "Phone":"6172456969",
    "Sales-rep":"BBB",
    "St":"MA",
    "Ytd-sls":4713.87
}, …
```

What can we do?

- Pull everything back (is next week okay?)
- Optimizations
  - Algorithms
  - Filters

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Designing Your Data Model for Integration

- Domain specific design is okay
- Consider future reporting needs
  - Either augment existing data…
  - …or provide an alternative endpoint
- Avoid limitless and over-generalized nesting

```
{
    "DATA" : {
        "RECORDS" : {},
        "ArrayIndex" : 1
    },
    "PEOPLE" : {
        "PERSON" : [
            {
                "DATA" : {
                    "RECORDS" : {},
                    "ArrayIndex" : 1
                },
                "PARTY" : {
                    "APPLICANT" : {
                        "BACKGROUND" : {
                            "CREDIT_HISTORY" : [
                                {
                                    "DATA" : {
                                        "HISTORY" : {
                                            "GOOD_STUFF" : [
                                                {
                                                    "YEAR" : 2007,
                                                    "SCORE" : "1"
                                                }
                                            ],
                                            "BAD_STUFF" : [
                                                {
                                                    "YEAR" : 2009,
                                                    "SCORE" : 37
                                                }
                                            ]
                                        },
                                        "ArrayIndex" : 1
                                    },
                                    "RULES_APPLIED" : "1",
                                    "DESCRIPTION" : "Success!"
                                },
                                {
                                    "DATA" : {
                                        "HISTORY" : {
                                            "GOOD_STUFF" : [
                                                {
                                                    "YEAR" : 2007,
                                                    "SCORE" : "1"
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

PUGCHALLENGE
AMERICAS

Designing for Change

Schema Evolution – as a Sender

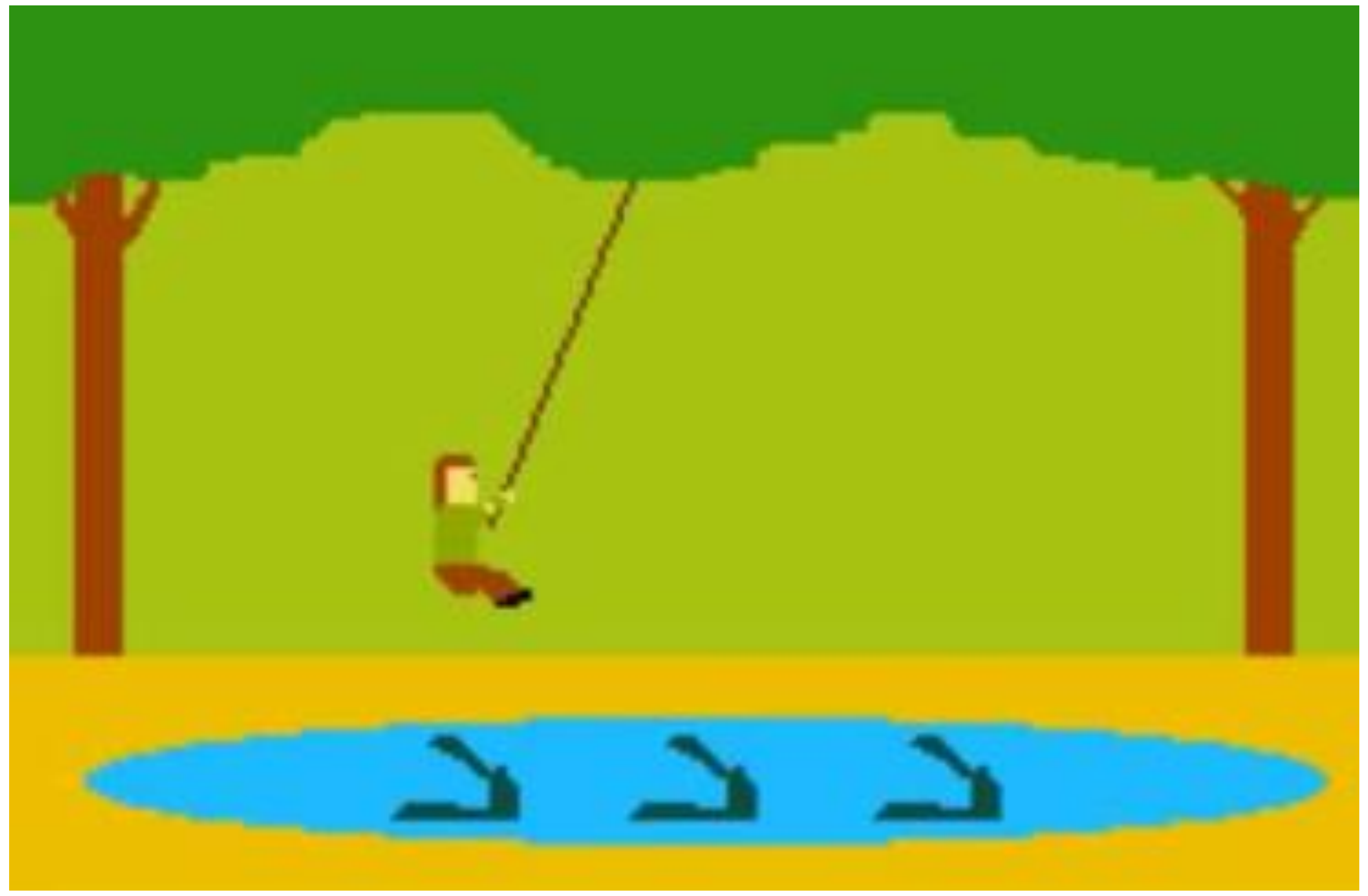
- If you're sending the payload, how do you protect clients from future changes?

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Schema Evolution – Beware the following changes

- Changing from scalar to array instances
- Using the same name for a field when changing its type
  - Especially simple to complex types
  - Tools' implementations will lag behind your API
- Big structural changes will cause problems for integrators
- Date and time representations
  - Yes: ISO. Yes: epoch. No: MMDDYY
- Floats/Numerics – some JSON systems put everything into a double.
  - Often see numbers quoted for this reason
  - What looks like an `int` may be a `long`
- **Hint: Version your APIs!**



© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

PUGCHALLENGE
AMERICAS

Designing for Performance

Large Result Sets

- The good news:
  Your entire database can now be queried with REST
- The bad news:
  All 15TB comes back

Like drinking from a firehose

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Paging

- Common pattern is **paging**
  - Page size/Row offset
  - Page size/Page number
  - Cursor-driven

  - Example from SpaceX →

GET All Rockets

https://api.spacexdata.com/v3/rockets

Returns all rockets

Param	Type	Description
limit	integer	Limit results returned, defaults to all documents returned
offset	integer	Offset or skip results from the beginning of the query

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Paging and Stability

- Common problem is **stability**
  - REST ain't ACID
    - No "consistent read" mode
  - If the source is active, inserted or deleted rows can leave "holes"

Tony

```
> GET Offset 0, Limit 1000
…returns Keys 0-999


> GET Offset 1000, Limit 1000
…returns keys 1001-2000 (!)
```

Key	Data
0	
…	
~~999~~	🟥
1000	
1001	
…	
1999	
2000	

Phil

```
> Delete where key = 999
```

(laughs maniacally)

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Filtering

- How to design filtering so that it’s easy for others to use
- The simpler, the better
  - Everybody wants to introduce their own query language
  - Resist the urge!
  - Do you really need all those operators?
  - If so, consider **OData**. *Why reinvent the wheel?*
    - .Net → NuGet Install-Package Microsoft.AspNet.Odata
    - Java → https://olingo.apache.org/
    - Standard → https://www.odata.org/ “OData – the best way to REST”

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

80/20 Rule Filtering

- Most of the time, people just want to use some simple filters
- Query parameters
  - Get all data
    `https://server/endpoint`
  - Use one filter: Get all data on “little” items
    `https://server/endpoint?size=little`
  - Use a different filter: Get all data on “green” items
    `https://server/endpoint?color=green`
  - Combine filters: Get all data on Martians
    `https://server/endpoint?size=little&color=green&object=men`

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Caching

- Congratulations!
- Because you used REST and not GraphQL or RPC, you can take advantage of *caching*
- HTTP Caching only works for HTTP GET requests

- HTTP Caching Headers:
  - **Cache-Control**
  - **Expires**
  - **ETag** and **If-None-Match**
  - **Apache**, **nginx**, **IIS**, **Tomcat**, etc.—most web servers support caching

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Compression – Just Do It

- “gzip” response compression
  - Alternatives?
    - “deflate” subsumed by “gzip”
    - “brotli” slightly better; not as well supported, source at https://github.com/google/brotli
    - “identity” for small payloads? Nah, best to just gzip everything
  - Easy to implement
    - GZipInputStream/GZipOutputStream (Java)
    - GZipStream (.Net)
    - zlib (C/C++)
  - Minimal performance impact (<5%)
  - Transparent to user

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

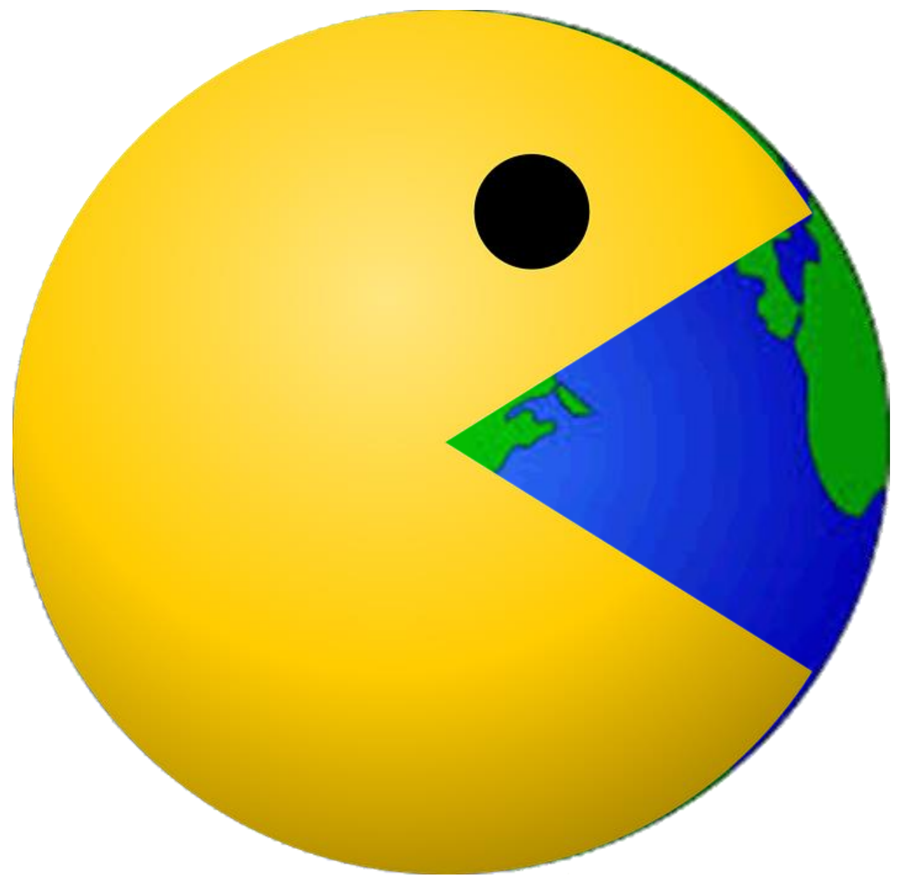
PUGCHALLENGE AMERICAS

PUGCHALLENGE AMERICAS

Authentication & Security

Authentication

- *The good news:*
  You've put on a REST interface, so now the whole world can see your data
- *The bad news:*
  The whole world can see your data



© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Recommended Authentication

- None
- Basic
- ~~Digest~~
- ~~OAuth 1~~
- OAuth 2
- SSO
  - Kerberos
  - Active Directory
- ~~Other~~

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

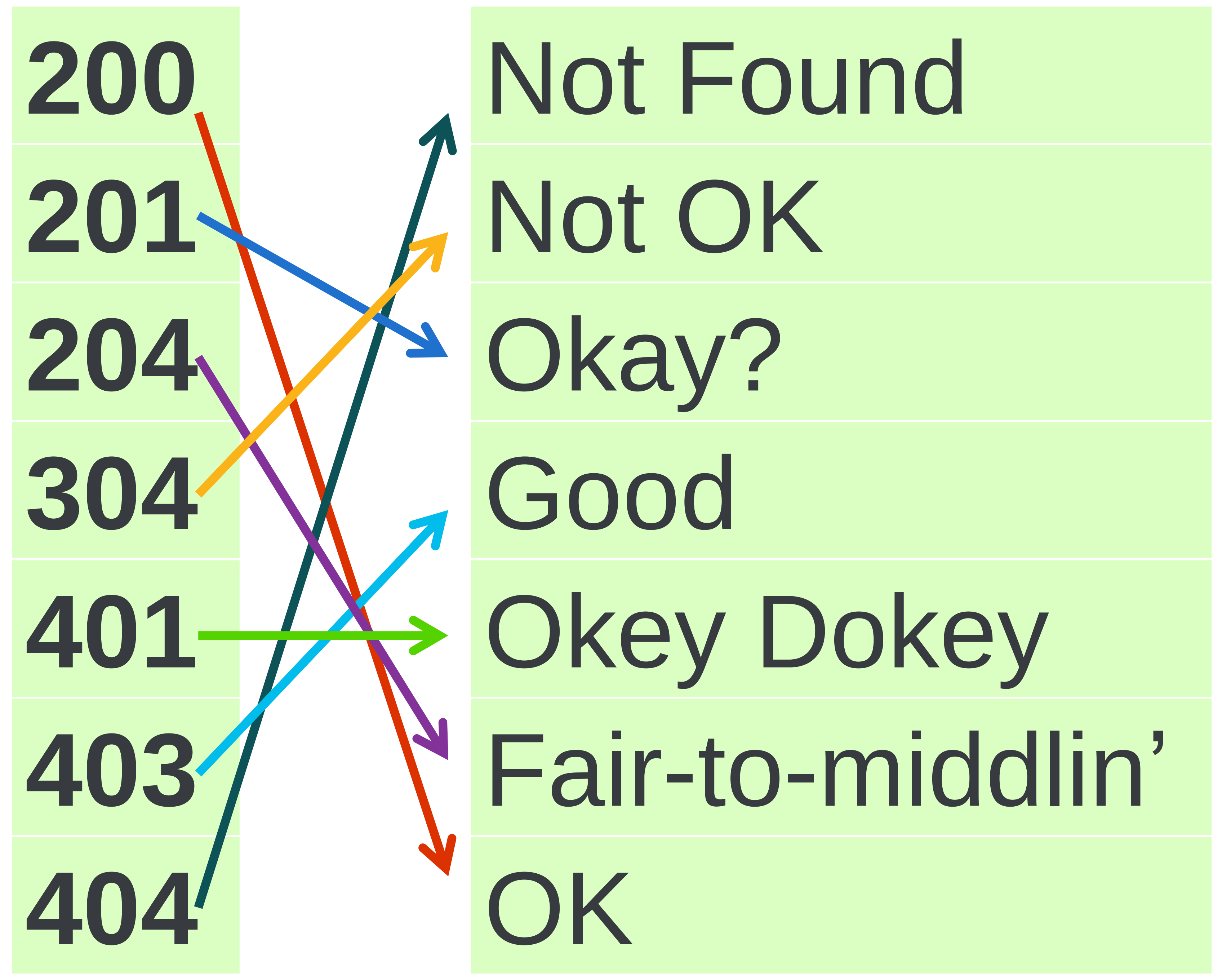
PUGCHALLENGE AMERICAS

PUGCHALLENGE AMERICAS

Error Handling

HTTP Status Codes

- Everything is not OK
  (and that's okay!)
  - Use the HTTP status codes they way they are meant to be used
- Retries
  - WSRetry
- As a client, what do you do with errors when you get them (and how do you decide?)



© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Say What You Mean, and Mean What You Say

- 200 and friends – Everything is fine
  - `200 OK` – okay, and something is in payload
  - `204 No Content` – okay, but no payload
    - Often used for DELETE
- 300 and friends – Who moved my cheese?
  - `301 Moved Permanently` – I've moved over there
  - `304 Not Modified`
    - Used in caching to mean "I already sent you this"
- 400 and friends – caller did something wrong
  - `401 Unauthorized` – need to authenticate (perhaps token expired)
  - `403 Forbidden` – don't touch that
  - `404 Not Found` – it's not there (not necessarily an error!)
  - `429 Too Many Requests` – you're going too fast for me
- 500 and friends – server is busted
  - It's not your fault, it's theirs
  - If at first you don't succeed, give up and come back later – retrying won't help

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

404 Is Tricky

- /service/custoxer/4
  - Since we don't have custoxers in our database; this is properly a `404 Not Found` error

- /service/customer/4
  - We have customers, but not a customer 4, so this also is a `404 Not Found`, but a 404 meaning 'you're in the right place, but nobody is home'

- /service/customer?id=4
  - If there is no customer 4, then the query will succeed in returning an set of zero rows, so the result should be a `200 OK`

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

Bad JSON

- Missing commas

```
[
    {"id":8, "name":"Butternut Squash Inc"}
    {"id":23, "name":"Sub Par Golf"}
    {"id":27, "name":"UFO Frisbee"}
]
```

- Missing escapes

```
[
    {"id":26, "name":"Jack's  "Jumpin' " Jacks"},
]
```

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

PUGCHALLENGE AMERICAS

In conclusion,
Get all the REST you can.

—the Progress database driver developers ;)

© 2018 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

Progress®