

DIY APIs

A Do-It-Yourself Approach To
Creating Custom Read-Only APIs
Using Sierra SQL and PHP

Ray Voelker
University of Dayton Libraries



Question ...

Has anyone started to use the
SQL features of Sierra?

Overview

1. What's an API?
2. DIY API Examples:
 1. Study Room Availability
 2. Inventory Barcode Scanning into Google Sheets
 3. New Books List
3. Conclusions / Answer Some Questions

What is an API?

In computer programming ...

Application Programming Interface

... A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together.

source:

https://en.wikipedia.org/wiki/Application_programming_interface

What is a RESTful API?

Representational State Transfer is a software architecture style for building scalable web services.

...

communicate over the Hypertext Transfer Protocol with the same HTTP verbs (GET, POST, PUT, DELETE, etc.) which web browsers use to retrieve web pages and to send data to remote servers.

source:

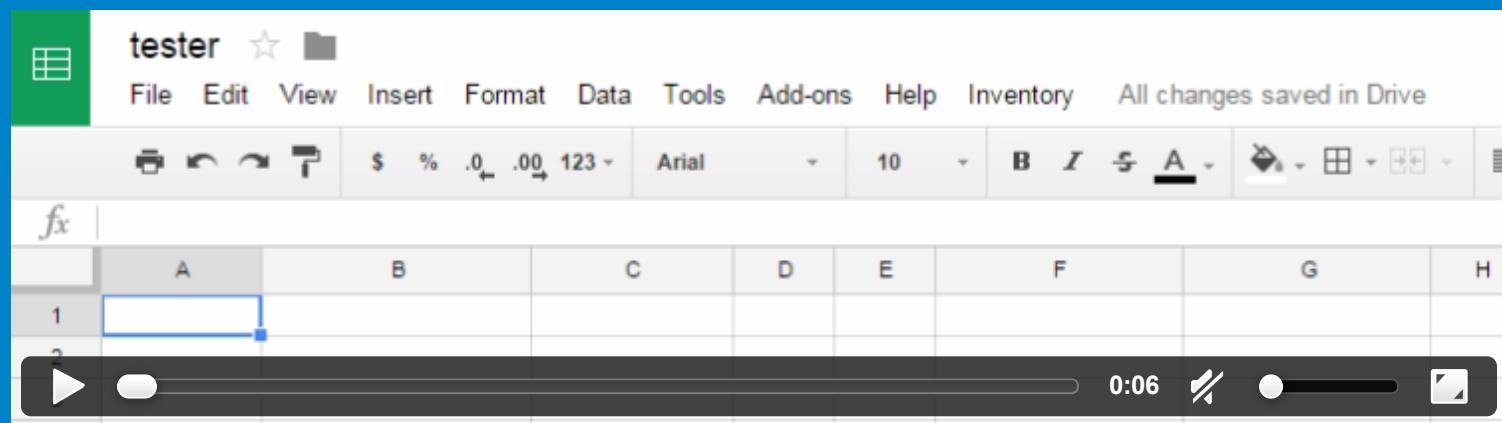
https://en.wikipedia.org/wiki/Representational_state_transfer

What is it really?!

A technique using HTTP requests to get data out of the Sierra Database quickly, and easily. The data we get out is in a format that can be readily used by a wide range of *other applications*.

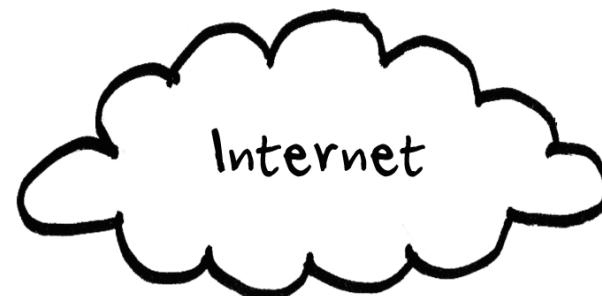
[http://library2.udayton.edu/api/getInventoryData/item_barcode
barcode=R701498024](http://library2.udayton.edu/api/getInventoryData/item_barcode?barcode=R701498024)

```
{  
    "call_number_norm": "PS 3553 R48 S6 1993",  
    "location_code": "rws",  
    "item_status_code": "-",
    "best_title": "Sphere : a novel",
    "due_gmt": "2015-12-29 04:00:00-05",
    "inventory_gmt": "2007-02-22 14:44:38-05"
}
```



RestFul API

APPLICATION/
CLIENT



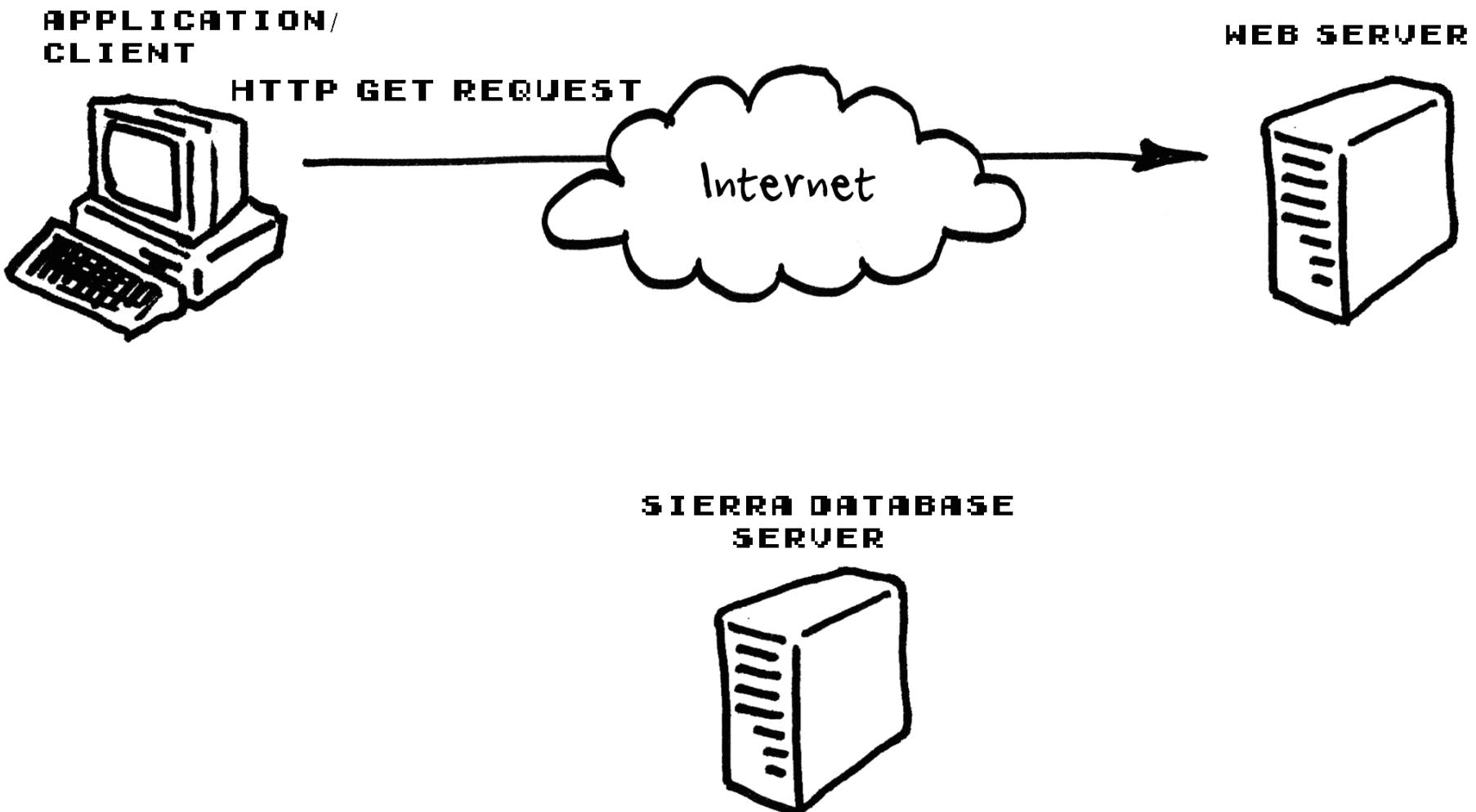
WEB SERVER



SIERRA DATABASE
SERVER



RestFul API

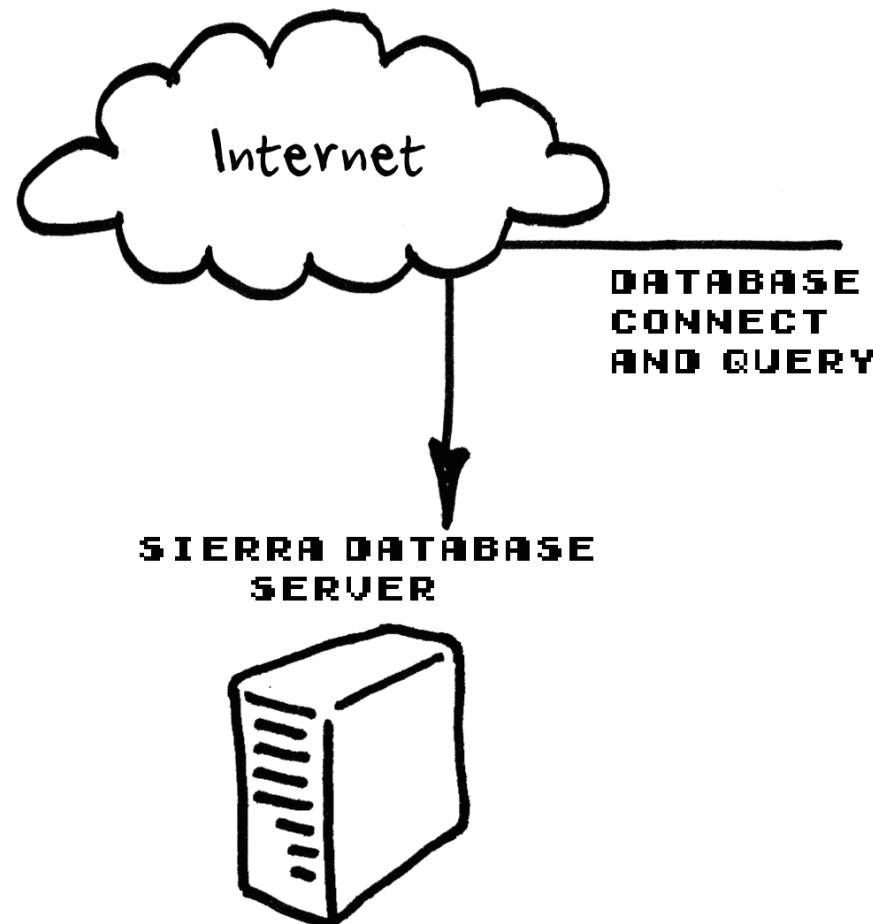


RestFul API

APPLICATION/
CLIENT



WEB SERVER

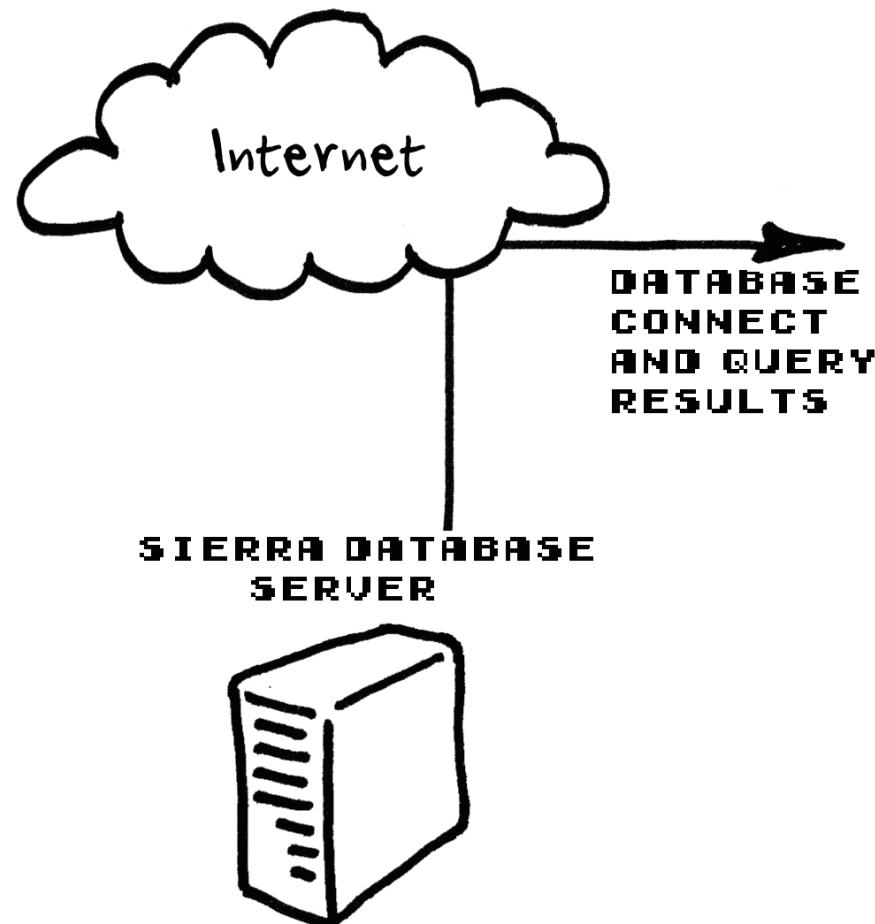


RestFul API

APPLICATION/
CLIENT



WEB SERVER



RestFul API

APPLICATION/
CLIENT



RESULT DATA
RETURNED
IN JSON,
JSONP,
CSV,
TSV, OR . . .

WEB SERVER



SIERRA DATABASE
SERVER

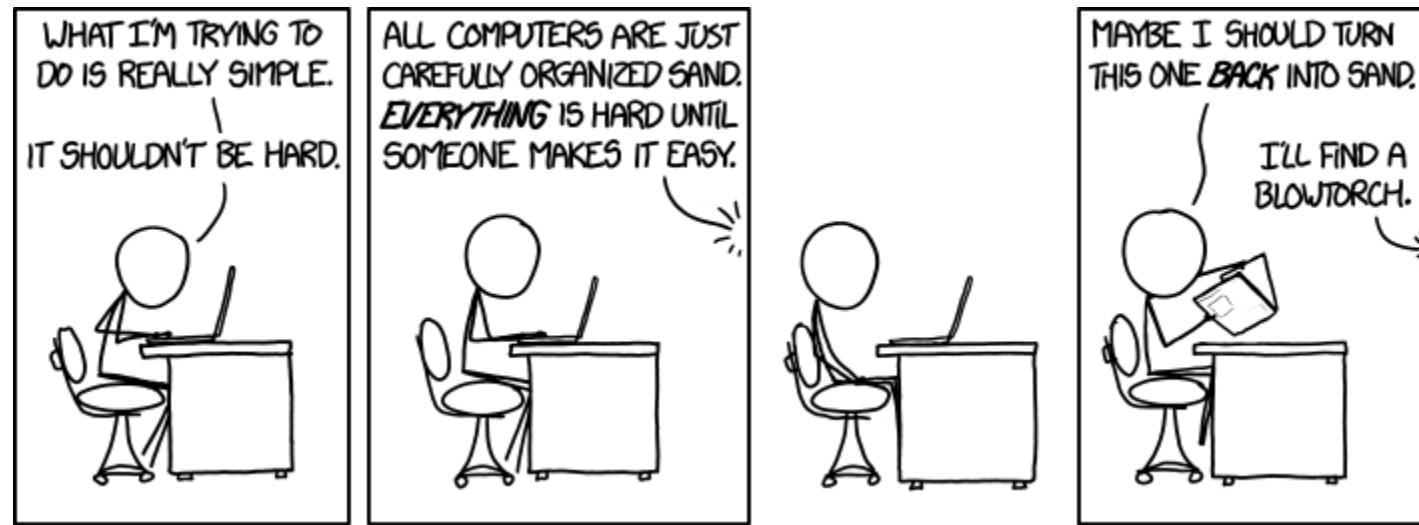


My Recipe For A DIY RESTful API

1. Keep It Simple Stupid

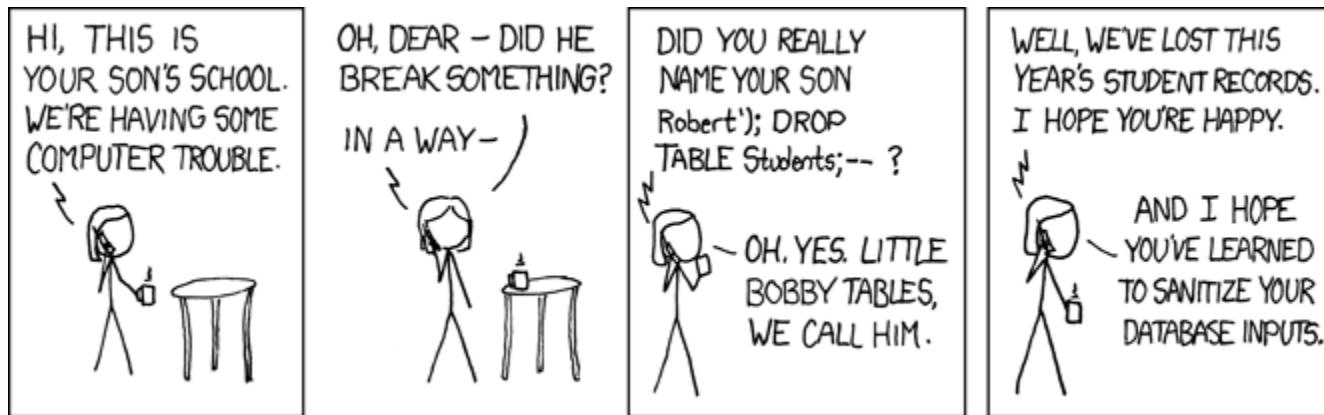


2. Create (or use) as simple of an SQL query as possible.



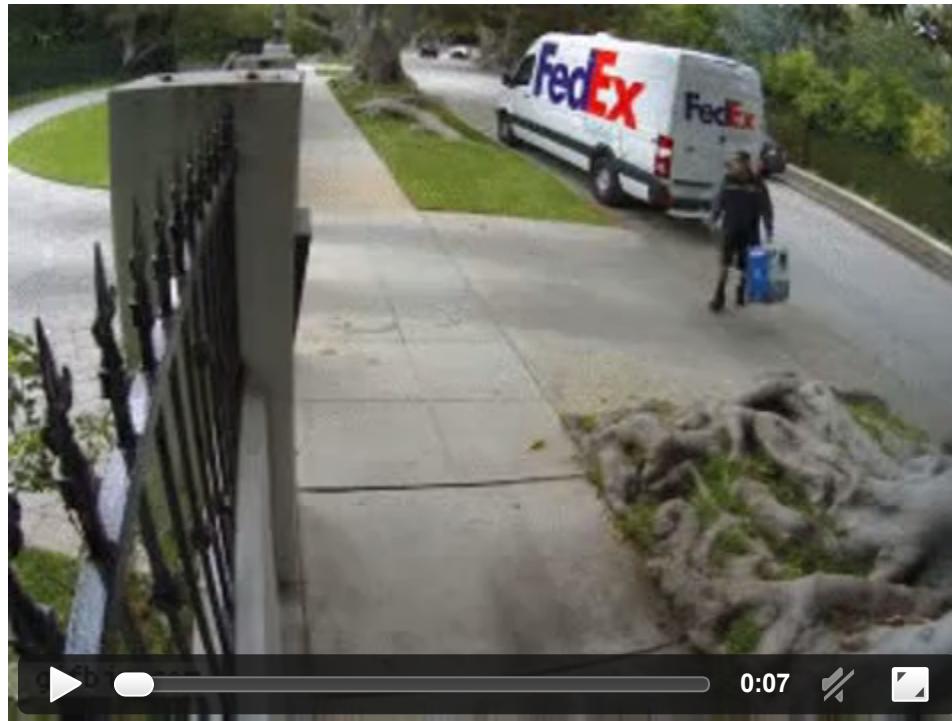
<https://xkcd.com/1349/>

3. Keep your inputs(if any) simple and limited. Sanitize any inputs!



<https://xkcd.com/327/>

4. Connect to database server, prepare the SQL statement, issue query, and get results.
5. Prepare and encode results into JSON (or JSONP) format, and return them over HTTP.



Example #1

Study Room Availability

Study Room Availability Sierra SQL

```
SELECT
i.item_status_code,
c.checkout_gmt,
c.due_gmt,
p.best_title,
p.best_title_norm

--- continued ... ---
```

Study Room Availability Sierra SQL (cont'd)

```
FROM
sierra_view.item_record AS i

LEFT OUTER JOIN sierra_view.checkout AS c
ON (i.record_id = c.item_record_id)

LEFT OUTER JOIN sierra_view.bib_record_item_record_link AS l
ON (i.record_id = l.item_record_id)

JOIN sierra_view.bib_record AS b
ON (l.bib_record_id = b.record_id)

JOIN sierra_view.bib_record_property AS p
ON (b.record_id = p.bib_record_id)

--- continued ... ---
```

Study Room Availability Sierra SQL (cont'd)

```
WHERE
(i.itype_code_num = 69 OR i.itype_code_num = 71)
AND i.is_suppressed = FALSE
AND b.is_suppressed = FALSE

ORDER BY
p.best_title_norm ASC
---
```

PHP Implementation of the Study Room RESTful API

Code Samples

Sanitize inputs

```
<?php
/* replace all the non alphanumeric characters from the
callback function name */
if(isset( $_GET['callback'] ) ) {
    $callback = preg_replace("/[^a-zA-Z0-9\s]/",
                           "",
                           $_GET['callback']
                           );
}
?>
```

PHP Implementation of the Study Room RESTful API

Code Samples (cont'd)

Connect to database server, prepare the SQL statement, issue query, and get results.

```
<?php
try {
    $connection = new PDO($dsn, $username, $password);
}

// ...

$stmt = $connection->prepare($sql);
$stmt->execute();
$row = $stmt->fetchAll(PDO::FETCH_ASSOC);
?>
```

PHP Implementation of the Study Room RESTful API

Code Samples (cont'd)

Encode results into JSON (or JSONP) format, and return them over HTTP

```
<?php
/* return JSONP data if a callback name was specified,
otherwise return plain JSON data */
if(isset ($callback)) {
    header('Content-Type: text/javascript; charset=utf8');
    echo $callback . '(' . json_encode($row) . ')';
}
else {
    header('Content-Type: application/json; charset=utf8');
    echo json_encode($row);
}
?>
```

Study Room Availability Output

(Sample JSON object)

```
[  
  {  
    "item_status_code": "-",  
    "checkout_gmt": null,  
    "due_gmt": null,  
    "best_title": "Study Room 2 - A",  
    "best_title_norm": "study room 2 a",  
    "available": true  
  },  
  {  
    "item_status_code": "-",  
    "checkout_gmt": "2015-08-31 16:09:14-04",  
    "due_gmt": "2015-08-31 17:45:00-04",  
    "best_title": "Study Room 2 - B",  
    "best_title_norm": "study room 2 b",  
    "available": false  
  }]  
]
```

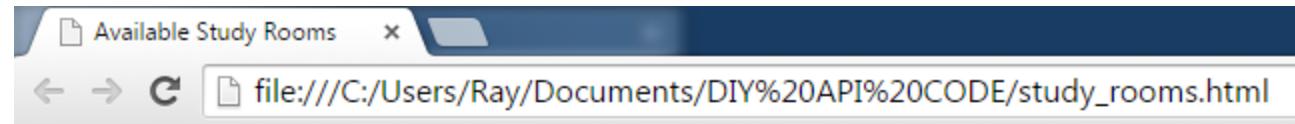
Study Room Availability Output

(Sample JSONP object)

```
callback([  
    {  
        "item_status_code": "-",  
        "checkout_gmt": null,  
        "due_gmt": null,  
        "best_title": "Study Room 2 - A",  
        "best_title_norm": "study room 2 a",  
        "available": true  
    },  
    {  
        "item_status_code": "-",  
        "checkout_gmt": "2015-08-31 16:09:14-04",  
        "due_gmt": "2015-08-31 17:45:00-04",  
        "best_title": "Study Room 2 - B",  
        "best_title_norm": "study room 2 b",  
        "available": false  
    }  
])
```

Study Room Availability Sample Web Page

(after JavaScript calls RESTful API, and processes the data)



University of Dayton Roesch Library

Available Study Rooms

Room	Available?	Due Date
Study Room 2 - A	YES	-
Study Room 2 - B	NO	2015-08-31 17:45:00-04

(The full sample, including the HTML, JavaScript, and RESTful API implemented in PHP are available for download from my GitHub linked later in this presentation)

Example #2

Inventory Barcode Scanning into Google Sheets

Inventory Barcode Scanning into Google Sheets Overview

For an inventory project, we create Google Spreadsheets, share them with students, and then have them scan item bar codes into them

Google Apps Scripts allow us to transfer data into the sheet via our DIY API call, triggered on the insertion of bar codes

Inventory Barcode Sierra SQL

```
SELECT
    upper(p.call_number_norm ||
          COALESCE(' ' || v.field_content, ''))
) as call_number_norm,
i.location_code,
i.item_status_code,
i.inventory_gmt,
b.best_title,
c.due_gmt

FROM
sierra_view.phrase_entry AS e

JOIN
sierra_view.item_record_property AS p
ON
e.record_id = p.item_record_id
```

PHP Implementation of the Inventory Data RESTful API

- Much of the code is the same from the first example!
- The only input parameter for this RESTful API is the item **barcode**
 - We "sanitize" this input based upon our library **barcode** length: 10 alphanumeric characters (no punctuation or symbols) in our case
- This particular API is only going to be used in **Google Apps Scripting** which will utilize the JSON object we return
 - A **Google Apps Scripting API** (`UrlFetchApp.fetch()`) calling our custom RESTful API!

Google Apps Script

The Application, in this example, that is calling our DIY API is
Google Apps Script

Google Apps Script

Quick look at the Google Apps Script for Google Sheets

```
function onEdit(e) {
  var sheet = SpreadsheetApp.getActiveSheet();

  // cell must have a value, be only one row, and be from the first sheet
  if( e.range.getValue() && (e.range.getNumRows() == 1) && sheet.getIndex() == 1 ) {
    var value = e.range.getValue(),
        spread_sheet_name = SpreadsheetApp.getActiveSpreadsheet().getName();
    e.range.setValue(value.toLowerCase());
  }

  var url = 'http://library2.udayton.edu/api/getInventoryData/item_barcode'
  + 'barcode=' + value;
  var result = UrlFetchApp.fetch(url);
  var json_data = JSON.parse(result.getContentText());

  //make sure we have data back ...
  if(json_data) {
    if (json_data.call_number_norm) {
      e.range.offset(0,1).setValue('=\''
          + json_data.call_number_norm.toUpperCase()
          + '\'');
    }
    //escape all double quotes
    if (json_data.best_title) {
      e.range.offset(0,2).setValue('=\''
          + json_data.best_title.replace(/\\"/g, '\\\\\"')
          + '\'');
    }
  }
}
```

```

        + json_data.best_title.replace(/\//g, '')
        + '\\"');

    e.range.offset(0,3).setValue('=\\"'
        + json_data.location_code
        + '\\"');

    e.range.offset(0,4).setValue('=\\"'
        + json_data.item_status_code
        + '\\"');

}

if(json_data.due_gmt != null) {
    e.range.offset(0,5).setValue('=\\"'
        + json_data.due_gmt.substring(0, json_data.due_gmt.length)
        + '\\"');

}
else {
    e.range.offset(0,5).setValue('=\\"\\\"');
}

e.range.offset(0,6).setValue('=\\"'
    + Utilities.formatDate(new Date(), "GMT-4:00", "yyyy-MM"
    + '\\"');

e.range.offset(0,7).setValue(e.range.getRow());
e.range.offset(0,8).setValue(spread_sheet_name);

} //end if
} //end function onEdit()

```

Google Apps Script

Adding the Google Apps Script to the Google Spreadsheet ...

```
/*
// Ray Voelker
// University of Dayton Libraries
// 300 College Park Dayton, OH 45419-1360
// rvoelker1@dayton.edu
// ray.voelker@gmail.com
// If you have any questions or comments
// about this script, page or feature, please
// feel free to contact me.
*/
/*
This is a Google Apps Script intended to be used in conjunction with a RESTful API (item_barcode.php)
To use this script:
1. Create a new Google Spreadsheet, select
   "Tools -> Script Editor" from the menu
2. Copy and paste this code into the script editor, save it, and then select
   "Resources -> Current project's triggers"
3. Add a trigger for the onEdit() function ...
   Run: onEdit
   Events: From spreadsheet :On edit
   select "Save" and give proper authorization
4. Test the sheet ... pasting this barcode into the first cell should yield results: R701498024
*/
function onEdit(e) {
  var sheet = SpreadsheetApp.getActiveSheet();
  // cell must have a value, be only one row, and be from the first sheet
  if( e.range.getValue() && (e.range.getNumRows() == 1) && sheet.getIndex() == 1) {
    //Logger.log( e.range.getValue());
    var value = e.range.getValue();
    spread_sheet_name = SpreadsheetApp.getActiveSpreadsheet().getName();
    e.range.setValue(value.toLowerCase());

    var url = 'http://library2.udayton.edu/api/getInventoryData/item_barcode.php?'
      + 'barcode=' + value;
    var result = UrlFetchApp.fetch(url);
    var json_data = JSON.parse(result.getContentText());

    //make sure we have data back ...
    if(json_data) {
      if (json_data.call_number_norm) {
        e.range.offset(0,1).setValue('=\\"'+ json_data.call_number_norm.toUpperCase() + '\\\"');
      }
      //escape all double quotes
      if (json_data.best_title) {
        e.range.offset(0,2).setValue('=\\"'+ json_data.best_title.replace(/\\"/g, '\"') + '\\\"');
      }
      e.range.offset(0,3).setValue('=\\"'+ json_data.location_code + '\\\"');
      e.range.offset(0,4).setValue('=\\"'+ json_data.item_status_code + '\\\"');
    }
  }
}
```

1:01

RESTful API - Inventory Barcode In Action

The screenshot shows a Google Sheets spreadsheet titled "live barcode scanning". The spreadsheet has a header row with columns A through M. Rows 1 through 28 are visible on the left side. The top menu bar includes File, Edit, View, Insert, Format, Data, Tools, Add-ons, Help, and Inventory. The "Inventory" option is highlighted. The status bar at the bottom indicates "All changes saved in Drive". The bottom navigation bar features buttons for "+", "inventory", "shelflist", "start/end", "barcodes", and a green checkmark icon. A video player interface at the very bottom shows a play button, a progress bar at 0:32, and other video controls.

Inventory Barcode Scanning into Google Sheets

(The full sample, including the HTML, JavaScript, and RESTful API implemented in PHP are available for download from my GitHub linked later in this presentation)

New Books List Overview

Show items to the public that were cataloged in the last 30 days

Allows patrons to limit the output based on classification
(Library of Congress)

May want to add the ability for patrons to "subscribe" to the new books list through e-mail ... sending them only the classifications they may be interested in

New Books List Sierra SQL

```
SELECT
    b.record_id,
    b.cataloging_date_gmt::DATE AS cataloging_date_gmt,
    p.best_author,
    p.best_title,
    p.publish_year,
    r.record_num,
    UPPER(
        x.call_number_prefix
    ) as call_number_prefix,
    UPPER(
        COALESCE (i.call_number_norm, x.call_number_prefix)
    ) as call_number_norm

FROM
    sierra_view.bib_record AS b
JOIN
    sierra_view.record_metadata AS r
```

PHP Implementation of the New Books RESTful API

Again, mostly the same code as the previous two examples!

PHP Implementation of the New Books RESTful API

Sanitize inputs

```
<?PHP
if(isset( $_GET['call_number_prefix'] )) {
    // call number prefix can be at most 3 characters, and must
    // only consist of letters
    // ... unless we are requesting something else from the
    // prefix ... like the null values for example
    switch($_GET['call_number_prefix']) {
        case 'null' :
            $call_number_prefix = null;
            $call_number_prefix_sql =
                'AND x.call_number_prefix is null';
            break;
        case 'all' :
        case 'ALL' :
        case 'All' :
            break;
        default :
            $call_number_prefix = substr(
```

New Books List Sample Web Page

UD Libraries New Books University of Dayton Lib... library2.udayton.edu/ap... Ray X

file:///C:/Users/Ray/Documents/GitHub/DIY_API/code_examples/new_

University of Dayton Roesch Library

New Books in the Last 30 Days (3231 new books) 8/17/2015 to 9/16/2015

ALL ▾

AA

Ramalho, Luciano, author
[Fluent Python : clear, concise, and effective programming](#)
2015

B

Küplen, Mojca, author
[Beauty, ugliness and the free play of imagination : an approach to Kant's aesthetics](#)
2015

B

Clerbout, Nicolas, author
[Linking game-theoretical approaches with constructive type theory : dialogical strategies, CTT demonstrations and the axiom of choice](#)
2015

New Books List Sample Web Page

UD Libraries New Books University of Dayton Libr library2.udayton.edu/api Ray X

file:///C:/Users/Ray/Documents/GitHub/DIY_API/code_examples/new_   

University of Dayton Roesch Library

New Books in the Last 30 Days (1 new books) 8/17/2015 to 9/16/2015

- A : GENERAL WORKS
- ALL
- A : GENERAL WORKS**
- B : PHILOSOPHY. PSYCHOLOGY. RELIGION
- C : AUXILIARY SCIENCES OF HISTORY
- D : WORLD HISTORY AND HISTORY OF EUROPE, ASIA, AFRICA, AUSTRALIA, NEW ZEALAND, ETC.
- E : HISTORY OF THE AMERICAS
- F : HISTORY OF THE AMERICAS
- G : GEOGRAPHY. ANTHROPOLOGY. RECREATION
- H : SOCIAL SCIENCES
- J : POLITICAL SCIENCE
- K : LAW
- L : EDUCATION
- M : MUSIC AND BOOKS ON MUSIC
- N : FINE ARTS
- P : LANGUAGE AND LITERATURE
- Q : SCIENCE
- S : AGRICULTURE
- T : TECHNOLOGY
- U : MILITARY SCIENCE
- V : NAVAL SCIENCE

New Books List Sample Web Page

A screenshot of a web browser window. The title bar shows three tabs: "UD Libraries New Books", "University of Dayton Libr", and "library2.udayton.edu/ap...". The main content area displays the text "University of Dayton Roesch Library" in large bold letters, followed by "New Books in the Last 30 Days (1 new books) 8/17/2015 to 9/16/2015". Below this, a dropdown menu is open, showing the option "A : GENERAL WORKS". Underneath the dropdown, the book "AA" by Ramalho, Luciano, author is listed, along with the title "Fluent Python : clear, concise, and effective programming" and the year "2015".

University of Dayton Roesch Library

New Books in the Last 30 Days (1 new books) 8/17/2015 to 9/16/2015

A : GENERAL WORKS

AA
Ramalho, Luciano, author
Fluent Python : clear, concise, and effective programming
2015

This presentation, along with all samples, can be found on GitHub!

https://github.com/rayvoelker/DIY_API