

# APIs for the Modern World

Addison Hart

# Background

- Reverse engineer and data scientist with experience in geospatial, transit and video game APIs
- Created the Public Mario Maker 2 API

## Super Mario Maker 2 Public API

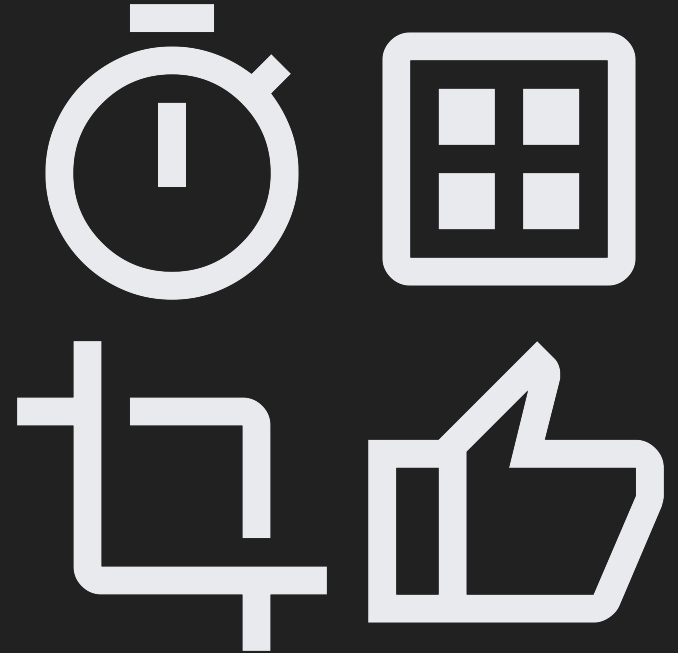
Made by TheGreatRambler with help from TatiAus, Gible\_V, Funnier04, Warspyking, AppleSinger, T04Dw, and Wizulus. This is a hosted instance of [MarioOver](#).

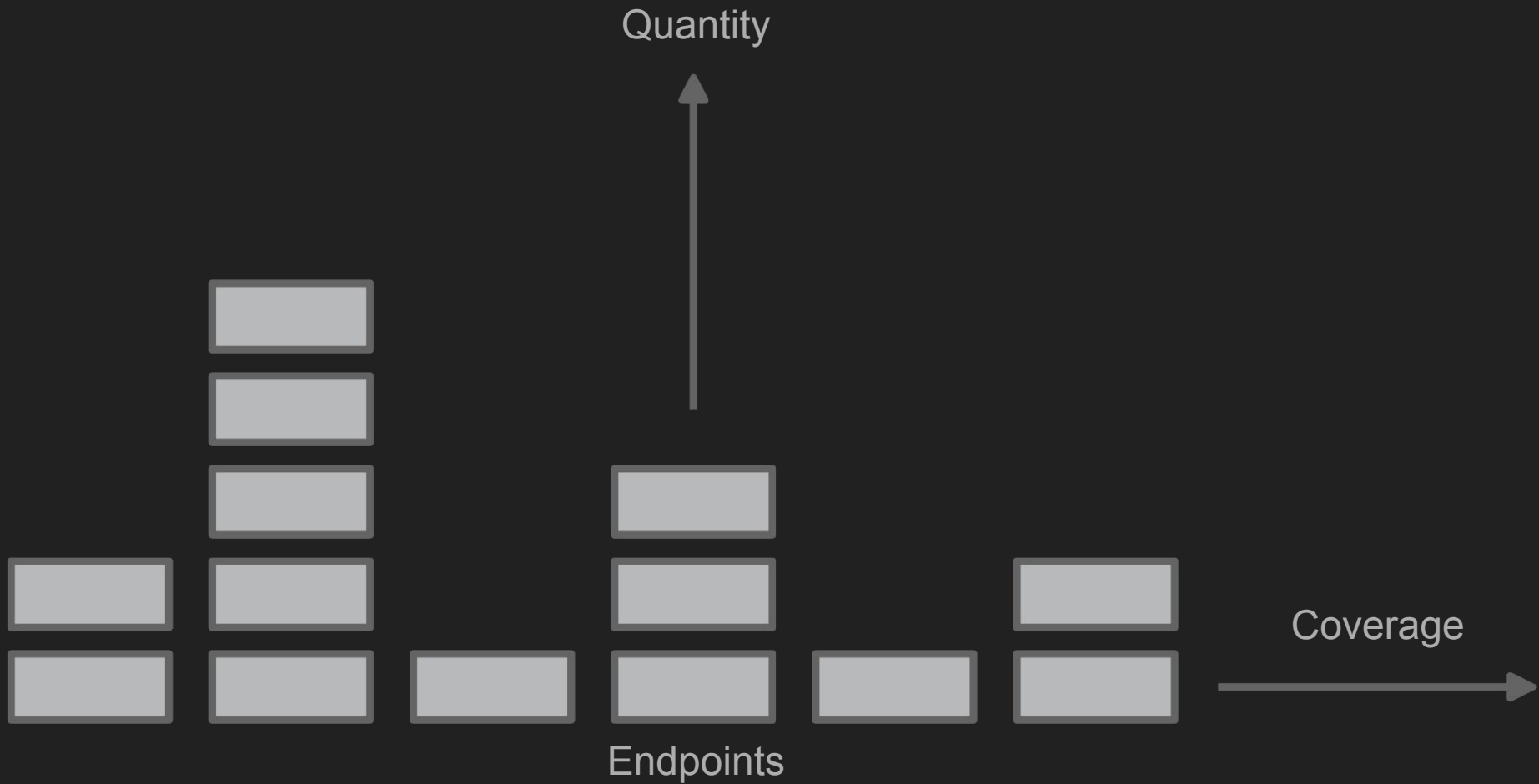
```
/level_info/{course_id}
/user_info/{maker_id}
/level_info_multiple/{course_ids}
/level_comments/{course_id}
/level_played/{course_id}
/level_deaths/{course_id}
/level_thun
/level_enti
/level_data
/level_da
/ninji_info
/ninji_ghosts
/get_posted/{
/get_liked/{maker
/get_played/{maker_id}
/get_first_cleared/{maker_id}
/get_world_record/{maker_id}
/get_super_worlds
/super_world/{super_world_id}
/search_endless_mode
/search_new
/search_popular
```



# What Makes a Great API?

- Latency
- Quantity
- Coverage
- Quality



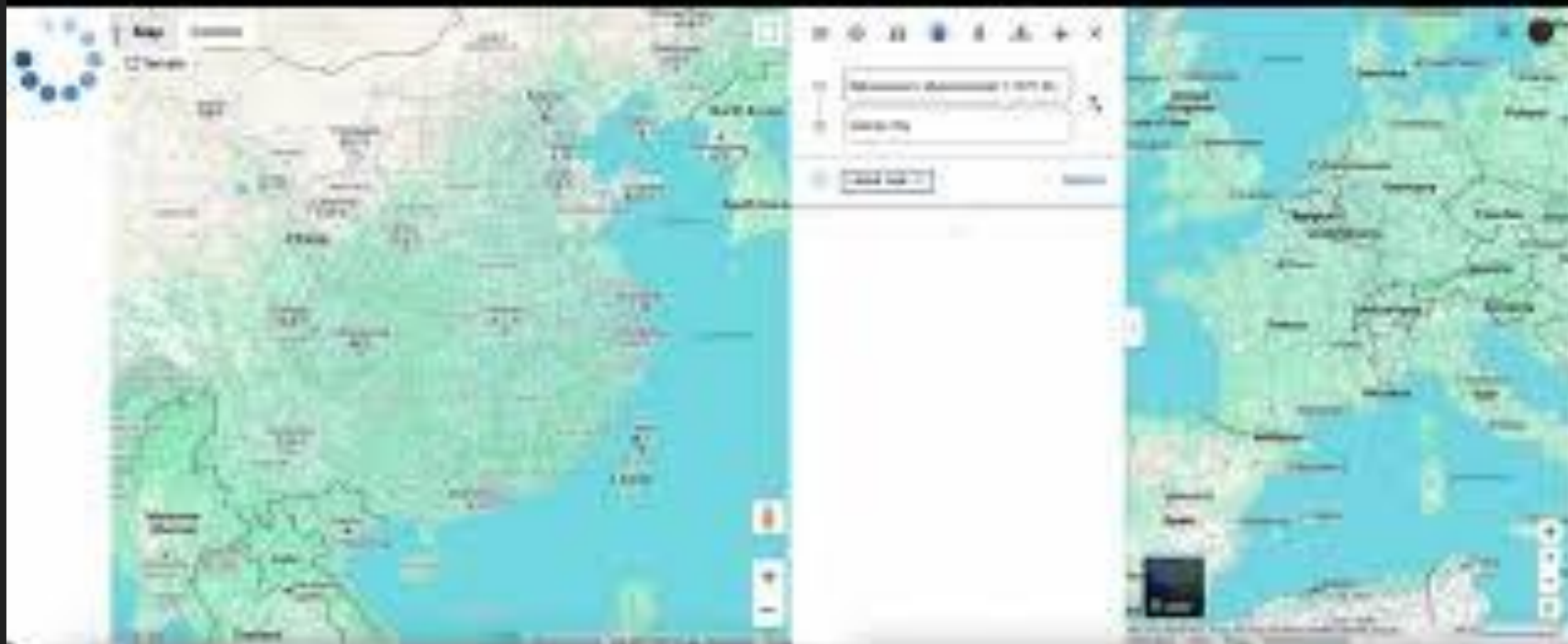


# Latency



Serving users fast with good performance

# Latency (Baidu Maps vs Google Maps)



# Latency (ranking in Mario Maker 2 Ninji)

```
294 // Obtain player rank
295 row := s.Conn.QueryRow(context.Background(), `
296     SELECT pid_rank FROM (SELECT pid,
297         DENSE_RANK() over (ORDER BY time ASC)
298         AS pid_rank FROM ninji_time WHERE
299         data_id = $2 AND active IS TRUE) AS _
300     WHERE pid = $1 LIMIT 1`, current_pid,
301         data_id)
302 err := row.Scan(&histogram.Rank)
303 if err == pgx.ErrNoRows {
304     // If have not played rank is 0
305     histogram.Rank = 0
306 } else if err != nil {
307     return histogram, err
308 }
```

```
321 // Get histogram of values
322 // Always starts at 10000, always ends at
323 // 120000, always 110 buckets
324 rows, err := s.Conn.Query(context.Background(),
325     `SELECT WIDTH_BUCKET(time, 10000, 120000,
326     110) FROM ninji_time WHERE data_id = $1
327     AND active IS TRUE`, data_id)
328 if err != nil {
329     return histogram, err
330 }
```



# Latency (ghosts in Mario Maker 2 Ninji)

```
531 // Select n times around a time, with an equal
532 // number on either side
533 only_active := "AND active IS TRUE"
534 if true {
535     | only_active = ""
536 }
537 rows, err := s.Conn.Query(context.Background(),
538     fmt.Sprintf(`(SELECT pid, time, replay_id
539     FROM ninji_time WHERE data_id = $1 AND
540     replay_id IS NOT NULL %s AND time > $2
541     ORDER BY ABS($2 - time) ASC LIMIT $3)
542     UNION ALL
543     (SELECT pid, time, replay_id FROM
544     ninji_time WHERE data_id = $1 AND
545     replay_id IS NOT NULL %s AND time < $2
546     ORDER BY ABS($2 - time) ASC LIMIT $3)
547     `, only_active, only_active),
548     param.DataId, param.Time, param.Count/2)
549 if err != nil {
550     | return infos, err
551 }
552 defer rows.Close()
```





# Latency (world records in Mario Maker 2)

```
func (s *DB) UpdateRecordsAndFirstClears(result PlayResult, player_result
PlayerResult) (bool, bool, error) {
    var newWorldRecord, newFirstClear bool
    // Modify status status of other player if they were beaten
    // This is for faster lookup later
    oldPid, oldTime, err := s.GetPlayerResultWorldRecord(int64(result
DataId))
    hadWorldRecord := true
    if err != nil {
        return false, false, err
    } else if oldPid == 0 {
        hadWorldRecord = false
    }

    if hadWorldRecord && uint32(result.Time.Int32) < oldTime {
        // Revert the other world record
        newWorldRecord = true
        err = s.Conn.Exec(context.Background(), `UPDATE player_result
SET is_world_record = False WHERE pid = $1 AND data_id = $2`,
oldPid, player_result.DataId)
        if err != nil {
            return false, false, err
        }
        err = s.Conn.Exec(context.Background(), `UPDATE player_result
SET is_world_record = True WHERE pid = $1 AND data_id = $2`,
player_result.Pid, player_result.DataId)
        if err != nil {
            return false, false, err
        }
    } else if !hadWorldRecord {
        // World record because no one has played it
        newFirstClear = true
        err = s.Conn.Exec(context.Background(), `UPDATE player_result
SET is_first_clear = True, is_world_record = True WHERE pid = $1
AND data_id = $2`, player_result.Pid, player_result.DataId)
        if err != nil {
            return false, false, err
        }
    }

    return newWorldRecord, newFirstClear, nil
}
```

```
96 func (s *DB) GetPlayerResultWorldRecord(dataId
int64) (datastore.Pid, uint32, error) {
97     // Get first person with the best time
98     rows, err := s.Conn.Query(context.Background
(), "SELECT pid, best_time FROM player_result
WHERE data_id = $1 AND is_world_record = True
LIMIT 1", dataId)
99     if err != nil {
100         return datastore.Pid(0), 0, err
101     }
102     defer rows.Close()
103
104     if !rows.Next() {
105         // No world record
106         //return datastore.Pid(0), 0xFFFFFFF, nil
107         return datastore.Pid(0), 0, nil
108     }
109
110     var pid datastore.Pid
111     var time pgtype.Int4
112     err = rows.Scan(&pid, &time)
113     return pid, uint32(time.Int32), err
114 }
```

VS

```
96 func (s *DB) GetPlayerResultWorldRecord(dataId
int64) (datastore.Pid, uint32, error) {
97     // Get first person with the best time
98     rows, err := s.Conn.Query(context.Background
(), "SELECT pid, best_time FROM player_result
WHERE data_id = $1 ORDER BY best_time ASC
LIMIT 1", dataId)
99     if err != nil {
100         return datastore.Pid(0), 0, err
101     }
102     defer rows.Close()
103
104     if !rows.Next() {
105         // No world record
106         return datastore.Pid(0), 0, nil
107     }
108
109     var pid datastore.Pid
110     var time pgtype.Int4
111     err = rows.Scan(&pid, &time)
112     return pid, uint32(time.Int32), err
113 }
```

## Is this overengineered?

# Latency (world records in Mario Maker 2)

```
56 // update user stats cache
57 func (s *DB) UpdateUserStatsCache(pid datastore.
Pid) error {
58     var stats UserStats
59
60     // query from player_result table
61     row := s.Conn.QueryRow(context.Background(),
"SELECT COUNT(pid) AS tries, SUM(deaths) AS deaths,
SUM(cleared::int) AS clears, SUM(is_world_record::int) AS world_records,
SUM(is_first_clear::int) AS first_clears FROM player_result
WHERE pid = $1", pid)
62     err := row.Scan(&stats.Plays, &stats.Tries, &
stats.Deaths, &stats.Clears, &stats.
WorldRecords, &stats.FirstClears)
63     if err != nil {
64         return nil
65     }
66
67     // Update the user_stats table
68     _, err = s.Conn.Exec(context.Background(), `
69     INSERT INTO user_stats (id, pid, plays,
tries, deaths, clears, world_records,
first_clears) VALUES (DEFAULT, $1, $2, $3,
$4, $5, $6, $7)
70     ON CONFLICT (pid) DO UPDATE SET plays =
$2, tries = $3, deaths = $4, clears = $5,
world_records = $6, first_clears = $7
71     `, pid, stats.Plays, stats.Tries, stats.
Deaths, stats.Clears, stats.WorldRecords,
stats.FirstClears)
72     return err
73 }
```

```
98 func (s *DB) SearchCoursesWorldRecord(state *user.
State, query_pid datastore.Pid, param view.
SearchCoursesParam) ([]datastore.CourseInfo, []
uint32, bool, error) {
99     // Get all world records held by this user
100     rows, err := s.Conn.Query(context.Background
(), "SELECT data_pid FROM player_result WHERE
pid = $1 AND is_world_record = True ORDER BY
time_obtained DESC LIMIT $2 OFFSET $3",
query_pid, param.Size, param.Offset)
101     if err != nil {
102         return []datastore.CourseInfo{}, []uint32
{}, false, err
103     }
104     defer rows.Close()
```

NO!  $O(1)$  vs  $O(n)$

# Latency (restrict access)

- Sometimes clever engineering is not enough
- Rate limiting or request throttling may be enough, as long as your users are aware of such measures and not trying to actively break them

```
lock = asyncio.Semaphore(5)
```

```
await check_tokens()
async with lock:
    async with backend.connect(s, HOST,
    PORT) as be:
        async with be.login(str(user_id),
        auth_info=auth_info) as client:
            store = datastore.
            DataStoreClientSMM2(client)
            print("Want course info for "
            + course_id)
            course_info_json = await
            obtain_course_info(course_id,
            store, noCaching)

            if invalid_level
            (course_info_json):
                return ORJSONResponse
                (status_code=400,
                content=course_info_json)

            return ORJSONResponse
            (content=course_info_json)
```

# Latency (benefits)

- Live data feeds enable real time reactions to events
- Tools like Google Takeout are not suitable for real time reactions



BronzeSpaceComet | CBB Rank 3 11/11/2021 7:20 PM

WE CAN USE THE API

LMAO

YES



CDR 11/11/2021 7:20 PM

Im



Wipeoutjack7 11/11/2021 7:20 PM

im gonna cry because this is so weird



Gible\_V | Otter | BR 37.966 11/11/2021 7:44 PM

Tatiasu how are you calling this



tatiasu 11/11/2021 7:44 PM

40.243 wr

```
{
  "time": 40243,
  "uploader_pid": 13441009339719647000,
  "replay_file": {
    "url": "https://dd04qv7ro5r21.cloudfront.net",
    "data_type": 40,
    "size": 2677,
    "filename": "event-course-ghost_a75_ba88ec7",
    "headers": {
      "Cookie": "CloudFront-
      Policy=eyJTdGF9Z2w1bnQ101t7111c291cmN1I
      Vz1RoYw410ns1QVdTOkVvb2HoVG1tZSI6MTYzNj
      JmBoNTYkUay2---5my4K6c2BrBuXQRpWA4eEUoaC
      AD95QBRTtu49x0TRPg_; CloudFront-Key-P
    }
  },
  "uploader_code": "KVGYLKQLF",
  "uploader_name": "Caesar's",
  "uploader_country": "US",
  "uploader_last_active": "12-11-2021 2:18:56"
}
```

@Gible\_V | Otter | BR 37.966 [http://tgrcode.com/mm2/ninji\\_ghosts/0?time=40021&num=10](http://tgrcode.com/mm2/ninji_ghosts/0?time=40021&num=10)



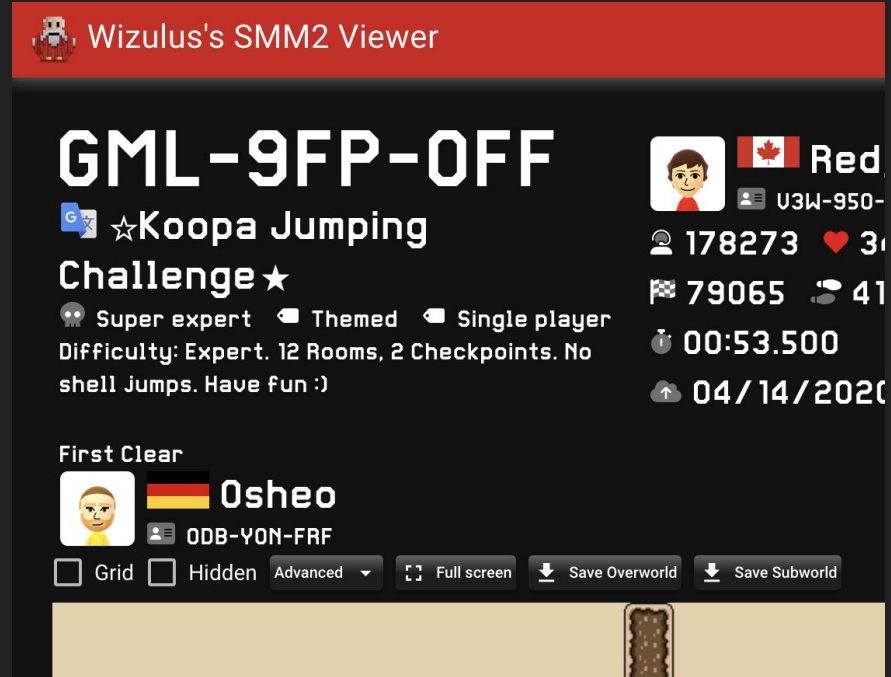
Gible\_V | Otter | BR 37.966 11/11/2021 7:45 PM

Okay so they got moved over

Also what are viewing them on?

# Latency (benefits)

- Have special exemptions on rate limiting or request throttling for specific trusted users with smart caching policies




Wizulus's SMM2 Viewer



## GML-9FP-OFF

☆ Koopa Jumping Challenge ☆

Super expert Themed Single player  
Difficulty: Expert. 12 Rooms, 2 Checkpoints. No shell Jumps. Have fun :)

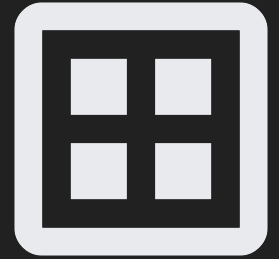
Red  U3W-950-  
178273 3  
79065 41  
00:53.500  
04/14/2020

First Clear

  **Osheo**  
ODB-YON-FRF

Grid  Hidden

# Quantity



Providing as much data as possible over each endpoint

# Quantity

- Mario Maker 2 only serves the last 1000 comments of a level

```
"courses": [  
  {  
    "name": "Celestial Sprint Speedrun [20s]",  
    "description": "Never stop running! Trust the fake traps! And most importantly, enjoy!",  
    "uploaded_pretty": "29-6-2019 21:08:13",  
    "uploaded": 1561842493,  
    "data_id": 3456358,  
    "num_comments": 133034,  
  }  
]
```

```
☰ /level_comments/{course_id}
```

Returns the comments of the specified course

Course ID

! {course\_id}: The 9 digit code used to identify a course 7N1MVBWKF

[https://tgrcode.com/mm2/level\\_comments/7N1MVBWKF](https://tgrcode.com/mm2/level_comments/7N1MVBWKF)

```
unk14: ,  
unk17: 0,  
poster: [Object]  
},  
{  
  comment_id: '20241016150033345909_81c7a1db10a5a50f_34bd66',  
  posted_pretty: '16-10-2024 15:00:33',  
  posted: 1729090833,  
  clear_required: false,  
  reaction_image_id: 5,  
  reaction_image_id_name: 'Wow!',  
  type_name: 'Reaction Image',  
  type: 2,  
  has_beaten: false,  
  x: 2148,  
  y: 128,  
  reaction_face: 0,  
  reaction_face_name: 'Normal',  
  unk8: 0,  
  unk10: 0,  
  unk12: false,  
  unk14: '',  
  unk17: 0,  
  poster: [Object]  
},  
... 883 more items  
}
```

# Quantity (pagination)

- Enable pagination to reduce average payload size while still allowing access to all data
- Up to clients how much they value quantity (in both time and money)





# Quantity (options for advanced users)

Single request option  
for normal use

## ARIN Whois/RDAP

5.161.193.184

Search

» Search [www.arin.net](http://www.arin.net) instead

► Search Filter: **Automatic**

all requests subject to [terms of use](#)

"5.161.193.184"

### Network: 5.161.193.0 - 5.161.193.255

#### Source Registry

RIPE NCC

#### Net Range

5.161.193.0 - 5.161.193.255

#### CIDR

5.161.193.0/24

#### Name

CLOUD-ASH

#### Handle

5.161.193.0 - 5.161.193.255

Bulk request option for  
advanced use

## Accessing Bulk Whois Data

If your request is approved, you will be able to access the Bulk Whois data by performing these steps:

1. Log in to ARIN Online.
2. Select **Downloads & Services** from the navigation menu.
3. In the Bulk Whois section, choose **Bulk Whois Data**. The next window provides a list of files and formats in which the data is available. You have the option to download a file of all Whois object types (POCs, Orgs, NETs, and ASNs) or to selectively download a particular Whois object type in compressed archive (.zip), text (.txt), or extended markup language (.xml) format.

If you want to download a file that contains a combination of Whois object types, you can enter the download URL directly into your browser window, using the following format:

```
https://accountws.arin.net/public/rest/downloads/  
bulkwhois/XXX+YYY.zip
```

where **XXX** and **YYY** can be any combination of:

# Quantity (options for advanced users)

This use case...

Is different from this use case

## Related Entities

### ▼ 4 Entities

#### Kind

Group

#### Full Name

Hetzner Online GmbH - Contact Role

#### Handle

HOAC1-RIPE

#### Email

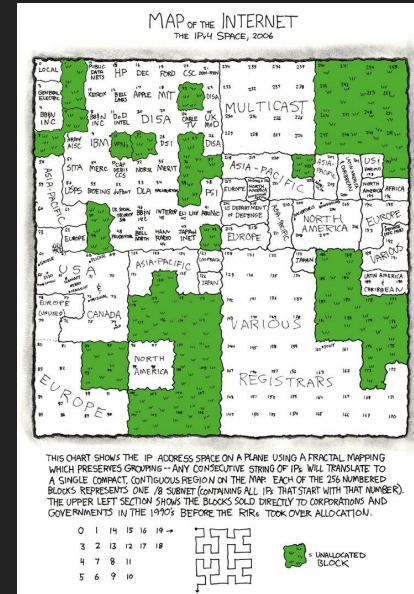
abuse@hetzner.com

#### Telephone

+49 9831 505-0

#### Telephone

+49 9831 505-3



But you want to support both

# Quantity (provide cross references to other data)

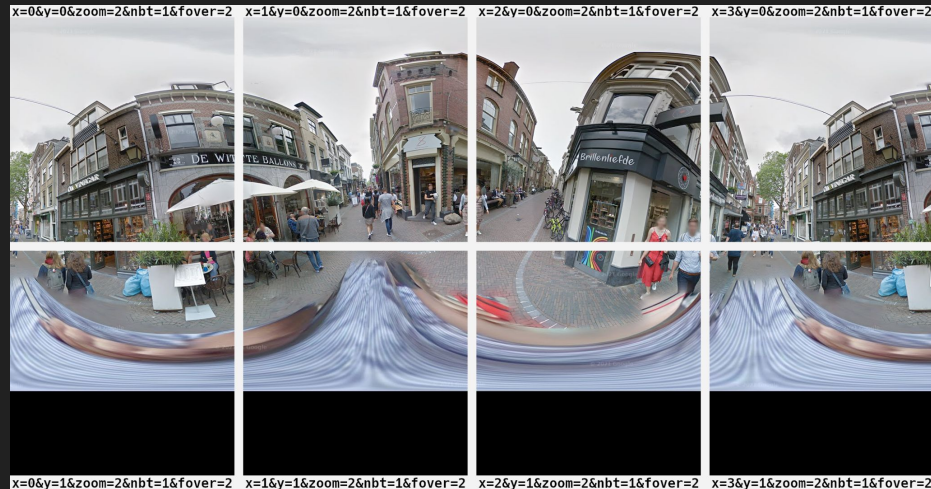
- Provide cross references to additional data if the average user does not use it
- Most users access this endpoint for times, not for replay files

```
"ghosts": [  
  {  
    "time": 32518,  
    "replay_file": {  
      "url": "https://dd04qv7ro5r2l.  
cloudfront.net/10.  
ngs_lp1_22306d00_datastore/ds/1/  
relation_data/event_course_ghost/  
event-course-ghost_982_13ef99468f2d4233  
_b9b71a_20191207232345036295",  
      "size": 2434,  
      "filename":  
      "event-course-ghost_982_13ef99468f2d423  
3_b9b71a_20191207232345036295"  
    },  
    "region": 0,  
    "region_name": "Asia",  
    "code": "TQ61RQJHF",  
    "pid": 1436535334483345971,  
    "name": "あさひざし",  
    "country": "JP",
```

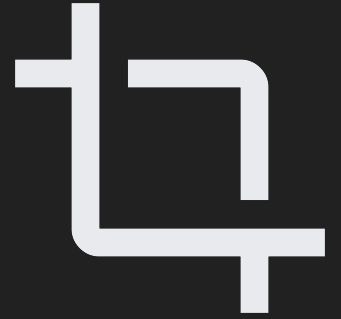


# Quantity (progressive loading)

- Some APIs can benefit from serving in chunks in different resolutions
- Google Maps serves different zoom values (resolution) and different offsets into an equirectangular image
- Once again, up to clients how much they value quantity (in both time and money)



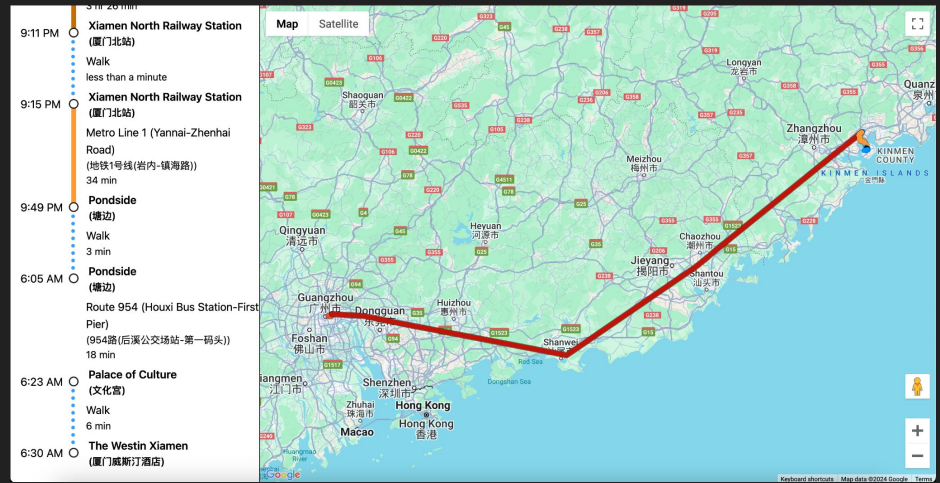
# Coverage



Providing to users as much  
data as you have

# Coverage (expose all)

- Baidu Maps (website version) does not expose bus or metro schedules, had to RE 2 additional apps
- App had access to additional data that it did not expose to users



VS



# Coverage (accommodate new use cases)

- Listen to the community for new endpoint requests
- Try to design API such that new endpoints are improvements on old ones, you could still access that data earlier

Commits on Feb 6, 2022

Merge pull request #1 from TheGreatRambler/wizulus



Verified

1c75c35



TheGreatRambler authored on Feb 6, 2022

Ability to run and test in Docker



wizulus committed on Feb 6, 2022

6c41d54



Upload files



TheGreatRambler committed on Feb 5, 2022

f07b34f



Commits on Dec 3, 2022

Add super\_world\_multiple endpoint (no caching yet)



UncaughtCursor committed on Dec 3, 2022

2f52315



Commits on Feb 28, 2023

Expose super world unk1



TheGreatRambler committed on Feb 28, 2023

75858cb





# Quality (interoperability)

- Expose GUIDs that are compatible with other APIs
- Data like names and coordinates do not always correspond with other APIs

```
"kind": 0,  
"last_station": "\u7b2c\u4e00\u7801\u5934",  
"name": "954\u8def(\u540e\u6eaa\u516c\u4e13)",  
"st_uid": "1340124536317357055",  
"startTime": "06:05",  
"station_num": 4,  
"time": 471,  
"timetable": "06:05-21:00",  
"timetable_ext": ""
```



```
"kind": 0,  
"last_station": "\u9547\u6d77\u8def",  
"name": "\u5730\u94c1\u53f7\u7ebf(\u5ca9\u5185)",  
"st_uid": "847393018162041728",  
"startTime": "06:01",  
"station_num": 13,  
"time": 1860,  
"timetable_ext": ""
```



```
"start_station": "后溪公交场站",  
"end_time": "2100",  
"lineId": "0592-954-1",  
"final_station": "第一码头站",  
"start_time": "0605",  
"buses": [  
  {  
    "syncTime": 1,  
    "pRate": -1,  
    "travelTime": 1441
```



uno.csv

```
1  XMMML01<,>ML<,>Line 1<,>1号线<,>1號綫<,>  
2  XMMML02<,>ML<,>Line 2<,>2号线<,>2號綫<,>  
3  XMMML03<,>ML<,>Line 3<,>3号线<,>3號綫<,>  
4  XMMW01A<,>MW<,>For Zhenhai Road 镇海路  
5  XMMW01B<,>MW<,>For Yannei<,>岩内  
6  XMMW02A<,>MW<,>For Wuyuanwan<,>五缘湾
```





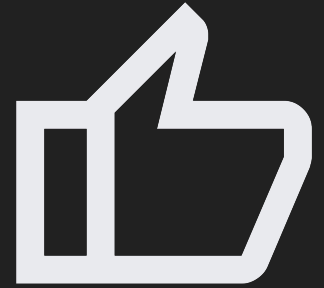
# Examples of Good GUIDs

- ISBN
  - Google Books
  - Open Book
  - Library of Congress
- UPC (Universal Product Code)
  - Walmart
  - Go-UPC
- ISIN / ticker symbol
  - Bloomberg
  - Robinhood

Oftentimes the biggest API should become the GUID



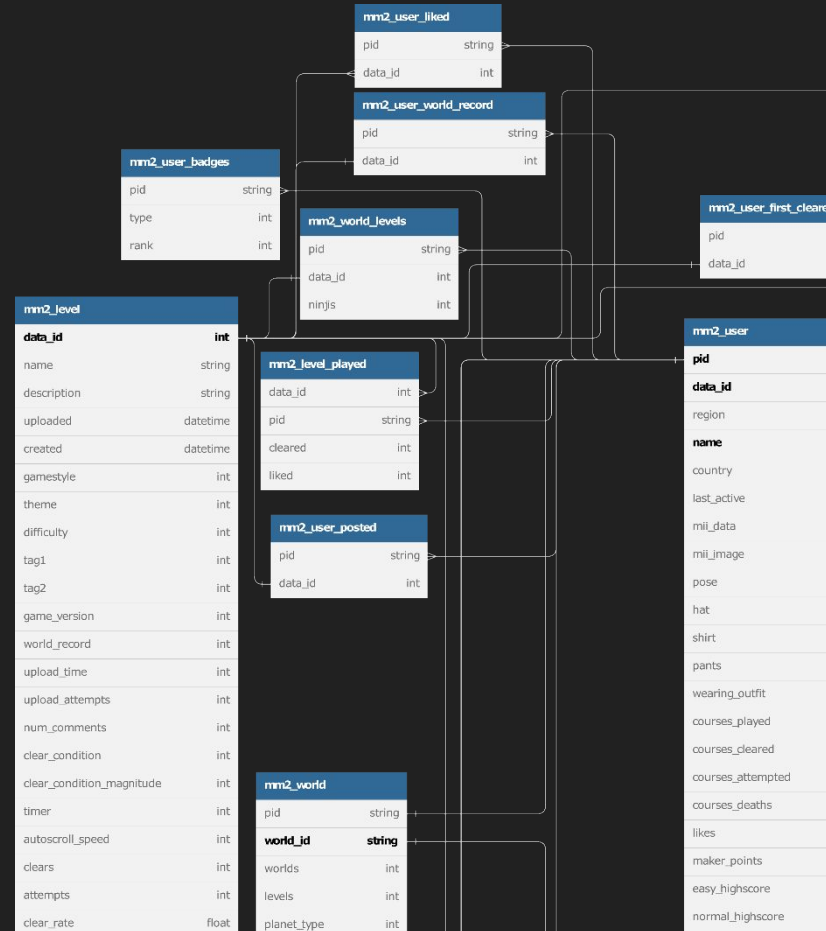
# Quality



Providing a public service that  
inspires experimentation

# Quality (self documenting)

- Give fields representing the same data type the same name
- Ensure every endpoint taking in the same data type accepts the same domain for that data type



# Quality (right data format for the job)

- Sending the wrong data format can incur additional performance penalties or impact developer experience
- Use well accepted standards when possible
  - JSON
  - Protobuffer
- When the data format isn't right, a reverse engineer like me will make a new one!

```
{  
  "jsonr": {  
    "data": {  
      "encryptResult": "xJE7SF7oqbMpNHvQ9Nu1cQCpwIhcVb5N  
+BI2s4kvPS5e4oKuNFGLweos+kPVMhYjwybFVa/aaeXF10YKltriwz+CJGcREESy  
+vsc/  
Ftee01ULCzBttrpGmwm7QMR10iqHZnH0TxfXTRPdc9qZV6A7nu5rvsCu0lMQqirRpCs  
nfS7X6P5wX6X4WFCwUaJP4sfnex7iL8Qe0iu7qKUGk3ub5tiKA1m1xpQHdFSQru1rrY  
+ydQGURDVayqK0nMHtYZVPnmDxVeKQt03awUuWbKnND6KtY0Goa0IFcoCw2ZyZzo1EY  
N1eAjQcKLaRF0  
+4BURRy0prTx90UGMmDAsJhwU9FhzFCPIQBDxnMcnrTeQpUU43aHUjq4EwzG  
+j4I454UzhTaqXCrmR0AbPBZ6+DHIW7fzStb8iIydJAMUj6/7t0fSdsmSEjFdzraR  
+Vh3vHF93W5EwGY+4yITCdP2uXw3gf7t07TCNzB3vyf0r2he8SPBp8TojExGk56h/  
0ebexLl0XAY/01E/ShcYPo0x8a1/  
0E4WpzUDlKkFv2y5pPjXkpncaK9Z0axBqHpA3Iv8hrYsuneM5Snvuu1Ity7kNhN/qT/  
Af6T0NugDBuBftihT80wCL9tYtc1fgLpeqFiT5/CE3MwF/  
781SZYeKQnu1g7k460DUF2F0aXJdoHAtzVrkdVoDUDRoizIFUgie/  
Lgon1cLZ3hyTp0tRhWU6+RoMTp/  
GqJwxXhcd5D8xb9m35PRlyh8sTyjGnkeHRSrH0tiPdVEclkgWU9yV52riBMDjhu8Von  
MpGT2do9ifCwNlco1MliZgVFQveLkd0bcuhRD54amsuf3dW5ws9BH0cL9zQQJ8gGfcE  
RuXR0F0sLH0qzEVZs9BbTphgXfffgaFuNBN89E6VJ  
+p0dh9D1qytA8EfpADwZqR9rvGP+5MeMA0W2nTBIqmpFUSCsL/  
}}
```

# Quality (use logging to understand user behaviors)

- One of the most popular requests is level info for 3 seemingly random levels with no caching
- Upon further inspection, and asking the community, this user is using the “first cleared” field of these levels as a replacement for a “user\_info\_multiple” endpoint
- These 3 players are the top 3 in endless easy, a competitive gamemode leaderboard

TX. AMOUNT ↕	METHOD ↕	PROTOCOL ↕	DATA ↕
—	—	—	—
148.67 GiB (22.50%)	—	—	—
388.7 KiB (00.00%)	—	—	—
1.4 GiB (00.22%)	GET	HTTP/2	/mm2/search_new?count=20
99.8 MiB (00.01%)	GET	HTTP/2	/mm2/search_new?count=1
1.4 GiB (00.21%)	GET	HTTP/2	/mm2/search_new?count=19
146.1 MiB (00.02%)	GET	HTTP/1.1	/mm2/user_info/9V7-4TX-YHG?noCaching=1
12.3 MiB (00.00%)	GET	HTTP/1.1	/mm2/user_info/7W0-HB0-0GF
1.5 GiB (00.23%)	GET	HTTP/1.1	/mm2/level_info_multiple/3DD7TVTGG,4L09WLBXG,5NC1G96MF?noCaching=1
33.5 GiB (05.07%)	GET	HTTP/1.1	/mm2/search_endless_mode?count=300&difficulty=ex
660.73 GiB	—	—	1,782,409

# Quality (example code)

Provide example code for both kinds of users

Python example constructing  
raw request

```
coords = [
    [(39.63186983565579, -105.07732880667866), ...],
    [(49.05043326270549, 7.982416821619636), ...],
    ...
]
KEY = "..."
```

```
for i in range(0, len(coords), 512):
    chunk = coords[i : i + 512]
    url = (
        "https://maps.googleapis.com/maps/api/
        elevation/json?key=%s&locations=%s"
        % (
            KEY,
            "|".join(map(lambda coord: "%f,%f" %
                (coord[1], coord[0]), chunk)),
        )
    )
```

Go example using a client  
library

```
c, _ := NewClient(WithAPIKey(apiKey), WithBaseUrl(server.URL))
r := &ElevationRequest{
    Locations: []LatLng{
        {
            Lat: 39.73915360,
            Lng: -104.9847034,
        },
    },
}

resp, err := c.Elevation(context.Background(), r)
```

# Quality (documentation)

- Document both the domain (kinds of inputs) and range (kinds of outputs) of your endpoints
- Link to external documentation when using a standardized format, like GTFS or DZI (Deep Zoom Image)

```
GameStyles = {
  0: "SMB1",
  1: "SMB3",
  2: "SMW",
  3: "NSMBU",
  4: "SM3DW"
}

Difficulties = {
  0: "Easy",
  1: "Normal",
  2: "Expert",
  3: "Super expert"
}

CourseThemes = {
  0: "Overworld",
  1: "Underground",
  2: "Castle",
  3: "Airship",
  4: "Underwater",
  5: "Ghost house",
  6: "Snow",
  7: "Desert",
  8: "Sky",
  9: "Forest"
}
```

```
TagNames = {
  0: "None",
  1: "Standard",
  2: "Puzzle solving",
  3: "Speedrun",
  4: "Autoscroll",
  5: "Auto mario",
  6: "Short and sweet",
  7: "Multiplayer versus",
  8: "Themed",
  9: "Music",
  10: "Art",
  11: "Technical",
  12: "Shooter",
  13: "Boss battle",
  14: "Single player",
  15: "Link"
}

Regions = {
  0: "Asia",
  1: "Americas",
  2: "Europe",
  3: "Other"
}
```

# Quality (compatibility)

- Have an established plan to preserve compatibility
  - Endpoint prefixes
  - Backwards compatibility
- Advertise changes
- Choose the right UIDs at the start

## ↔ 2024 update

The endpoint `https://www.google.com/maps/preview/photo` has been turned into `https://www.google.com/maps/rpc/photo/listentityphotos`. That change is reflected in this [blogpost](#) where applicable.

## ↔ Motivation

There are two websites which have always led to hours of unplanned exploration for me: Wikipedia and Google Maps.



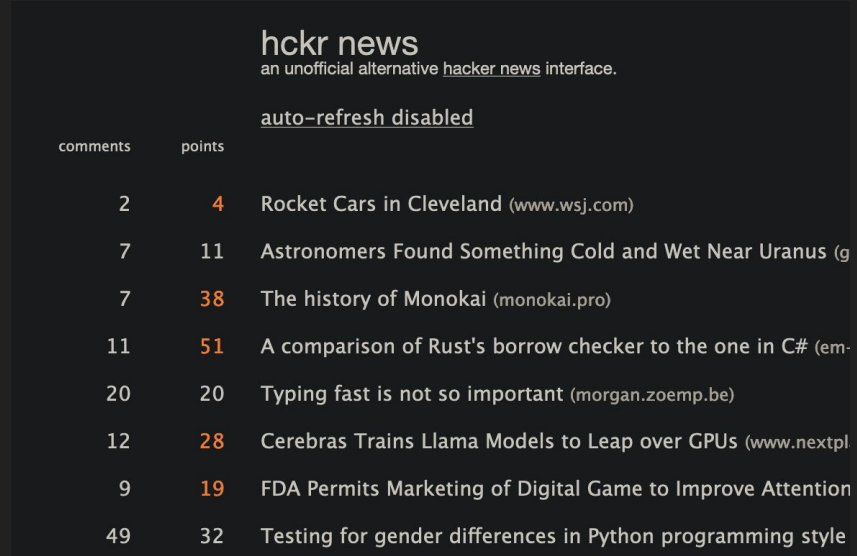
# APIs Done Right | Github

- Latency: Webhook informs your website immediately upon some action
- Quantity: Pagination allows retrieval of all available data, like list of all stargazers
- Coverage: Covers a ton of endpoints, from repositories to available emojis
- Quality: Example code for cURL and JS

```
{
  "data": {
    "repository": {
      "stargazers": {
        "edges": [
          {
            "node": {
              "login": "friedkeenan"
            },
            "starredAt": "2020-11-10T20:20:59Z"
          },
          {
            "node": {
              "login": "noahc3"
            },
            "starredAt": "2020-11-11T01:21:31Z"
          },
          {
            "node": {
              "login": "XorTroll"
            },
            "starredAt": "2020-11-11T20:52:36Z"
          },
          {
            "node": {
              "login": "Pysis868"
            },
            "starredAt": "2020-11-11T20:52:36Z"
          }
        ]
      }
    }
  }
}
```

# APIs Done Right | Hacker News

- Latency: Low latency and small payloads
- Quantity: Provides public, incrementing, ID you can use to walk backwards
- Coverage: Advertised as “a dump of our in-memory data structures”, has been used to create a custom client
- Quality: Prefixed endpoints to preserve compatibility



**hckr news**  
an unofficial alternative [hacker news](#) interface.

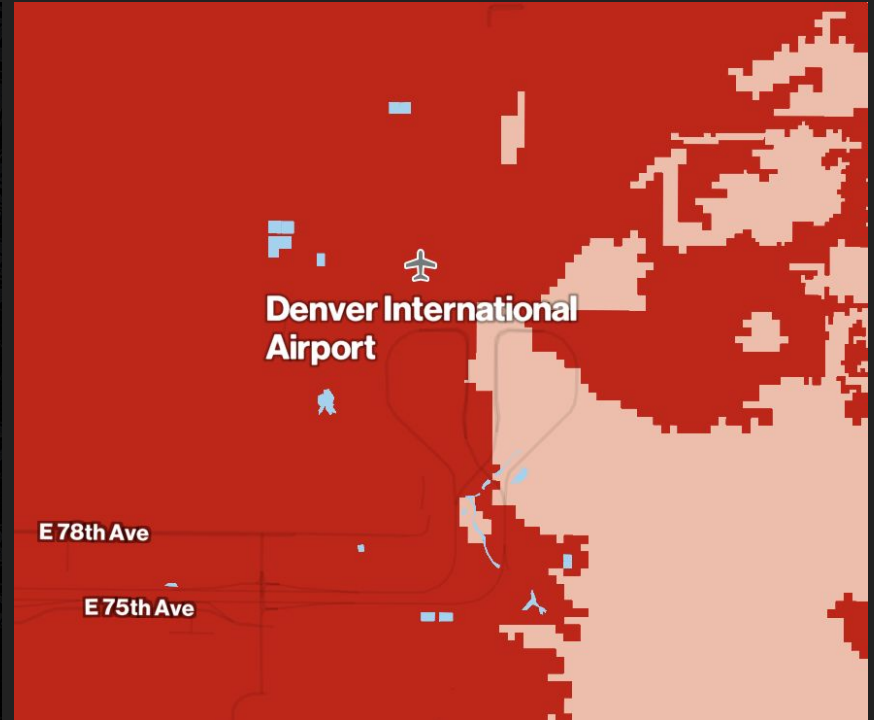
[auto-refresh disabled](#)

comments	points	
2	4	Rocket Cars in Cleveland ( <a href="#">www.wsj.com</a> )
7	11	Astronomers Found Something Cold and Wet Near Uranus ( <a href="#">g</a> )
7	38	The history of Monokai ( <a href="#">monokai.pro</a> )
11	51	A comparison of Rust's borrow checker to the one in C# ( <a href="#">em</a> -
20	20	Typing fast is not so important ( <a href="#">morgan.zoemp.be</a> )
12	28	Cerebras Trains Llama Models to Leap over GPUs ( <a href="#">www.nextpl</a>
9	19	FDA Permits Marketing of Digital Game to Improve Attention
49	32	Testing for gender differences in Python programming style

# Expose All the Data!

World Records Worldwide

Cup	Track	Time	Player	Nation
	Mario Kart Stadium	1'34"042 	Rai-Oh	
	Water Park	1'38"816	jacob	
	Sweet Sweet Canyon	1'47"589	Shira	
	Thwomp Ruins	1'45"499	jacob	
	Mario Circuit	1'42"913	Shira	
	Toad Harbor	1'53"947	jacob	
	Twisted Mansion	1'53"646	Fi <sup>[alt]</sup> <sub>m</sub>	
	Shy Guy Falls	1'51"596	jacob	
	Sunshine Airport	1'54"623	Alberto	
	Dolphin Shoals	1'56"984	Darren	
	Electrodrome	1'54"706	Fi <sup>[alt]</sup>	
	Mount Wario	1'39"450	jacob	
	Cloudtop Cruise	1'56"353 	ROA+どらさ	
	Bone-Dry Dunes	1'46"527	jacob	
	Bowser's Castle	1'57"758	jacob	
	Rainbow Road	1'59"170	jacob	
	Wii Moo Moo Meadows	1'20"283	Kyle Wade	
	GBA Mario Circuit	1'20"357	-	
	DS Cheep Cheep Beach	1'42"852	さーもんほうさく	
	NG4 Toad's Turnpike	1'38"961	m <sup>[alt]</sup>	



# Credits

- <https://tgrcode.com/>
  - [https://tgrcode.com/posts/mario\\_maker\\_2\\_api](https://tgrcode.com/posts/mario_maker_2_api)
  - [https://tgrcode.com/posts/mario\\_maker\\_2\\_datasets](https://tgrcode.com/posts/mario_maker_2_datasets)
  - [https://tgrcode.com/posts/reverse\\_engineering\\_google\\_streetview](https://tgrcode.com/posts/reverse_engineering_google_streetview)
- <https://smm2.wizul.us/>
- <https://search.arin.net/rdap/>
- <https://www.arin.net/reference/research/bulkwhois/>
- <https://xkcd.com/195/>
- <https://developers.google.com/maps/documentation/elevation/start>
- <https://docs.github.com/en/rest>
- <https://github.com/HackerNews/API>
  - <https://hckrnews.com/>
- <https://www.mkdatabase.com/>
- <https://www.verizon.com/coverage-map/>