

Tracing the Black Box of Serverless

matt williams – evangelist @ datadog









Mar 1, 2019


OPEN



Baby Diaries Global


Feb 27, 2019






Baby Tracker.


Feb 27, 2019






Huckleberry Baby Sleep Tracker


Feb 27, 2019

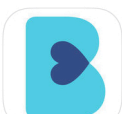




Sprout Baby (Baby Tracker)


Feb 27, 2019






BlueSmart Baby Tracker

Feb 27, 2019





Hatch Baby

Feb 15, 2019

OPEN



Baby Stats App

Feb 14, 2019



3:28

Search

Add Nursing

Nursing

Bottle

Feeding Amount

Ex. 10 oz

Start Date / Time

March 11, 2019 at 3:28 PM

Total Duration

Left: 12 min

Right: 12 min

Last Side

Left

Right

Notes (optional)

Ex. Spit up between sides.

Add Nursing Entry

Dashboard

Feedings

Sleep

Weight

More

S

1

I can build a better app

Matt Williams

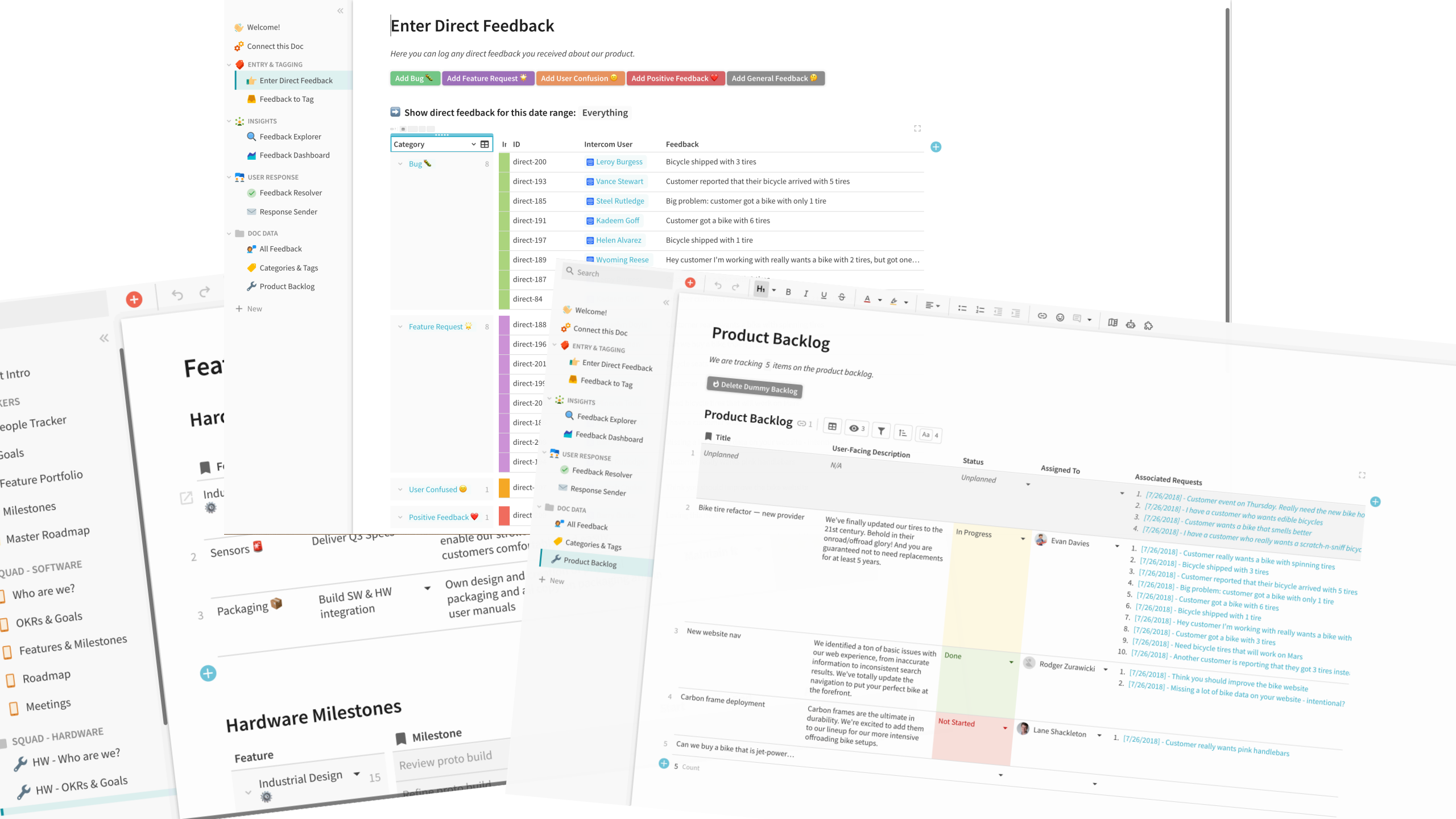
Evangelist at Datadog

@technovangelist

mattw@datadoghq.com

youtube.com/technovangelist





Enter Direct Feedback

Here you can log any direct feedback you received about our product.

[Add Bug](#) [Add Feature Request](#) [Add User Confusion](#) [Add Positive Feedback](#) [Add General Feedback](#)

Show direct feedback for this date range: Everything

Category	ID	Intercom User	Feedback
Bug	direct-200	Leroy Burgess	Bicycle shipped with 3 tires
Bug	direct-193	Vance Stewart	Customer reported that their bicycle arrived with 5 tires
Bug	direct-185	Steel Rutledge	Big problem: customer got a bike with only 1 tire
Bug	direct-191	Kadeem Goff	Customer got a bike with 6 tires
Bug	direct-197	Helen Alvarez	Bicycle shipped with 1 tire
Bug	direct-189	Wyoming Reese	Hey customer I'm working with really wants a bike with 2 tires, but got one...
Bug	direct-187		
Bug	direct-84		
Feature Request	direct-188		
Feature Request	direct-196		
Feature Request	direct-201		
Feature Request	direct-195		
Feature Request	direct-20		
Feature Request	direct-18		
Feature Request	direct-2		
Feature Request	direct-1		
User Confused	direct-		
Positive Feedback	direct		

Product Backlog

We are tracking 5 items on the product backlog.

Delete Dummy Backlog

Product Backlog

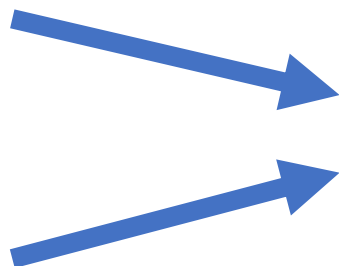
Title	User-Facing Description	Status	Assigned To	Associated Requests
1. Unplanned	N/A	Unplanned		1. [7/26/2018] - Customer event on Thursday. Really need the new bike ha 2. [7/26/2018] - I have a customer who wants edible bicycles 3. [7/26/2018] - Customer wants a bike that smells better 4. [7/26/2018] - I have a customer who really wants a scratch-n-sniff bicyc
2. Bike tire refactor — new provider	We've finally updated our tires to the 21st century. Behold in their onroad/offroad glory! And you are guaranteed not to need replacements for at least 5 years.	In Progress	Evan Davies	1. [7/26/2018] - Customer really wants a bike with spinning tires 2. [7/26/2018] - Bicycle shipped with 3 tires 3. [7/26/2018] - Customer reported that their bicycle arrived with 5 tires 4. [7/26/2018] - Big problem: customer got a bike with only 1 tire 5. [7/26/2018] - Customer got a bike with 6 tires 6. [7/26/2018] - Bicycle shipped with 1 tire 7. [7/26/2018] - Hey customer I'm working with really wants a bike with 8. [7/26/2018] - Customer got a bike with 3 tires 9. [7/26/2018] - Need bicycle tires that will work on Mars 10. [7/26/2018] - Another customer is reporting that they got 3 tires inste-
3. New website nav	We identified a ton of basic issues with our web experience, from inaccurate information to inconsistent search results. We've totally update the navigation to put your perfect bike at the forefront.	Done	Rodger Zurawicki	1. [7/26/2018] - Think you should improve the bike website 2. [7/26/2018] - Missing a lot of bike data on your website - intentional?
4. Carbon frame deployment	Carbon frames are the ultimate in durability. We're excited to add them to our lineup for our more intensive offroading bike setups.	Not Started	Lane Shackleton	1. [7/26/2018] - Customer really wants pink handlebars
5. Can we buy a bike that is jet-power...				

Hardware Milestones

Feature	Milestone
Industrial Design	Review proto build



amazon alexa



coda

Search...

Getting Started

Using the API >

Authentication

DOCS

Docs v

GET List available docs

POST Create a doc

GET Get info about a doc

DOC STRUCTURE

Sections >

Folders >

TABLES

Tables >

Columns >

Rows >

FORMULAS & CONTROLS

Formulas >

Controls >

MISCELLANEOUS

Account >

Miscellaneous >

Get info about a doc

GET /docs/{docId} v

Returns metadata for the specified doc.

AUTHORIZATIONS:

Bearer

PATH PARAMETERS

docId string
required Example: "AbCDeFGH"
ID of the doc.

Responses

- 200 Basic Coda doc metadata.
- 401 The API token is invalid or has expired.
- 404 The resource could not be located with the current API token.
- 429 The client has sent too many requests.

Request samples

Python 3.7 Shell Google Apps Script

```
import requests

headers = {'Authorization': 'Bearer <your API token>'}
uri = f'https://coda.io/apis/v1betal/docs/<doc ID>'
res = requests.get(uri, headers=headers).json()

print(f'The name of the doc is {res["name"]}')
# => The name of the doc is New Document
```

Response samples

200 401 404 429

```
application/json

{
  "id": "AbCDeFGH",
  "type": "doc",
  "href": "https://coda.io/apis/v1betal/docs/AbCDeFGH",
  "browserLink": "https://coda.io/d/_dAbCDeFGH",
  "name": "Product Launch Hub",
  "owner": "user@example.com",
  "sourceDoc": {
    "id": "AbCDeFGH",
    "type": "doc",
    "href": "https://coda.io/apis/v1betal/docs/AbCDeFGH"
  },
  "createdAt": "2018-04-11T00:18:57.946Z",
  "updatedAt": "2018-04-11T00:18:57.946Z"
}
```


How to get data into Coda (API)

1. Provide API Token
2. Get Document ID
3. Get Table ID
4. Get Column IDs
5. POST data


```
def putItem(event, context):  
    docId = getDocumentId()  
    amount = json.loads(event['body'])['amount']  
    tableId = getTableId()  
    columns = getColumns(tableId)  
    timeId = next((x for x in columns if x["name"] == "Date/Time"), None)["id"]  
    amountId = next((x for x in columns if x["name"] == "Amount"), None)["id"]  
    unitsId = next((x for x in columns if x["name"] == "Units"), None)["id"]  
    postToCoda(tableId, timeId, amountId, unitsId, amount)
```

```
def getDocumentId():  
    headers = {'Authorization': f'Bearer {os.environ["codaapitoken"]}' }  
    uri = f'https://coda.io/apis/v1beta1/docs'  
    res = requests.get(uri, headers=headers).json()  
    docid = next((x for x in res["items"] if x["name"] == "Baby Data"), None)["id"]  
    return docid
```


How I use this





Search

H₁ ▾

B

I

U

A ▾



Section 1

+ New

Bottles

	Local Time	Amount	Units	
1	3/10/2019, 9:06:53 PM	2.8	oz	
2	3/6/2019, 8:37:00 AM	3.8	oz	
3	3/6/2019, 3:00:00 AM	3	oz	
4	3/5/2019, 7:46:00 PM	2	oz	
5	3/5/2019, 5:39:00 PM	3.5	oz	
6	3/10/2019, 9:25:21 PM	3	oz	
7	3/10/2019, 10:41:36 PM	3.5	oz	
8	3/11/2019, 4:13:37 AM	3	oz	
9	3/11/2019, 6:32:48 PM	1.4	oz	
10	3/11/2019, 6:36:05 PM	2.8	oz	
11	3/11/2019, 6:37:21 PM	2.5	oz	



It works great

No errors...

Data shows up where its
supposed to

But it feels slow

How do I find out where the problem is...

...when there is no problem

Three pillars of observability

- Metrics
 - Spotting trends/patterns
 - Send alerts
 - Catch known unknowns
- Logs
- Traces

Save

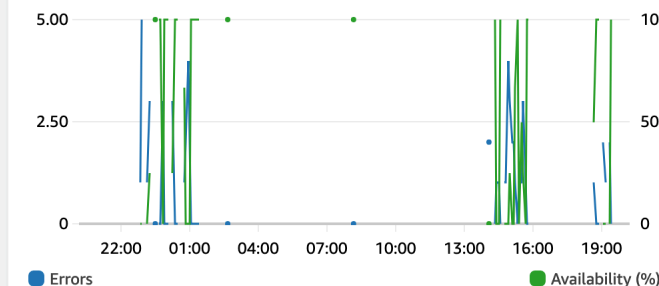
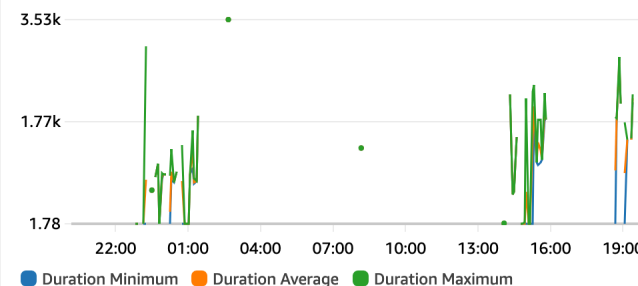
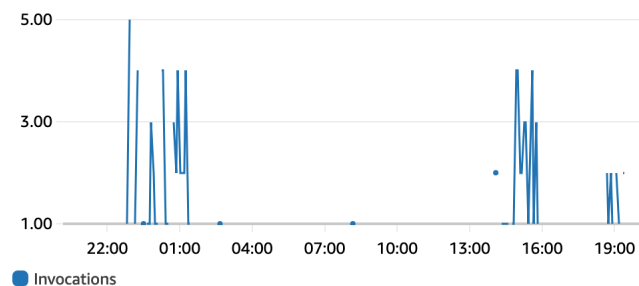
 This function belongs to the AWS CloudFormation stack **babydata-dev**. [Manage this stack](#) on the CloudFormation console.

Monitoring

CloudWatch metrics at a glance

View traces in X-Ray

1h 3h 12h **1d** 3d 1w custom ▾



```
def putItem(event, context):  
    docId = getDocumentId()  
    amount = json.loads(event['body'])['amount']  
  
    cloudwatch = boto3.client('cloudwatch')  
    cloudwatch.put_metric_data(  
        MetricData=[  
            {  
                'MetricName': 'Amounts',  
                'Unit': 'None',  
                'Value': float(amount)  
            },  
        ],  
        Namespace='BabyStats'  
    )
```

CloudWatch

Dashboards

Alarms

ALARM

INSUFFICIENT

OK

Billing

Events

Rules

Event Buses

Logs

Insights

Metrics

Favorites

Bottle Amounts

1h 3h 12h 1d 3d 1w custom (15m)

Stacked area

Actions



No unit

2.80

1.40

0

22:24

22:25

22:26

22:27

22:28

22:29

22:30

22:31

22:32

22:33

22:34

22:35

22:36

22:37

22:38

Amounts

All metrics

Graphed metrics (1)

Graph options

Source

+ Add a math expression ?

Statistic: Average

Period: 30 Seconds

Remove all



Label

Details

Statistic

Period

Y Axis

Actions



Amounts

BabyStats • Amounts

Average

30 Seconds



Relevant metrics to help me

- Total ounces eaten
- Performance timings for each function

Three pillars of observability

- Metrics
- Logs
 - Structured & easy to grep
 - Find the details of an event
 - Catch the unknown unknowns
- Traces

CloudWatch

Dashboards

Alarms

ALARM

1

INSUFFICIENT

4

OK

23

Billing

Events

Rules

Event Buses

Logs

Insights

Metrics

Favorites

CloudWatch > Log Groups > /aws/lambda/babydata-dev-putItem > All streams

Expand all



Row



Text



Filter events

all

30s

5m

1h

6h

1d

1w

custom (2h) ▾

	Time (UTC +00:00)	Message	Show in stream
2019-03-12			
No older events found for the selected date range. Adjust the date range.			
▶	00:29:04	START RequestId: d6c743f1-2c39-4ec2-a6e6-b60245e23958 Version: \$LATEST	2019/03/12/[\$LATEST]c41f188...
▶	00:29:05	END RequestId: d6c743f1-2c39-4ec2-a6e6-b60245e23958	2019/03/12/[\$LATEST]c41f188...
▶	00:29:05	REPORT RequestId: d6c743f1-2c39-4ec2-a6e6-b60245e23958 Duration: 753.08 ms Billed I	2019/03/12/[\$LATEST]c41f188...
▶	00:29:11	START RequestId: 6688d82c-d515-4c27-bcfb-991a60a35f3b Version: \$LATEST	2019/03/12/[\$LATEST]c41f188...
▶	00:29:12	END RequestId: 6688d82c-d515-4c27-bcfb-991a60a35f3b	2019/03/12/[\$LATEST]c41f188...
▶	00:29:12	REPORT RequestId: 6688d82c-d515-4c27-bcfb-991a60a35f3b Duration: 1191.77 ms Billed	2019/03/12/[\$LATEST]c41f188...
▶	01:35:27	START RequestId: c96f1644-e487-446e-a57f-30c11e41e4e9 Version: \$LATEST	2019/03/12/[\$LATEST]dcdb21f...
▶	01:35:29	END RequestId: c96f1644-e487-446e-a57f-30c11e41e4e9	2019/03/12/[\$LATEST]dcdb21f...
▶	01:35:29	REPORT RequestId: c96f1644-e487-446e-a57f-30c11e41e4e9 Duration: 1472.01 ms Billed	2019/03/12/[\$LATEST]dcdb21f...
▶	02:23:12	START RequestId: 7da58a9f-35cd-42de-85e3-6f2ec472308d Version: \$LATEST	2019/03/12/[\$LATEST]b0065a7...
▶	02:23:13	END RequestId: 7da58a9f-35cd-42de-85e3-6f2ec472308d	2019/03/12/[\$LATEST]b0065a7...
▶	02:23:13	REPORT RequestId: 7da58a9f-35cd-42de-85e3-6f2ec472308d Duration: 748.86 ms Billed C	2019/03/12/[\$LATEST]b0065a7...
▶	02:26:33	START RequestId: b6c54194-735a-4d51-a197-73f2c04fbffc Version: \$LATEST	2019/03/12/[\$LATEST]b0065a7...
▶	02:26:34	END RequestId: b6c54194-735a-4d51-a197-73f2c04fbffc	2019/03/12/[\$LATEST]b0065a7...
▶	02:26:34	REPORT RequestId: b6c54194-735a-4d51-a197-73f2c04fbffc Duration: 728.06 ms Billed D	2019/03/12/[\$LATEST]b0065a7...
No newer events found for the selected date range. Adjust the date range.			

```
def getColumns(tableid, docid):  
    headers = {'Authorization': f'Bearer {os.environ["codaapitoken"]}' }  
    uri = f'https://coda.io/apis/v1beta1/docs/{docid}/tables/{tableid}/columns'  
    res = requests.get(uri, headers=headers).json()  
    print(f'Columns for table: {tableid} -- {res["items"]}')  
    return res["items"]
```



CloudWatch

Dashboards

Alarms

ALARM

INSUFFICIENT

OK

Billing

Events

Rules

Event Buses

Logs

Insights

Metrics

Favorites

CloudWatch > Log Groups > /aws/lambda/babydata-dev-pullItem > All streams

Expand all



Row



Text



Filter events

all 30s 5m 1h 6h 1d 1w custom ▾

Time (UTC +00:00)	Message	Show in stream
2019-03-11		
No older events found for the selected date range. Adjust the date range.		
▶ 23:56:08	START RequestId: 4944bc29-ebbb-4a4c-9bd3-846108b3f28d Version: \$LATEST	2019/03/11/[\$LATEST]31c2cd9...
▶ 23:56:08	[ERROR] Runtime.UserCodeSyntaxError: Syntax error in module 'handler': f-string: expectin	2019/03/11/[\$LATEST]31c2cd9...
▶ 23:56:08	END RequestId: 4944bc29-ebbb-4a4c-9bd3-846108b3f28d	2019/03/11/[\$LATEST]31c2cd9...
▶ 23:56:08	REPORT RequestId: 4944bc29-ebbb-4a4c-9bd3-846108b3f28d Duration: 2.06 ms Billed D	2019/03/11/[\$LATEST]31c2cd9...
▶ 23:58:02	START RequestId: 13700d00-22cd-4731-9522-109a304d5966 Version: \$LATEST	2019/03/11/[\$LATEST]3e4f111...
▼ 23:58:04	Columns for table: grid-VrgyrNH3as - [{'id': 'c-E5zQPiu5Vo', 'type': 'column', 'name': 'Date/	2019/03/11/[\$LATEST]3e4f111...
Columns for table: grid-VrgyrNH3as - [{'id': 'c-E5zQPiu5Vo', 'type': 'column', 'name': 'Date/Time', 'href': 'https://coda.io/apis/v1beta1/docs/4qnPRKi-2c/tables/grid-VrgyrNH3as/columns/c-E5zQPiu5Vo', 'display': True}, {'id': 'c-VebmUxm5aj', 'type': 'column', 'name': 'Amount', 'href': 'https://coda.io/apis/v1beta1/docs/4qnPRKi-2c/tables/grid-VrgyrNH3as/columns/c-VebmUxm5aj'}, {'id': 'c-ycjixf1rFv', 'type': 'column', 'name': 'Units', 'href': 'https://coda.io/apis/v1beta1/docs/4qnPRKi-2c/tables/grid-VrgyrNH3as/columns/c-ycjixf1rFv'}, {'id': 'c-M6q614WT-o', 'type': 'column', 'name': 'Local Time', 'href': 'https://coda.io/apis/v1beta1/docs/4qnPRKi-2c/tables/grid-VrgyrNH3as/columns/c-M6q614WT-o', 'calculated': True}]		
▶ 23:58:05	END RequestId: 13700d00-22cd-4731-9522-109a304d5966	2019/03/11/[\$LATEST]3e4f111...
▶ 23:58:05	REPORT RequestId: 13700d00-22cd-4731-9522-109a304d5966 Duration: 2746.22 ms Bille	2019/03/11/[\$LATEST]3e4f111...
No newer events found for the selected date range. Adjust the date range.		



Relevant log lines for me

- Function entry and exit timestamps, total time
- What is being collected by each step

Three pillars of observability

- Metrics
- Logs
- Traces
 - Request-based
 - Follow activity across function and service
 - Answers “Where?” and “How long?”

BASIC SETTINGS

AWS CloudTrail can log this function's invocations for operational and risk auditing, governance, and compliance. [Get started](#) on the CloudTrail console.

```
service: • babydata •
plugins:
• • - • serverless-pseudo-parameters
• • - • serverless-plugin-tracing
• • - • serverless-python-requirements
provider:
• • name: • aws
• • runtime: • python3.7
• • tracing: • true
• • environment: •
• • • codaapitoken: • ${env:CODAAPITOKEN}
```



AWS X-Ray

Getting started

Service map

Traces

Configuration

Sampling

Encryption



Default ▾



Enter service name, annotation, trace ID. Or click the Help icon for additional details.



Last 1 minute ▾



Service map

Updated on 2019-03-11 22:27:13 (UTC -04:00)

Enter a service name to find and select the node on map

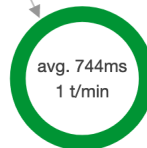
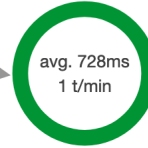


AWS::Lambda

Clients

WildRydes/prod
AWS::ApiGateway::Stagelambdatolambda tester
AWS::Lambda::Function

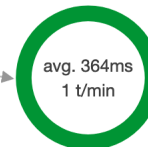
Clients

dev-babydata/dev
AWS::ApiGateway::Stagebabydata-dev-putItem
AWS::Lambdababydata-dev-putItem
AWS::Lambda::Function

Clients

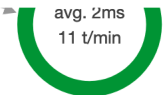


pinkerton-datadog-dev-hello



pinkerton-datadog-dev-hello

Map legend ⓘ

LangerCustomMetricPutter
AWS::Lambda::Function

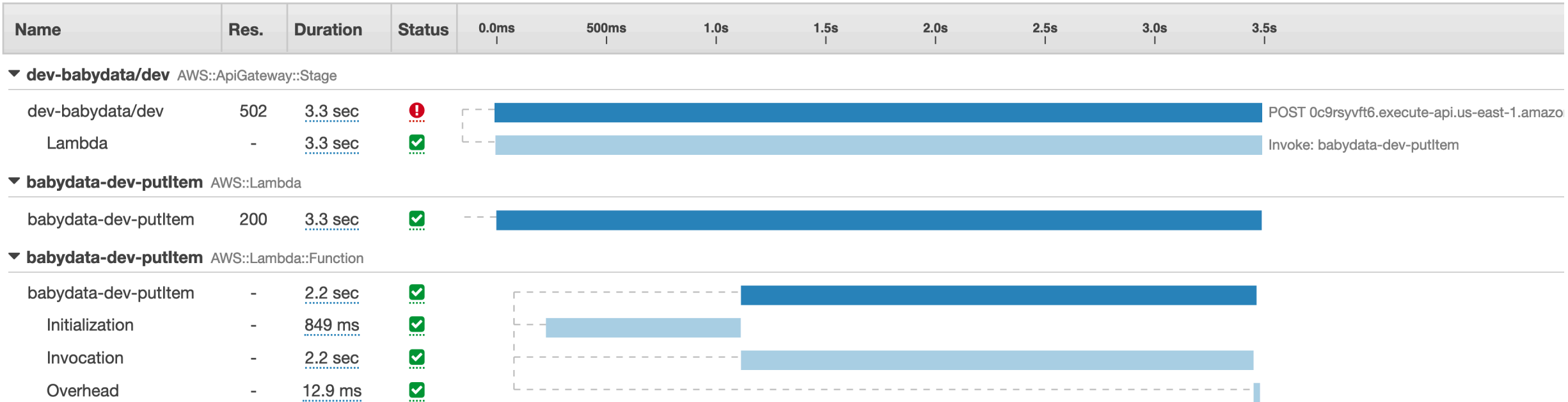
1-5c86f97f-8fa49c5ef70eeae4c9ce0d6c

[Traces](#) > Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	502	3.3 sec	12.7 sec (2019-03-12 00:12:47 UTC)	1-5c86f97f-8fa49c5ef70eeae4c9ce0d6c



Q 1-5c86f97f-8fa49c5ef70eeae4c9ce0d6c



Traces > Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	502	3.3 sec	12.7 sec (2019-03-12 00:12:47 UTC)	1-5c86f97f-8fa49c5ef70eeae4c9ce0d6c

Name	Res.	Duration	Status	0.0ms	500ms	1.0s	1.5s	2.0s	2.5s	3.0s	3.5s
------	------	----------	--------	-------	-------	------	------	------	------	------	------

▼ dev-babydata/dev AWS::ApiGateway::Stage

dev-babydata/dev	502	3.3 sec	❗	POST 0c9rsyvft6.execute-api.us-east-1.amazo							
Lambda	-	3.3 sec	✅	Invoke: babydata-dev-putItem							

▼ babydata-dev-putItem AWS::Lambda

babydata-dev-putItem	200	3.3 sec	✅								
----------------------	-----	-------------------------	---	--	--	--	--	--	--	--	--

▼ babydata-dev-putItem AWS::Lambda::Function

babydata-dev-putItem	-	2.2 sec	✅								
Initialization	-	849 ms	✅								
Invocation	-	2.2 sec	✅								
Overhead	-	12.9 ms	✅								

Q 1-5c86b693-13104c7bd86be3535db8a384



Traces > Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	502	2.1 sec	4.7 hr (2019-03-11 19:27:15 UTC)	1-5c86b693-13104c7bd86be3535db8a384

Name	Res.	Duration	Status	0.0ms	200ms	400ms	600ms	800ms	1.0s	1.2s	1.4s	1.6s	1.8s	2.0s	2.2s
------	------	----------	--------	-------	-------	-------	-------	-------	------	------	------	------	------	------	------

▼ dev-babydata/dev AWS::ApiGateway::Stage

dev-babydata/dev	502	2.1 sec														POST 0c9rsyvft6.execute-api.us-east-1.amazon
Lambda	-	2.1 sec														Invoke: babydata-dev-putItem

▼ babydata-dev-putItem AWS::Lambda

babydata-dev-putItem	200	2.1 sec														
----------------------	-----	-------------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--

▼ babydata-dev-putItem AWS::Lambda::Function

babydata-dev-putItem	-	2.1 sec														
Invocation	-	2.1 sec														
Overhead	-	2.9 ms														

But I want more visibility into
what is going on

⊕ `def putItem(event, context): ...`

⊕ `def getDocumentId(): ...`

⊕ `def getTableId(): ...`

⊕ `def getColumns(tableid, docid): ...`

⊕ `def postToCoda(tableid, timeid, amountid, unitsid, amount): ...`

```
@xray_recorder.capture('Put · Bottle · into · Coda')
```

```
+ def · putItem(event, · context): ...
```

```
@xray_recorder.capture('Get · DocumentID')
```

```
+ def · getDocumentId(): ...
```

```
@xray_recorder.capture('Get · TableID')
```

```
+ def · getTableId(): ...
```

```
@xray_recorder.capture('Get · Columns')
```

```
+ def · getColumns(tableid, · docid): ...
```

```
@xray_recorder.capture('Post · to · Coda')
```

```
+ def · postToCoda(tableid, · timeid, · amountid, · unitsid, · amount): ...
```


Q 1-5c86fb1a-4ac6e39814df2d54738cb3cc



Traces > Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	502	3.6 sec	13.4 sec (2019-03-12 00:19:38 UTC)	1-5c86fb1a-4ac6e39814df2d54738cb3cc

Name	Res.	Duration	Status	0.0ms	500ms	1.0s	1.5s	2.0s	2.5s	3.0s	3.5s	4.0s
------	------	----------	--------	-------	-------	------	------	------	------	------	------	------

▼ dev-babydata/dev AWS::ApiGateway::Stage

dev-babydata/dev	502	3.7 sec	!										POST 0c9rsyvft6.execute-api.us-east-1.am
Lambda	-	3.6 sec	✓										Invoke: babydata-dev-putItem

▼ babydata-dev-putItem AWS::Lambda::Function

babydata-dev-putItem	-	3.6 sec	✓										
Invocation	-	3.6 sec	✓										
Put Bottle into Coda	-	3.6 sec	✓										
Get DocumentID	-	450 ms	✓										
Get TableID	-	439 ms	✓										
Get Columns	-	476 ms	✓										
Post to Coda	-	2.2 sec	✓										
Overhead	-	0.5 ms	✓										

▼ babydata-dev-putItem AWS::Lambda

babydata-dev-putItem	200	3.6 sec	✓										
----------------------	-----	---------	---	--	--	--	--	--	--	--	--	--	--



Time wasted getting static values

```
def putItem(event, context):  
    print(event)  
    docId = os.environ["babydocid"]  
    timeId = 'c-E5zQPxiu5Vo'  
    amountId = 'c-VebmxUxm5aj'  
    unitsId = 'c-ycjixfx1rFv'  
    tableId = 'grid-VrgxyrNH3as'  
    amount = json.loads(event['body'])['amount']  
    # docId = getDocumentId()  
    # tableId = getTableId()  
    # columns = getColumns(tableId, docId)  
    # timeId = next((x for x in columns if x["name"] == "Date/Time"), None)["id"]  
    # amountId = next((x for x in columns if x["name"] == "Amount"), None)["id"]  
    # unitsId = next((x for x in columns if x["name"] == "Units"), None)["id"]  
    postToCoda(tableId, timeId, amountId, unitsId, amount)  
    return {  
        "statusCode": 200,  
        "body": "complete"  
    }
```

1-5c86fd57-54fa767af2453f069399c892

[Traces](#) > Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	502	1.2 sec	12.1 sec (2019-03-12 00:29:11 UTC)	1-5c86fd57-54fa767af2453f069399c892

Name	Res.	Duration	Status	0.0ms	100ms	200ms	300ms	400ms	500ms	600ms	700ms	800ms	900ms	1.0s	1.1s	1.2s	1.3s
------	------	----------	--------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------	------	------	------

▼ **dev-babydata/dev** AWS::ApiGateway::Stage

dev-babydata/dev	502	1.2 sec																POST 0c9rsyvft6.execute-api.us-east-1.am
Lambda	-	1.2 sec																Invoke: babydata-dev-putItem

▼ **babydata-dev-putItem** AWS::Lambda::Function

babydata-dev-putItem	-	1.2 sec																
Invocation	-	1.2 sec																
Put Bottle into Coda	-	1.2 sec																
Post to Coda	-	1.2 sec																
Overhead	-	4.6 ms																

▼ **babydata-dev-putItem** AWS::Lambda

babydata-dev-putItem	200	1.2 sec																
----------------------	-----	-------------------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

But I need more data

```
... xray_recorder.put_annotation("tableid", tableid)
```

```
... xray_recorder.put_metadata("tableid", tableid)
```

```
...
```

Overview

Resources

Annotations

Metadata

Exceptions

Key

Value

tableid	grid-VrgyrNH3as
---------	-----------------

Overview

Resources

Annotations

Metadata

Exceptions

```
{
  "default": {
    "tableid": "grid-VrgyrNH3as"
  }
}
```


- AWS X-Ray
- Getting started
- Service map
- Traces
- Configuration
- Sampling
- Encryption

1-5c87d521-3be786c35f9791120bc4592e

?

↺

Traces

Details

Timeline

Raw data

Method	Response	Duration	Age	ID
POST	200	2.5 sec	20.8 sec (2019-03-12 15:49:53 UTC)	1-5c87d521-3be786c35f9791120bc4592e

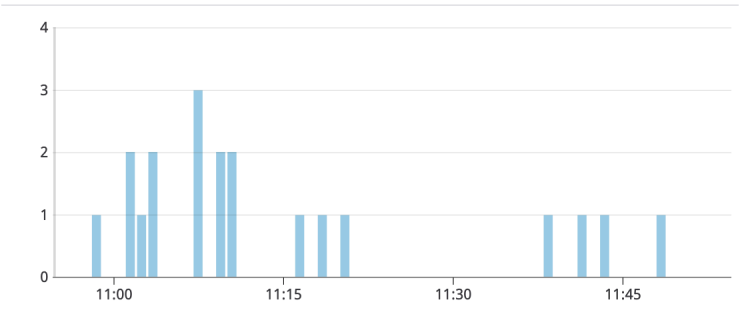
Name	Res.	Duration	Status	
▼ dev-babydata/dev AWS::ApiGateway::Stage				
dev-babydata/dev	200	2.5 sec	✓	POST 0c9rsyvft6.execute-api.us-east-1.amazonaws.co...
Lambda	200	2.5 sec	✓	Invoke: babydata-dev-putItem
▼ babydata-dev-putItem AWS::Lambda				
babydata-dev-putItem	200	2.5 sec	✓	
▼ babydata-dev-putItem AWS::Lambda::Function				
babydata-dev-putItem	-	2.5 sec	✓	
Invocation	-	2.5 sec	✓	
putItem	-	2.4 sec	✓	
getDocumentId	-	470 ms	✓	
https://coda.io/apis/v1beta1/docs	200	468 ms	✓	Remote: get coda.io/apis/v1beta1/docs
getTableId	-	470 ms	✓	
https://coda.io/apis/v1beta1/docs/4qnPI	200	452 ms	✓	Remote: get coda.io/apis/v1beta1/docs/4qnPRKI-2c/t...
getColumns	-	559 ms	✓	
https://coda.io/apis/v1beta1/docs/4qnPI	200	540 ms	✓	Remote: get coda.io/apis/v1beta1/docs/4qnPRKI-2c/t...
postToCoda	-	943 ms	✓	
https://coda.io/apis/v1beta1/docs/4qnPI	202	940 ms	✓	Remote: post coda.io/apis/v1beta1/docs/4qnPRKI-2c/...
Overhead	-	0.6 ms	✓	
▼ https://coda.io/apis/v1beta1/docs (Client Response)				
babydata-dev-putItem	200	468 ms	✓	
▼ https://coda.io/apis/v1beta1/docs/4qnPRKI-2c/tables/ (Client Response)				
babydata-dev-putItem	200	452 ms	✓	
▼ https://coda.io/apis/v1beta1/docs/4qnPRKI-2c/tables/grid-VrgyrNH3as/columns (Client Response)				
babydata-dev-putItem	200	540 ms	✓	
▼ https://coda.io/apis/v1beta1/docs/4qnPRKI-2c/tables/grid-VrgyrNH3as/rows (Client Response)				
babydata-dev-putItem	202	940 ms	✓	



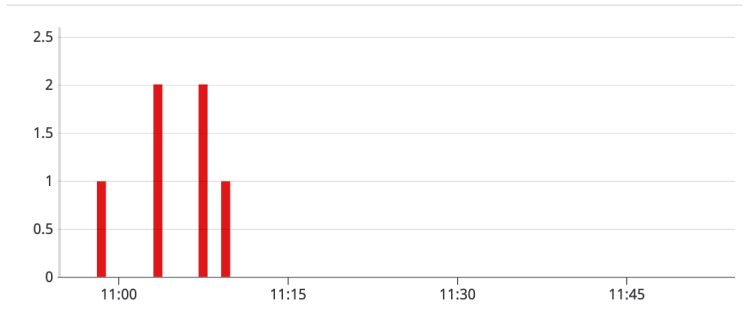
- Watchdog
- Events
- Dashboards
- Infrastructure
- Monitors
- Metrics
- Integrations
- APM
- Notebooks
- Logs
- Synthetics

- Help
- Team
- matt.williams (Datadog Demo (112...

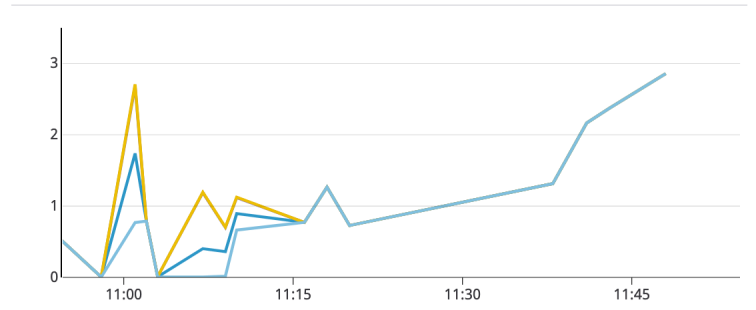
Invocations



Errors



Duration




Traces

Date ↓		Duration		Status	Latency Breakdown by %	Cold Start	Error
11:49:53 AM	5 mins ago	2.46 s		OK			
11:43:54 AM	11 mins ago	2.36 s		OK			
11:09:02 AM	46 mins ago	700 ms		OK			
11:07:38 AM	47 mins ago	3.23 ms		ERROR			
11:07:33 AM	47 mins ago	4.06 ms		ERROR			
10:58:56 AM	56 mins ago	4.80 ms		ERROR		Yes	

Logs

DATE ↓	[LAMBDA] REQUEST ID	MESSAGE
Mar 12 11:48:45.135	0e366015-6ae5-4fd4-9097-7373fc910e96	> REPORT RequestId: 0e366015-6ae5-4fd4-9097-7373fc910e96 Duration: 2858.89 ms Billed Duration: 2900 ms Memory Size: 1024 MB Max Memory Used: 51 MB
Mar 12 11:48:45.135	0e366015-6ae5-4fd4-9097-7373fc910e96	> END RequestId: 0e366015-6ae5-4fd4-9097-7373fc910e96
Mar 12 11:48:44.258		> Columns for table: grid-VrgyrNH3as - [{"id": "c-E5zQPiu5Vo", "type": "column", "name": "Date/Time", "href": "https://coda.io/apis/v1beta1/docs/4qnPRKi-2c/..."}]
Mar 12 11:48:43.382		> Table ID: grid-VrgyrNH3as
Mar 12 11:48:42.894		> Document ID: 4qnPRKi-2c
Mar 12 11:48:42.277		> Input Event: {"resource": "/bottle", "path": "/bottle", "httpMethod": "POST", "headers": {"Accept": "/*/*", "Accept-Encoding": "br, gzip, deflate", "Accept-La..."}}
Mar 12 11:48:42.276	0e366015-6ae5-4fd4-9097-7373fc910e96	> START RequestId: 0e366015-6ae5-4fd4-9097-7373fc910e96 Version: \$LATEST
Mar 12 11:38:40.215	ddf902f6-0661-4602-8604-6a7e7e6ac3ce	> REPORT RequestId: ddf902f6-0661-4602-8604-6a7e7e6ac3ce Duration: 1315.13 ms Billed Duration: 1400 ms Memory Size: 1024 MB Max Memory Used: 52 MB
Mar 12 11:38:40.215	ddf902f6-0661-4602-8604-6a7e7e6ac3ce	> END RequestId: ddf902f6-0661-4602-8604-6a7e7e6ac3ce
Mar 12 11:38:38.901		> {"resource": "/bottle", "path": "/bottle", "httpMethod": "POST", "headers": {"Accept": "/*/*", "Accept-Encoding": "br, gzip, deflate", "Accept-Language": "en-..."}}
Mar 12 11:38:38.896	ddf902f6-0661-4602-8604-6a7e7e6ac3ce	> START RequestId: ddf902f6-0661-4602-8604-6a7e7e6ac3ce Version: \$LATEST



DATADOG

Watchdog

Events

Dashboards

Infrastructure

Monitors

Metrics

Integrations

APM


Notebooks

Logs

Synthetics

Help

Team

matt.williams

(Datadog Demo (112...))

f(⚡)

aws.apigateway.stage

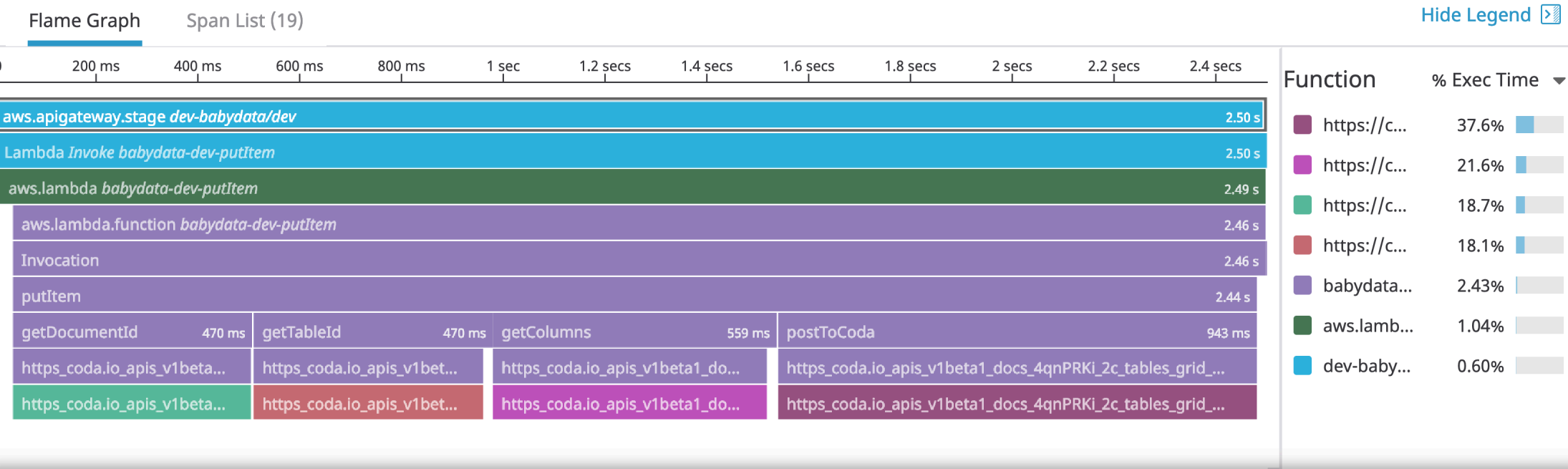
Mar 12, 11:49 am

2.50 s

POST

https://0c9rsyvft6.execute-api.us-east-1.amazonaws.co...

200 ...



f(⚡)

dev-babydata/dev

Span Metadata

Logs

```
{
  aws:api_id      0c9rsyvft6
  aws:api_stage  dev
  http {
    content_length 0
    method         POST
    status_code    200
    url            https://0c9rsyvft6.execute-api.us-east-1.amazonaws.com/dev/bottle
  }
  resource_arn    arn:aws:apigateway:us-east-1::/restapis/0c9rsyvft6/stages/dev
```

Matt Williams

Evangelist at Datadog

@technovangelist

mattw@datadoghq.com

youtube.com/technovangelist





