


# Drupal, JSON:API, and Cheap Canned Beer



Paul McKibben  
Drupalcamp Atlanta  
September 13, 2019



# About me...

---

- Working with Drupal since 2007.
- Run my own consultancy.
- Implemented sites of all sizes on Drupal 5, 6, 7, and 8.
- I like cheap canned beer.



[paul@turbojettech.com](mailto:paul@turbojettech.com)

Twitter: [@paulmckibben](https://twitter.com/paulmckibben)

[www.turbojettech.com](http://www.turbojettech.com)

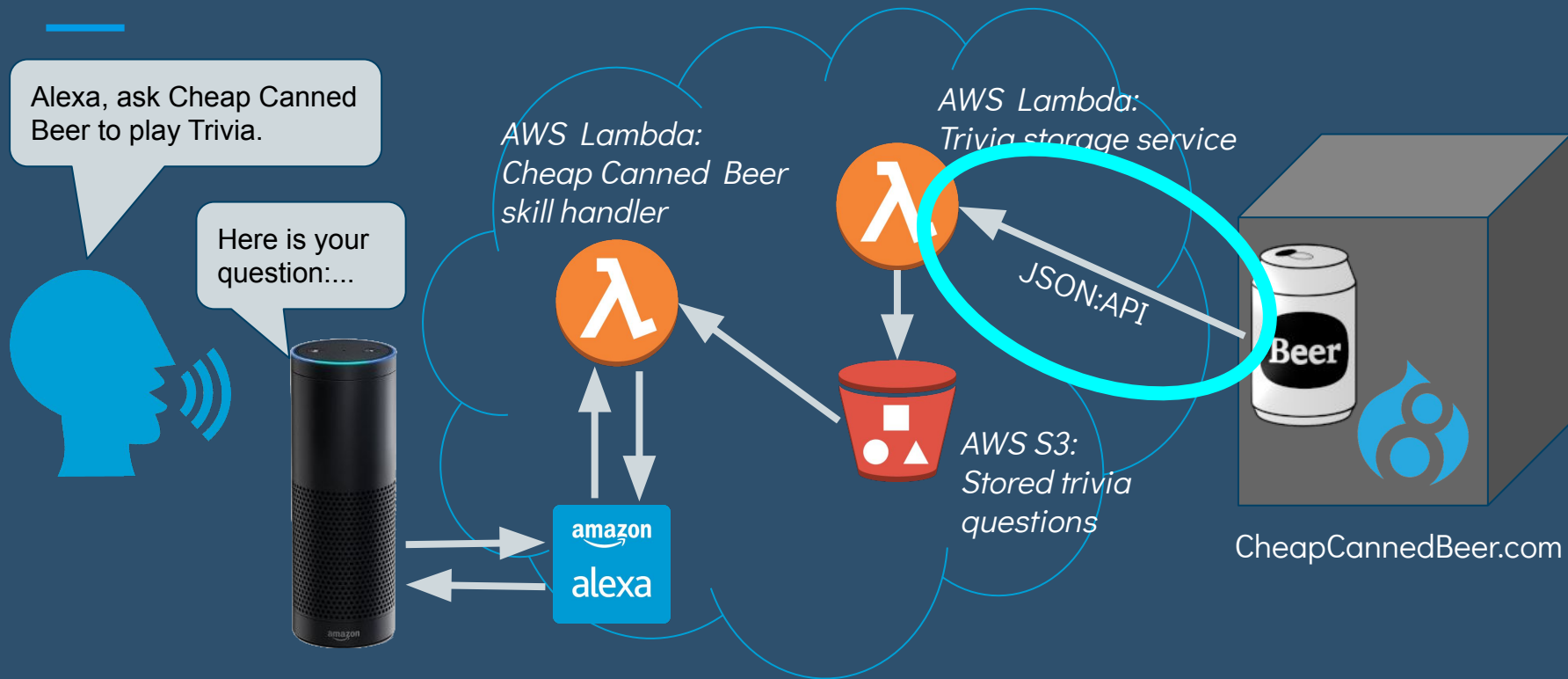
# The backstory...

- 2013: On a whim, I bought the domain cheapcannedbeer.com
- 2017: Slowly started building site in my (ahem) spare time.
- November 2017: integrated with Alexa using custom Drupal module.
- September 2019: no more custom module! Now using core JSON:API.



Image from <https://www.facebook.com/BeerCanMuseumandHallofFame> - The Beer Can Museum in East Taunton, MA

# What am I showing you today?



# What is JSON:API?

---

## **JSON**

Javascript Object Notation

## **API**

Application Programming  
Interface

## **JSON:API**

A specification for building APIs  
using JSON to create, read,  
update, and delete content.

# JSON:API? Tell me more...

---

- Enables anything to use Drupal content.
- Drupal's strength: content structure/architecture.
- Alternatives: GraphQL, core REST, custom HTTP



Dries compared REST, GraphQL, and JSON:API -  
<https://dri.es/headless-cms-rest-vs-jsonapi-vs-graphql>

# What does a JSON:API document look like?

---

```
{
  "links": {
    "self": "http://example.com/articles",
    "next": "http://example.com/articles?page[offset]=2",
    "last": "http://example.com/articles?page[offset]=10"
  },
  "data": [{
    "type": "articles",
    "id": "1",
    "attributes": {
      "title": "JSON:API paints my bikeshed!"
    },
    "relationships": {
      "author": {
        "links": {
          "self": "http://example.com/articles/1/relationships/author",
```

This is a partial example. See the full example on the homepage of <https://jsonapi.org>.

# Setting up JSON:API on Drupal

---

- In core - Drupal 8.7.x and beyond. Just enable the module!
- Defaults to read-only (HTTP GET).
- Consider `jsonapi_extras` module to refine exposed entities.
- Consider `entity_access_audit` module to review exposed entities and fields.
- If enabling PUT, POST, and DELETE, be especially careful.



# JSON:API - a Drupal example

## Good for young mothers?

### TRIVIA QUESTION

Which beer brand once placed newspaper advertisements aimed at young mothers, claiming, "the malt in the beer supplies nourishing qualities," "the hops act as an appetizing, stimulating tonic," and, "obviously baby participates in the benefits?"

- A. Pabst Blue Ribbon
- B. Blatz
- C. Olympia
- D. Old Milwaukee

Show answer

## Answer

B. Blatz



An online image search for "Blatz" will turn up multiple clippings of this advertisement aimed at nursing mothers.

# JSON:API - structured data is important

**Summary \***

**Question (Edit summary) \***  

**B** **I** Format Source

Which beer brand once placed newspaper advertisements aimed at young mothers, claiming, "the malt in the beer supplies nourishing qualities," "the hops act as an appetizing, stimulating tonic," and, "obviously baby participates in the benefits?"

Text format Basic HTML [About text formats ?](#)

**CHOICES \***

Question Choice

Choice Description \*

☐ Correct answer

Question Choice

Choice Description \*

☒ Correct answer

Question Choice

Choice Description \*

☐ Correct answer

Question Choice

Choice Description \*

☐ Correct answer


**Correct answer description**  

**B** **I** Format Source

An online image search for "Blatz" will turn up multiple clippings of this advertisement aimed at nursing mothers.

Text format Basic HTML [About text formats ?](#)

Explanation of why the correct answer is correct.

**Answer Image**  


**Alternative text \***  
  
Short description of the image used by screen readers and displayed when the image is not loaded. This is important for accessibility.

# Parts of a JSON:API document

---

data: the document's primary data, e.g. the properties/fields of a Drupal node

```
"data": {  
  "type": "node--trivia_question",  
  "id": "217def0d-6250-4008-945a-6b3ae5174597",  
  "attributes": {  
    "drupal_internal__nid": 22,  
    "drupal_internal__vid": 26,  
    "langcode": "en",  
    "revision_timestamp": "2017-11-02T15:33:08+00:00",  
    "revision_log": null,  
    "status": true,  
    "title": "Good for young mothers?",  
    "created": "2017-11-02T15:09:19+00:00",  
    "changed": "2017-11-02T15:33:08+00:00",  
    "promote": false,  
    "sticky": false,  
    "default_langcode": true,  
    "revision_translation_affected": true,  
    "path": null,  
    "body": {  
      "value": "<p>Which beer brand once placed newspaper advertisements aimed at young mothers, claiming, \"the malt in the beer supplies nourishing qualities,\" &nbsp;&nbsp;&nbsp;\"the hops act as an appetizing, stimulating tonic,\" and, \"obviously baby participates in the benefits?\"</p>\r\n",
```

# Parts of a JSON:API document

---

data : attributes      represent information about the primary object

```
"data": {
  "type": "node--trivia_question",
  "id": "217def0d-6250-4008-945a-6b3ae5174597",
  "attributes": {
    "drupal_internal__nid": 22,
    "drupal_internal__vid": 26,
    "langcode": "en",
    "revision_timestamp": "2017-11-02T15:33:08+00:00",
    "revision_log": null,
    "status": true,
    "title": "Good for young mothers?",
    "created": "2017-11-02T15:09:19+00:00",
    "changed": "2017-11-02T15:33:08+00:00",
    "promote": false,
    "sticky": false,
    "default_langcode": true,
    "revision_translation_affected": true,
    "path": null,
    "body": {
      "value": "<p>Which beer brand once placed newspaper advertisements aimed at young mothers, claiming, \"the malt in the beer supplies nourishing qualities,\" &nbsp;&nbsp;&nbsp;\"the hops act as an appetizing, stimulating tonic,\" and, \"obviously baby participates in the benefits?\"</p>\r\n",
```

# Parts of a JSON:API document

data : relationships      represent objects related to the primary object

```
"field_answer_image": {
  "data": {
    "type": "file--file",
    "id": "56d3deed-39e9-4391-9ef6-d8a7f295b78b",
    "meta": {
      "alt": "Old newspaper ad for Blatz: How Mother and Baby \"Picked Up\"",
      "title": "",
      "width": 500,
      "height": 262
    }
  },
  "links": {
    "self": {
      "href": "https://www.cheapcannedbeer.com/jsonapi/node/trivia_question/217def0d-6250-4008-945a-6b3ae5174597/relationships/field_answer_image?resourceVersion=id%3A26"
    },
    "related": {
      "href": "https://www.cheapcannedbeer.com/jsonapi/node/trivia_question/217def0d-6250-4008-945a-6b3ae5174597/field_answer_image?resourceVersion=id%3A26"
    }
  }
}
```

# Parts of a JSON:API document

---

**included** represent related objects loaded as part of the same request

```
"included": [  
  {  
    "type": "file--file",  
    "id": "56d3deed-39e9-4391-9ef6-d8a7f295b78b",  
    "attributes": {  
      "drupal_internal__fid": 3,  
      "langcode": "en",  
      "filename": "BlatzMama.jpg",  
      "uri": {  
        "value": "public://2017-11/BlatzMama.jpg",  
        "url": "/sites/default/files/2017-11/BlatzMama.jpg"  
      },  
      "filemime": "image/jpeg",  
      "filesize": 64814,  
      "status": true,  
      "created": "2017-11-02T15:32:41+00:00",  
      "changed": "2017-11-02T15:33:08+00:00"  
    },  
  },  
]
```

# JSON:API request for trivia questions

---

- To get all nodes of a certain type, HTTP GET this URL:  
`http(s)://(domain)/jsonapi/node/(type)`
- For Cheap Canned Beer trivia question nodes:  
`https://www.cheapcannedbeer.com/jsonapi/node/trivia_question`
- Problem: “reference” fields (paragraphs, files) refer to other objects!
- They need to be “included,” or we only see their IDs.

# Example: paragraph fields without “include”

---

```
"field_choices": {  
  "data": [  
    {  
      "type": "paragraph--question_choice",  
      "id": "d0efa244-a384-43c2-ad75-ad8f8faddcd3",  
      "meta": {  
        "target_revision_id": 23  
      }  
    },  
    {  
      "type": "paragraph--question_choice",  
      "id": "c6c80402-387c-4f87-903d-67e48a211d34",  
      "meta": {  
        "target_revision_id": 24  
      }  
    }  
  ]  
}
```



# JSON:API “include” parameter

---

- Use “include” to load referenced paragraphs and images in the same request.
- Specify the fields that reference other objects you need to load.
- For Cheap Canned Beer trivia question nodes:

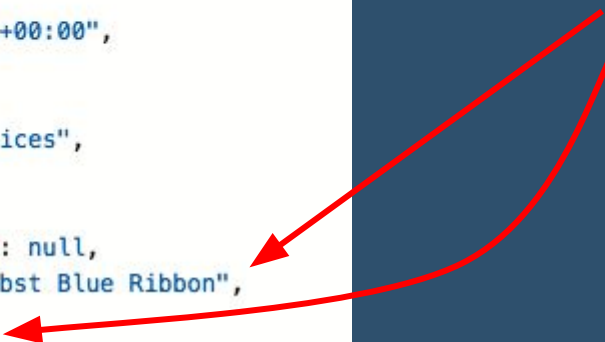
`https://www.cheapcannedbeer.com/jsonapi/node/trivia\_question?  
include=field\_choices,field\_answer\_image`

# Example: included paragraph

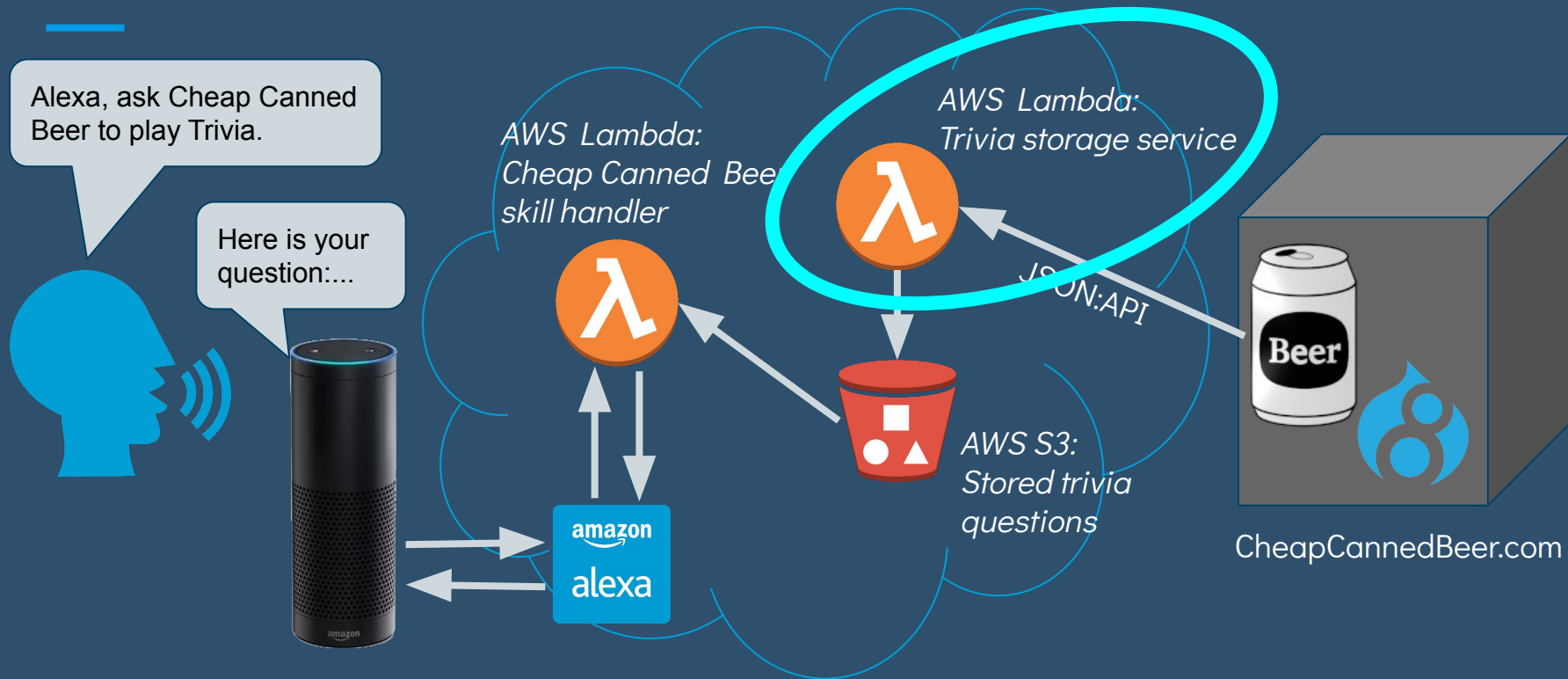
---

```
"included": [  
  {  
    "type": "paragraph--question_choice",  
    "id": "d0efa244-a384-43c2-ad75-ad8f8faddcd3",  
    "attributes": {  
      "drupal_internal__id": 15,  
      "drupal_internal__revision_id": 23,  
      "langcode": "en",  
      "status": true,  
      "created": "2017-11-02T15:12:52+00:00",  
      "parent_id": "22",  
      "parent_type": "node",  
      "parent_field_name": "field_choices",  
      "behavior_settings": [],  
      "default_langcode": true,  
      "revision_translation_affected": null,  
      "field_choice_description": "Pabst Blue Ribbon",  
      "field_correct_answer": false  
    },  
  },  
]
```

We need these!



# Next: Pulling and Parsing JSON:API data



# AWS Lambda Function

---

- “Serverless”
- Stateless
- Cheap/free way to run a small task.
- Easy to integrate with other AWS services.



Image:  
<https://gideonpyzer.dev/blog/2017/02/05/create-a-n-aws-lambda-function-in-javascript/>

# Trivia Storage Service

- Triggered by CloudWatch every 3 hours.
- Uses JSON:API to read trivia questions from Drupal.
- Parses out the important bits; removes HTML tags.
- Stores result in S3 as JSON object.
- Written in Javascript (Node.js).

*AWS Lambda:  
Trivia Storage Service*



*AWS S3:  
Stored trivia  
questions*

*JSON:API*

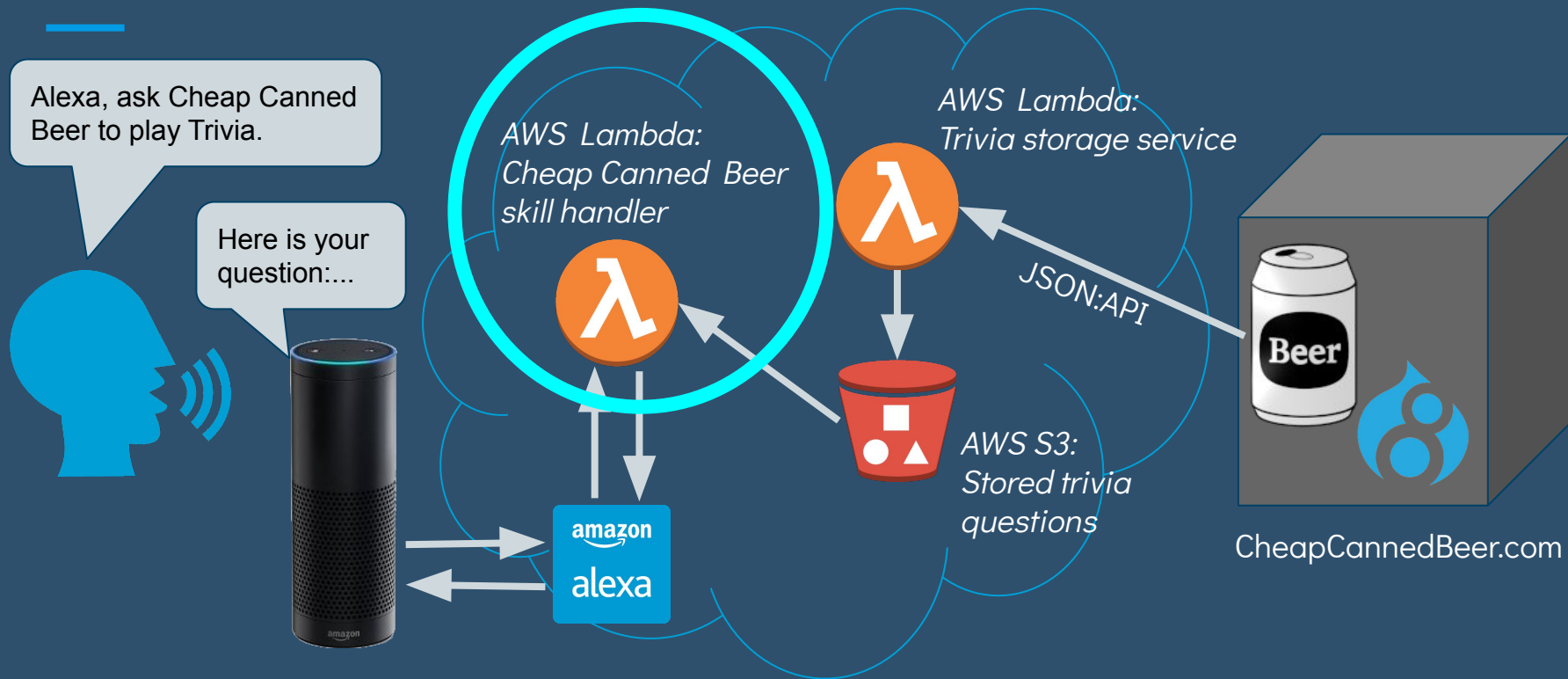


*CheapCannedBeer.com*

# After Trivia Storage Service parses/stores data:

```
{
  "title": "Good for young mothers?",
  "question": "Which beer brand once placed newspaper advertisements aimed at young mothers, claiming, \"the malt  
the beer supplies nourishing qualities,\" \"the hops act as an appetizing, stimulating tonic,\" and, \"obviously  
baby participates in the benefits?\"/r/n/r/n",
  "correctAnswerDescription": "An online image search for \"Blatz\" will turn up multiple clippings of this advert  
sement aimed at nursing mothers./r/n/r/n",
  "choices": [
    {
      "choiceDescription": "Pabst Blue Ribbon",
      "isCorrect": false
    },
    {
      "choiceDescription": "Blatz",
      "isCorrect": true
    },
    {
      "choiceDescription": "Olympia",
      "isCorrect": false
    },
    {
      "choiceDescription": "Old Milwaukee",
      "isCorrect": false
    }
  ],
  "answerImageUrl": "https://www.cheapcannedbeer.com/sites/default/files/2017-11/BlatzMama.jpg"
},
```

# Next: Skill Handler uses S3 for trivia questions



# Alexa Skill Handler

---

- Reads JSON object from S3.
- Uses Alexa Skills Kit to define handlers for Cheap Canned Beer intents and slots.
- Also written in Javascript (Node.js).

*AWS Lambda:  
Cheap Canned Beer  
skill handler*



*AWS S3:  
Stored trivia  
questions*



# Terminology for Alexa Skills

---

- **Invocation name**

The name users call out to activate your skill.

- **Interaction model**

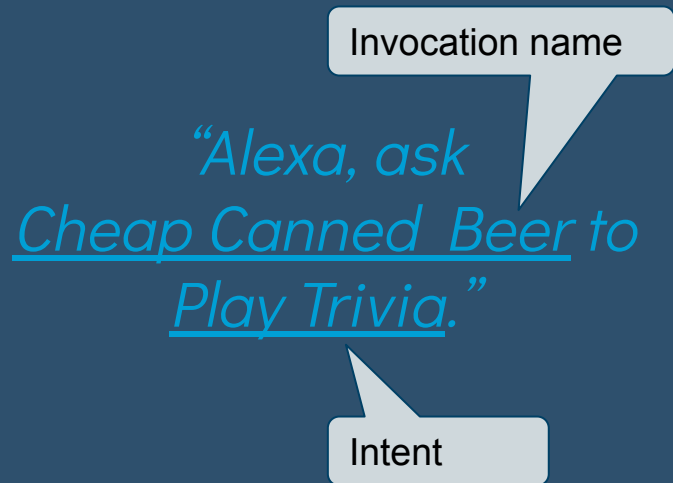
The words and phrases users can say to use your skill. These are modeled as *Intents*.

- **Intent**

Something a user wants to do with your skill.

- **Slot**

A variable that gets filled in the course of an interaction, e.g. the answer to a trivia question.



# The PlayTrivia Dialog Model

---

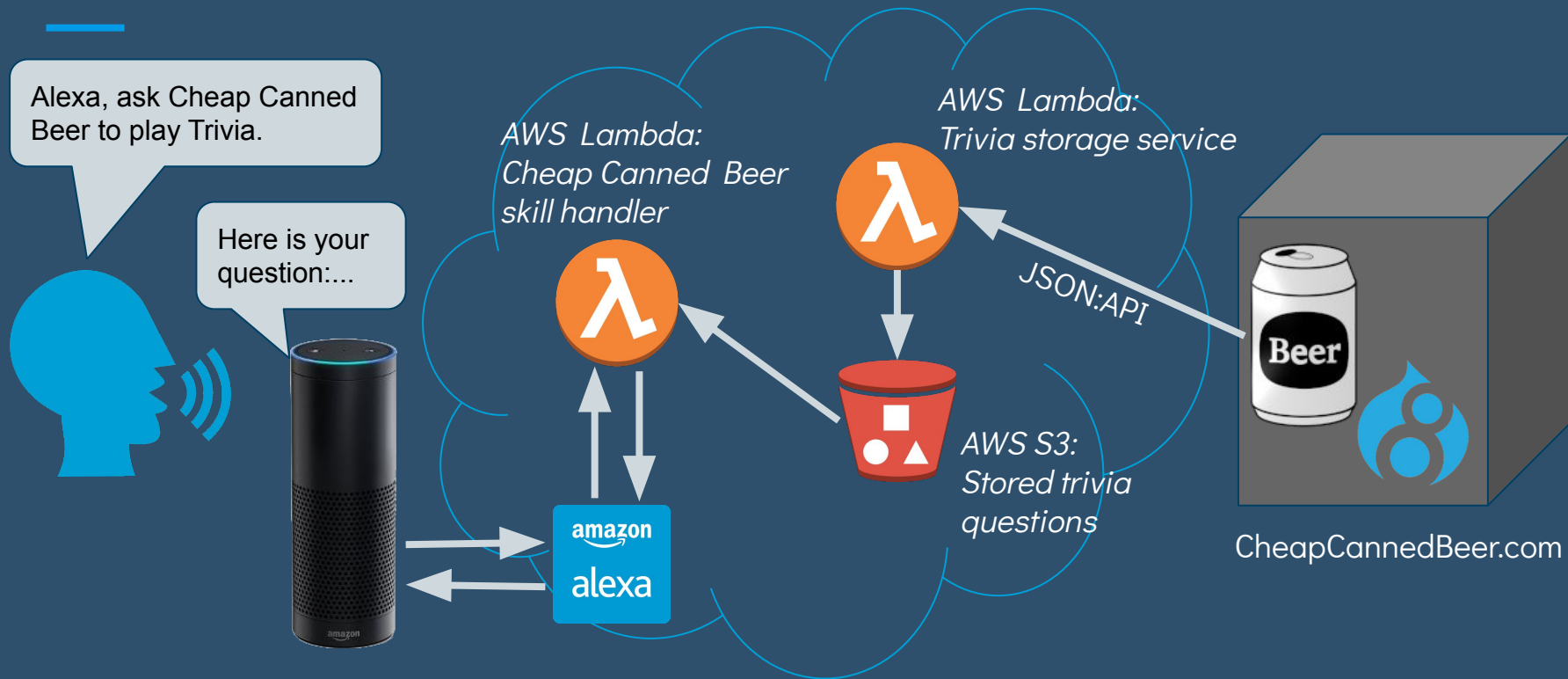


# Cheap Canned Beer on Alexa!

---



# The full picture, one more time...



# Take-aways

---

- Drupal is well-positioned: a key open source content manager for applications of all kinds.
- JSON:API in core makes it possible to write applications that interface with Drupal content.
- Good content structure is important! Don't put everything in the body field.

# Questions?

# Thank you!

---

Contact me:

Paul McKibben

Email: [paul@turbojettech.com](mailto:paul@turbojettech.com)

Twitter: [@paulmckibben](https://twitter.com/paulmckibben)



***TURBOJET***

[www.turbojettech.com](http://www.turbojettech.com)