

Introducing OpenWhisk

Rob Allen

November 2018

So, about this Serverless
thing...

Serverless

- Code
- Deployed to the cloud
- Executed in response to an event
- On-demand scaling
- Pay for execution, not when idle

Serverless is a great fit for

- APIs & Microservices
- One off end points
- Web hooks
- Timed events
- Data transformation
- IoT
- Cognitive

For me, it's about value



Beau @BeauVrolyk · 30m

Replies to @akrabat @kelseyhightower

1) "Serverless" is a point on the path to true app isolation. Apps want to just run, their authors don't care about infrastructure at all.



1



2



Beau @BeauVrolyk · 29m

Replies to @akrabat @kelseyhightower

2) The App author should not need to know, anymore than a Journalist knows about printing presses or what the voltage of the power used.



1



2



Beau @BeauVrolyk · 25m

Replies to @akrabat @kelseyhightower

3) We are relearning what was known in the time-share days. Pricing needs to be based on something customers value, not infra. items like VMs

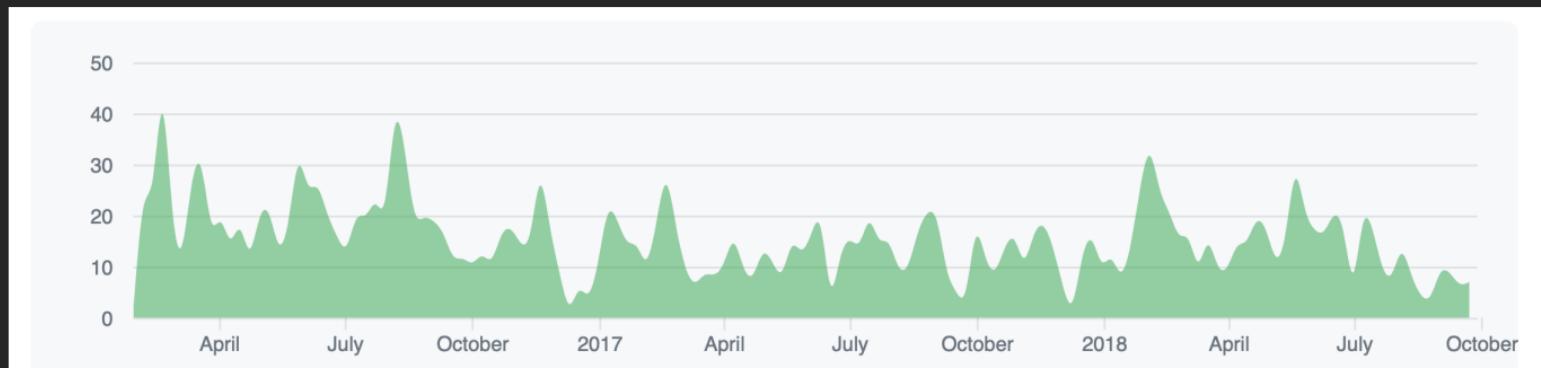




Apache OpenWhisk

Apache OpenWhisk

- <http://openwhisk.org>
- Incubator project working towards graduation
- Many contributors



Apache OpenWhisk

- OpenSource
- Self-hosted
- Multiple public providers:
IBM
RedHat
Adobe (for Adobe Cloud Platform APIs)

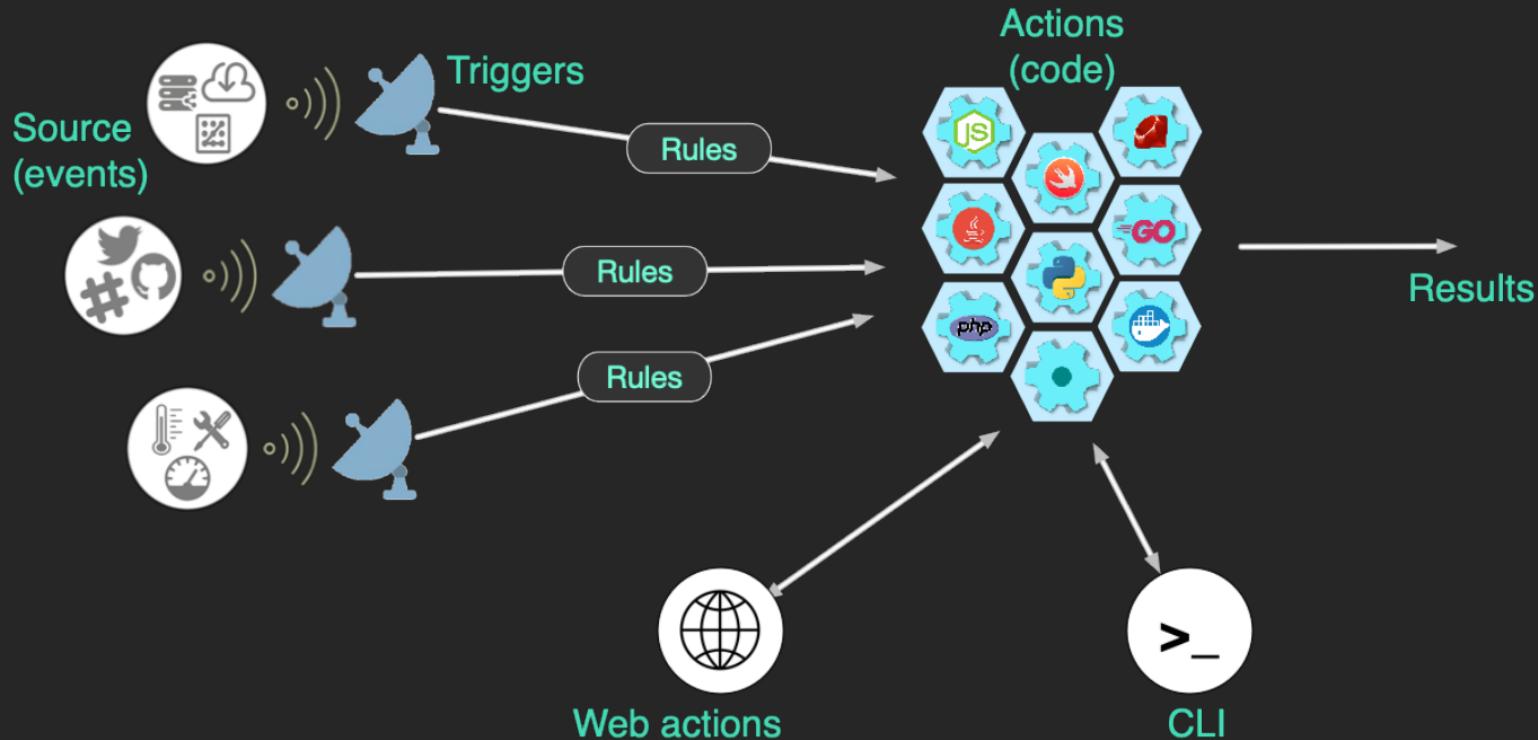
Supported languages

- .NET Core
- Go
- NodeJS
- Python
- Swift
- Ballerina
- Java
- PHP
- Ruby

Also, Your own Docker container can be deployed & invoked

Getting started with OpenWhisk

Concepts



Event providers (feeds)

- GitHub
- IBM App Connect
- IBM Cloudant (CouchDB)
- Message Hub (Kafka)
- Mobile Push
- Periodic

Hello World

Entry point



```
def main(args):
    name = args.get("name", "World")
    message = 'Hello {}!'.format(name)
    return {'body': message}
```

Event parameters

Service result

Deploying your action

```
$ zip -q hello.zip hello.py
$ wsk action update --kind python:3.7 hello hello.zip
ok: updated action hello
```

Deploying your action

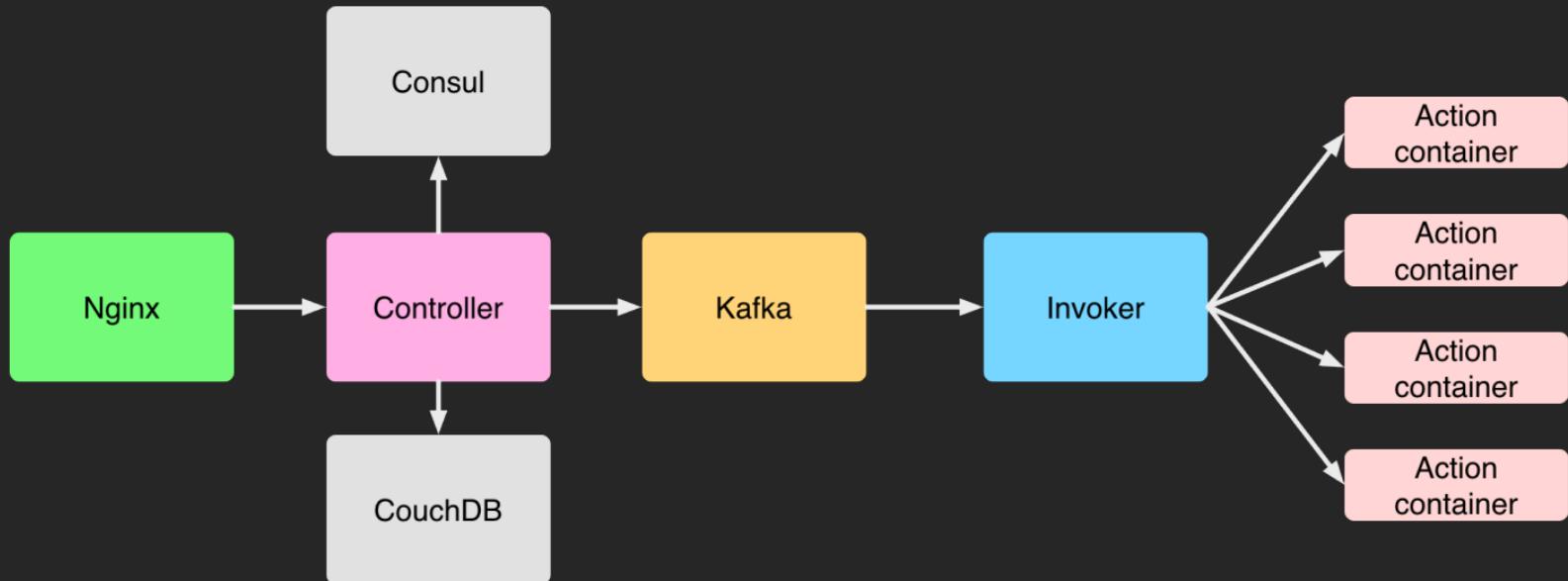
```
$ zip -q hello.zip hello.py
$ wsk action update --kind python:3.7 hello hello.zip
ok: updated action hello
```

Then run it:

```
$ wsk action invoke hello --result --param name Rob
{
    "body": "Hello Rob!"}
```

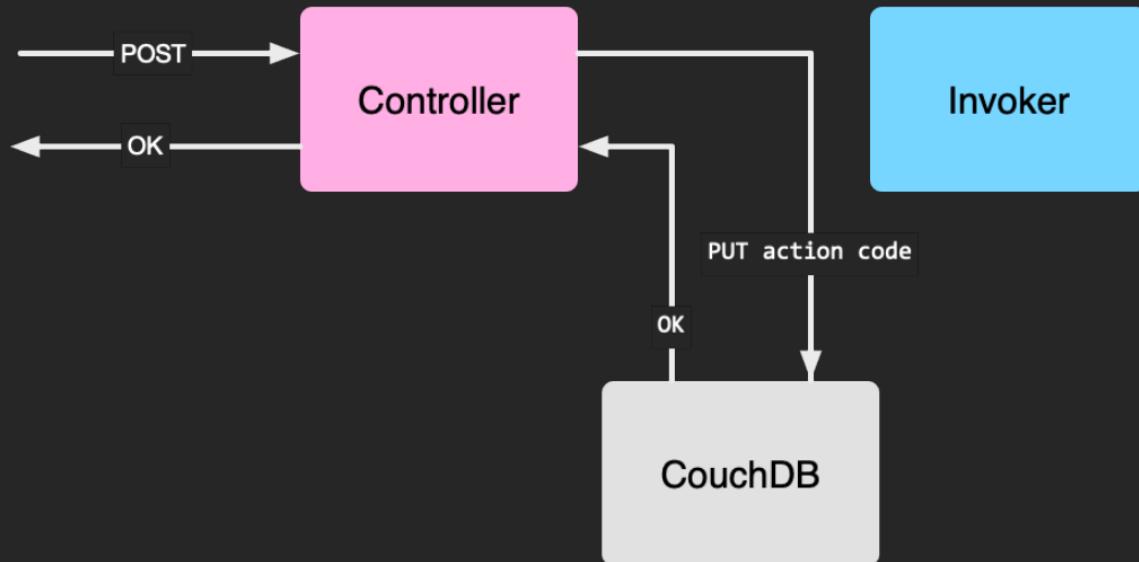
Seque: How did it do this?!

OpenWhisk's architecture



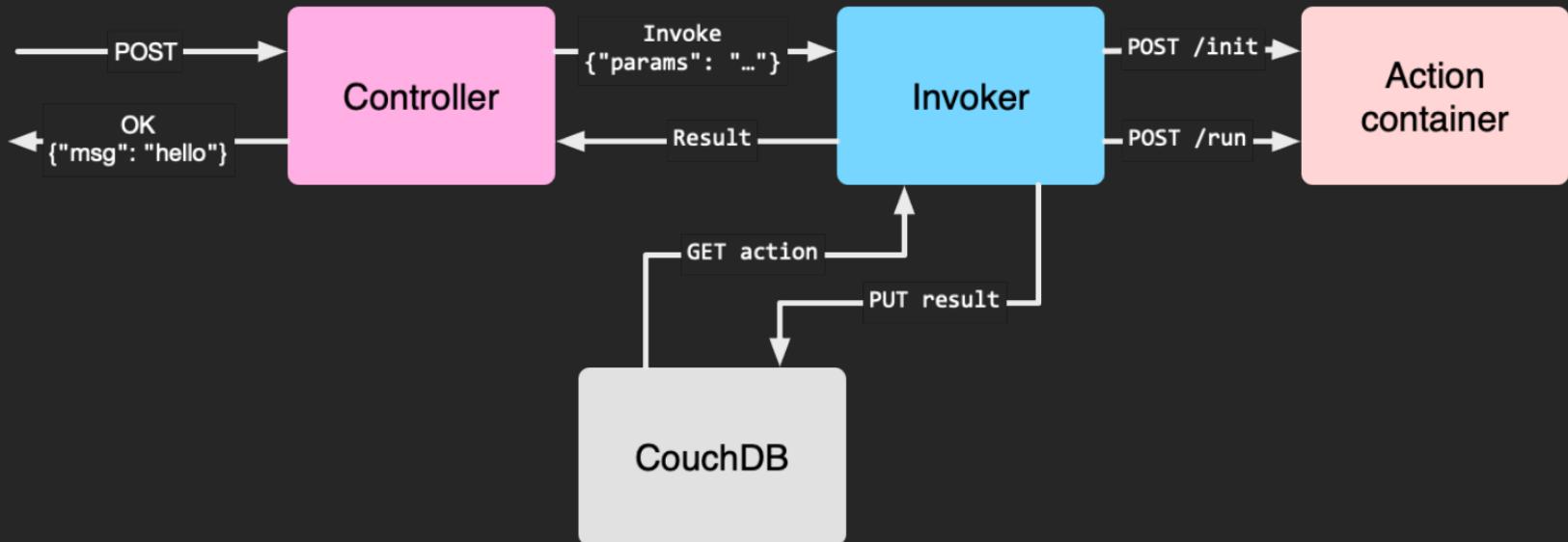
Create an action

```
$ wsk action create hello hello.py
```



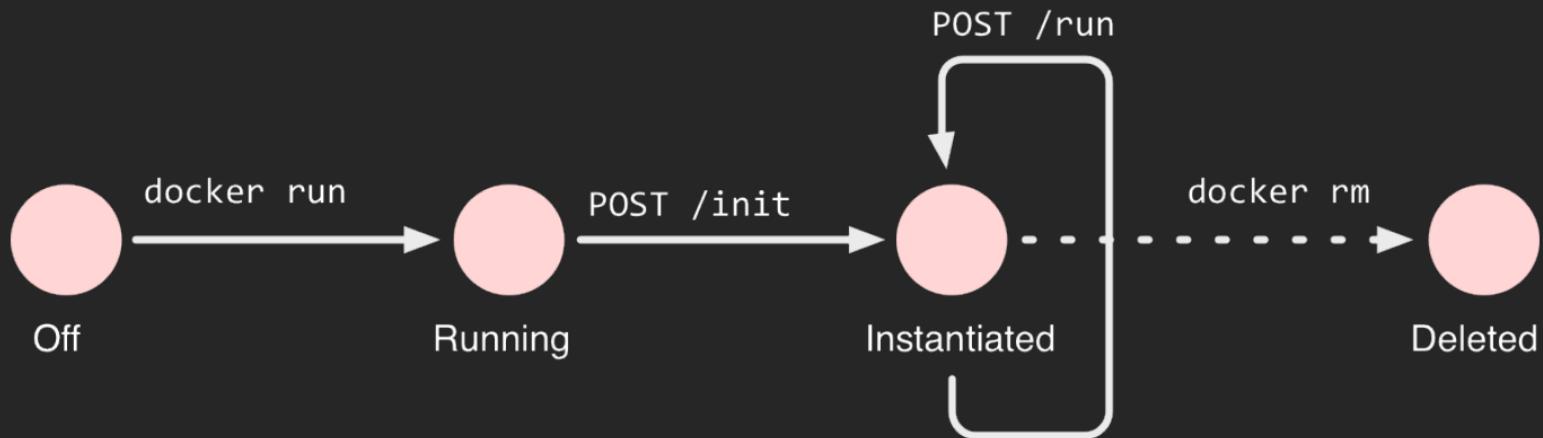
Invoke an action

```
$ wsk action invoke hello -r
```



Action container lifecycle

- Hosts the user-written code
- Controlled via two end points: /init & /run



End Sequ

Web-enabled actions

Access your action via HTTP: Add the --web flag:

```
$ wsk action update --kind python:3.7 --web true \  
demo/hello hello.zip
```

Web-enabled actions

Access your action via HTTP: Add the --web flag:

```
$ wsk action update --kind python:3.7 --web true \  
demo/hello hello.zip
```

Get URL and curl it:

```
$ wsk action get --url demo/hello  
$ curl https://192.168.1.17/api/v1/web/guest/demo/hello
```

API Gateway

When you want to do more with HTTP endpoints

- Route endpoint methods to actions (Open API Spec support)
- Custom domains
- Rate limiting
- Security (API keys, OAuth, CORS)
- Analytics

API Gateway



```
$ wsk api create /myapp /hello GET hello  
ok: created API /myapp/hello GET for action /_/hello
```

API Gateway



```
$ wsk api create /myapp /hello GET hello  
ok: created API /myapp/hello GET for action /_/hello
```

```
$ curl https://example.com/myapp/hello?name=Rob  
{  
  "message": "Hello Rob!"  
}
```

Packages

- Group actions together
- Set parameters used by all actions

```
$ wsk package update demo  
$ wsk action update --kind python:3.7 demo/hello hello.zip
```

Built-in packages

- alarms
- combinator
- jira
- pushnotifications
- samples
- utils
- weather
- cloudant
- github
- kafka
- rss
- slack
- watson
- websocket

List using: `wsk package list /whisk.system`

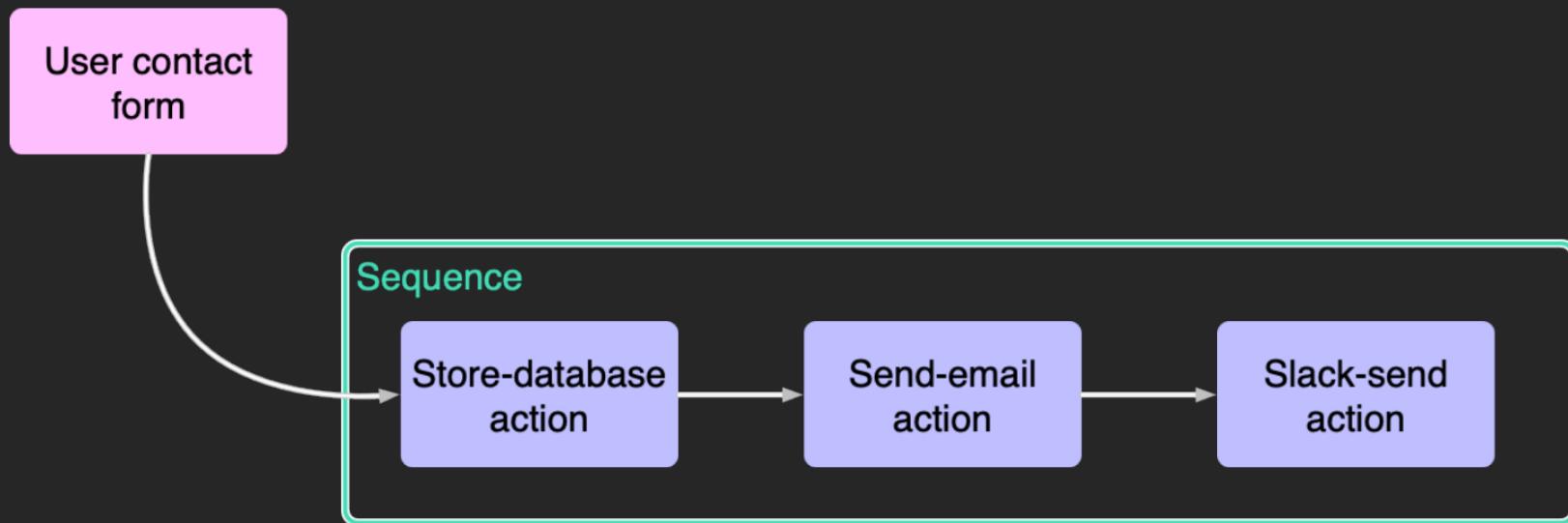
Using built-in actions

Count words in a string:

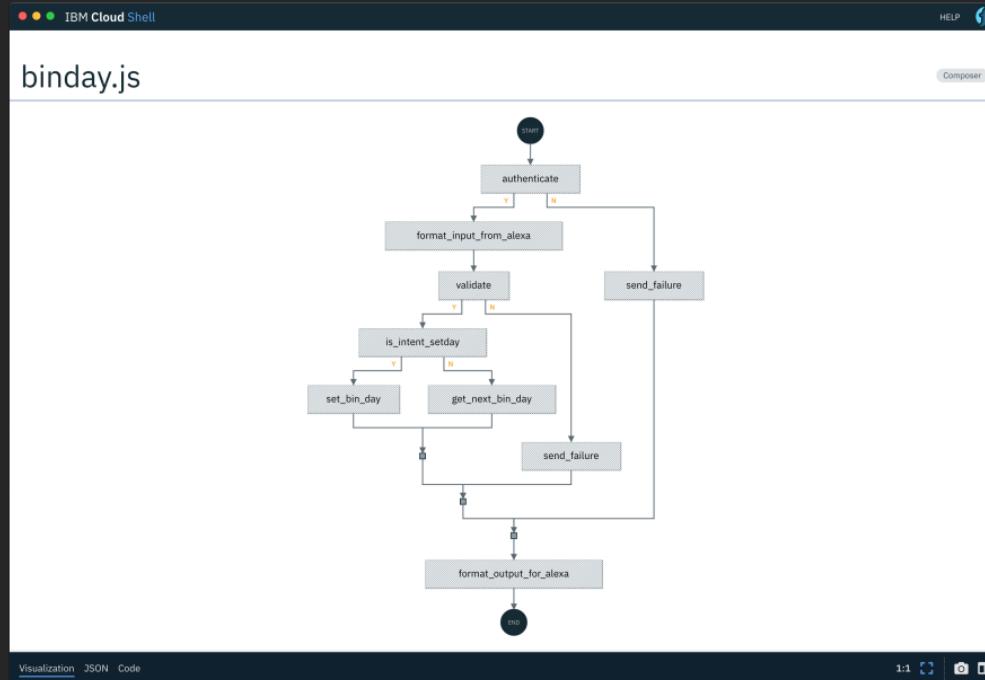
```
$ wsk action invoke -r /whisk.system/samples/wordCount \
-p payload "Twas brillig, and the slithy toves"
{
    "count": 6
}
```

Sequences

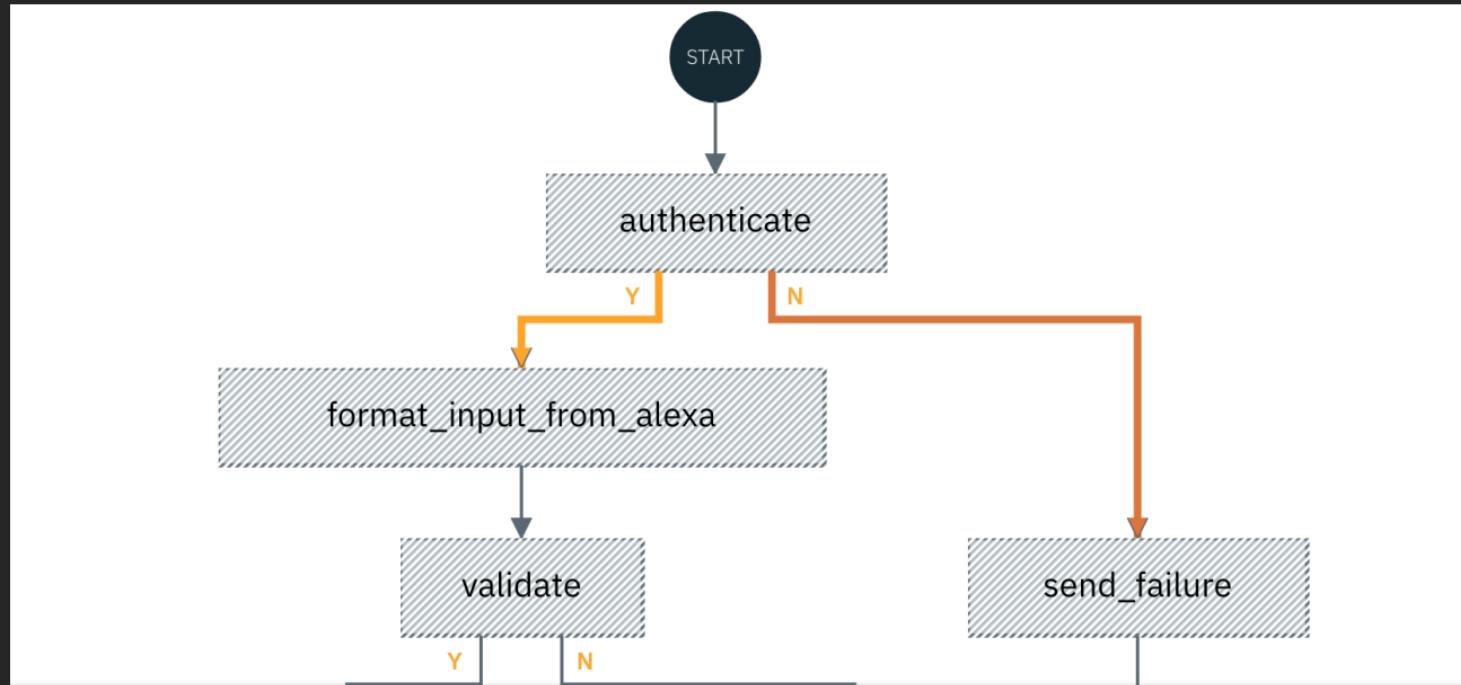
Invoke a set of actions in turn



Composer: Logic for your actions



Composer: Logic for your actions



OpenWhisk Composition

```
composer.sequence(  
    composer.if(  
        'binday/authenticate',  
        composer.sequence(  
            'format_input_from_alexa',  
            composer.if(  
                'binday/validate',  
                'binday/get_next_bin_day'  
                'binday/send_failure'  
            ),  
        ),  
        'binday/send_failure'  
    ),  
    'binday/format_output_for_alexa'  
)
```

Serverless Framework

Plugin available for the Serverless Framework

Set up:

```
$ npm install --global serverless serverless-openwhisk  
$ serverless create --template openwhisk-php --path app  
$ cd app  
$ npm install
```

serverless.yml

```
service: ow-todo-backend
```

```
provider:
```

```
  name: openwhisk  
  runtime: php
```

```
plugins:
```

```
  - serverless-openwhisk
```

serverless.yml

```
functions:  
  list-todos:  
    handler: "src/actions/listTodos.main"  
    name: "todo-backend/list-todos"  
    events:  
      - http:  
          path: /todos  
          method: get  
  add-todo:  
    handler: src/actions/addTodo.main  
    name: "todo-backend/add-todo"  
  ...
```

Deploy

```
$ serverless deploy
Serverless: Packaging service...
Serverless: Compiling Functions...
Serverless: Compiling Packages...
Serverless: Compiling API Gateway definitions...
Serverless: Compiling Rules...
Serverless: Compiling Triggers & Feeds...
Serverless: Compiling Service Bindings...
Serverless: Deploying Packages...
Serverless: Deploying Functions...
Serverless: Deploying API Gateway definitions...
[...]
```

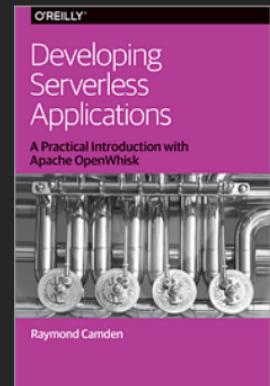
To sum up

Resources

- <http://www.openwhisk.org>
- <https://github.com/apache/incubator-openwhisk-workshop>
- <https://serverless.com/framework/docs/providers/openwhisk>

Developing Serverless Applications
by Raymond Camden

Free at <https://akrab.at/openwhiskbook>



Thank you!