Wellcome AWS Utils Documentation

Release 1.0.0

Wellcome Digital Platform

Jun 25, 2020

Contents

This package is a collection of utilities written at Wellcome for interacting with AWS.

Some of these utilities are very specific to Wellcome projects, others are more generic and should be generally useful.

The best place to start is the *API reference*, which describes all the utilities the package provides. It's a bit sparse at the moment, but hopefully we'll expand it soon!

CHAPTER 1

API reference

CHAPTER 2

Changelog

This is a record of all releases of wellcome_aws_utils.

2.1 2.3.3 - 2019-07-26

Makes sure that the Elasticsearch doc is sent over as a string.

2.2 2.3.2 - 2019-05-21

Adds ability to switch AWS roles when fetching elasticsearch credentials

2.3 2.3.1 - 2019-05-21

Now with fixed Travis credentials.

2.4 2.3.0 - 2019-05-21

This release modifies the way that secrets are handled by lambdas in the reporting pipleine. Previously, secrets were passed to lambdas as environment variables, defined in terraform. We now fetch secrets from AWS secretsmanager as records move through the pipeline.

2.5 2.2.1 - 2018-11-23

A large number of records in the Sierra VHS contain a `reindexShard` parameter which is not expected when initialising a `HybridRecord()` object. `attrs` can't handle data it doesn't expect, and the records with `reindexShard`

parameters therefore fail to pass through the pipeline.

We now throw away any unnecessary data in the received message, allowing originally dirty messages to pass through without issue.

2.6 2.2.0 - 2018-11-08

This release adds utils for the reporting pipeline.

The functions under `reporting_utils.py` describe a basic ETL pipeline from VHS to Elasticsearch, without a transformation specified. In this way, the shape of the pipeline remains independent of both the data within it and the transforms being applied.

As further data sources are added to the reporting pipeline and more Lambda functions are created, we keep repeated code to a minimum. In a new Lambda function, the user should specify a set of data-source-specific transformations in a `transform.py` file. The Lambda's `main` can then remain minimal and generic:

2.7 2.1.3 - 2018-08-17

This fixes a bug in the `@log_on_error` decorator where the return value of the original function would be replaced by `None`. This decorator now preserves the original return value.

2.8 2.1.2 - 2018-06-26

Previously sending a message with `sns_utils.publish_sns_message` would print a message upon success.

Now this message is only logged at debug level.

2.9 2.1.1 - 2018-06-04

Now `@log_on_error` can be used to decorate functions with arbitrary arguments/keyword arguments.

2.10 2.1.0 - 2018-06-04

This adds a new method: `lambda_utils.log_on_error`. This can be used to decorate the main function for a Lambda, and logs the event/context if the Lambda throws an unexpected exception.

For example, running the following snippet:

```
@log_on_error
def handler(event, context=None):
    if event == {1: '1', 2: '2'}:
        raise ValueError


handler(event={'foo': 'bar'})
handler(event='99 green bottles' * 99)
handler(event={1: '1', 2: '2'})
```

gives the following output:

This makes it easier to debug failed Lambdas, but without the expense of logging every event that a Lambda receives.

2.11 2.0.2 - 2018-06-04

Previously sending a message with `sns_utils.publish_sns_message` would log the entire SNS response.

Now the response is only logged if the SNS message is unsuccessful.

2.12 2.0.1 - 2018-01-12

This fixes a bug in `s3_utils.parse_s3_record`. If the key of a changed file included a character which is usually quoted in URLs (e.g. `+`), a parsed record from the S3 event stream would use the URL-quoted form of the object key.

For example, a change to `s3://example/foo+bar` would become `foo%2Bbar`.

This version unquotes the key when parsing the event.

2.13 2.0.0 - 2017-11-29

Replacing the DynamoImageFactory and DynamoImage classes with DynamoEventFactory and DynamoEvent

- Perform quite a bit of sanity checking on event object received
- DynamoEvent can: - return old and new images (if available) - return modified keys only - return deserialized or otherwise images and keys based on params

2.14 1.1.0 - 2017-11-15

Deprecates `sns_utils.extract_json_message` in favour of `sns_utils.extract_sns_messages_from_lambda_event`.

extract_sns_messages_from_lambda_event provides: - better error reporting if the event is malformed - loops over all available records from event not just the first - returns subject along with the json decoded message

This release also adds `UnWellcomeException` which will be used as the base exception for new errors.

2.15 1.0.0 - 2017-11-07

First production release!