



Optimize SEO with Serverless Sitemaps

Meeky Hwang

Ndevr

/ Since 2015

/ Distributed

/ Technology consulting

/ <https://ndevr.io>

/ <https://www.linkedin.com/company/ndevr>



Clients & Partners

Bloomberg

DIGITAL 360
COMMERCE

MTS

Health
Partners

PMIC

ESSENCE



**FOLEY
HOAG** LLP

cerberus

What Works Cities

NATURALLY *Curly*.COM®

**AFRO
PUNK**

tapestry



303MAGAZINE.COM



Me

/ CTO, Partner and Technical Architect - Ndevr

/ B.S. Systems and Information Engineering (UVA)

/ MFA 3D Animation (SVA)

/ 19+ Years Building Websites

/ WordPress core/plugins contributor

/ WordPress.com VIP

/ <https://www.linkedin.com/in/meekeyhwang/>

In this talk

/ WordPress sitemaps

/ Limitations and issues

/ Alternatives

/ AWS Serverless

/ Example

/ Next steps

Not in this talk

/ AWS Lambda / DyanmoDB / CloudWatch / API Gateway

/ WordPress development

/ NodeJS development

/ Sitemaps specifications (Ask Google)

WordPress Sitemaps

/ Popular plugins

// All in One SEO Pack

// Google XML Sitemaps

// Simple Sitemaps

// WordPress SEO by Yoast

// Better WordPress Google XML Sitemaps

Limitations & Issues

/ Query the contents upon sitemap request

/ Customize site structure not fully supported

// Custom content types

// Media embeds

// Custom embeds

// Remove default settings

/ Old and obsolete URL cleanup

Limitations & Issues (cont'd)

/ Video Sitemap

/ Caching

/ News Sitemap

Alternatives

- / Customized the plugin(s)
- / Set up the custom CLI command to generate sitemaps offline
- / Scheduled jobs to generate / update sitemaps
- / Generate xml files in smaller chunks

Issues persist...

Sitemaps: The Needs

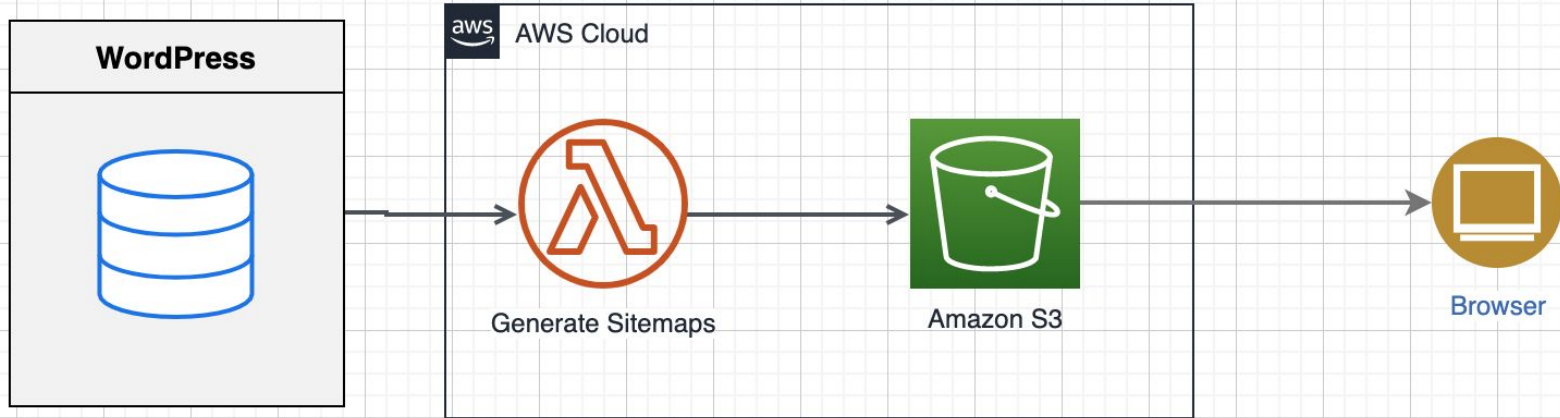
- / Basic XML sitemaps
- / Better promote rich media contents with image and video sitemaps
- / Reduce amount, increase size of each sitemap file
- / Sitemap update upon content updates
- / Sitemaps available without timing out
- / Submit to search engines upon sitemaps generation

Serverless

Serverless is the native architecture of the cloud that enables you to shift more of your operational responsibilities to AWS, increasing your agility and innovation. Serverless allows you to build and run applications and services without thinking about servers. It eliminates infrastructure management tasks such as server or cluster provisioning, patching, operating system maintenance, and capacity provisioning. You can build them for nearly any type of application or backend service, and everything required to run and scale your application with high availability is handled for you.

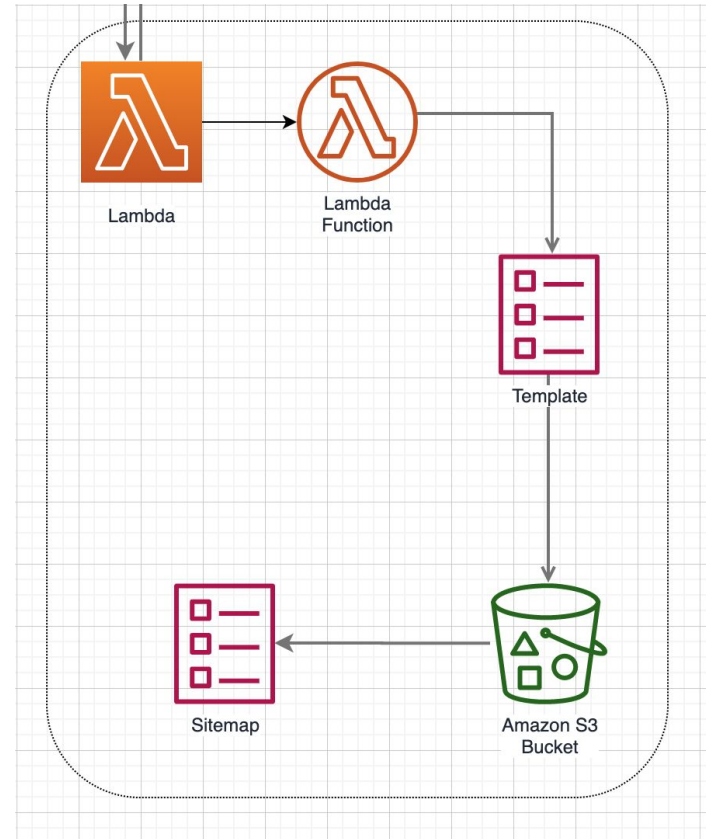
<https://aws.amazon.com/serverless/>

Version 1.0



Generate Sitemaps

Create a lambda function to generate sitemaps



```
let chunks = _1.chunk(result.Items, 10000);
async function(index){
  let chunk = chunks[index];
  let urls = [];
  _1.each(chunk, async function(item){
    let d = new Date(item.updatedAt * 1000);
    urls.push({
      url: item.url,
      img: item.images,
      video: item.videos,
      changefreq: 'daily',
      priority: 0.5,
      lastmod: d
    });
  });
  let sitemap = sm.createSitemap({
    hostname: 'essence.com',
    cacheTime: 600000,
    urls: urls
  });
  await upload(sitemap.toString(), sitemapGeneratedPrefix + (index + 1) + '.xml.gz');
}
```

packages.json

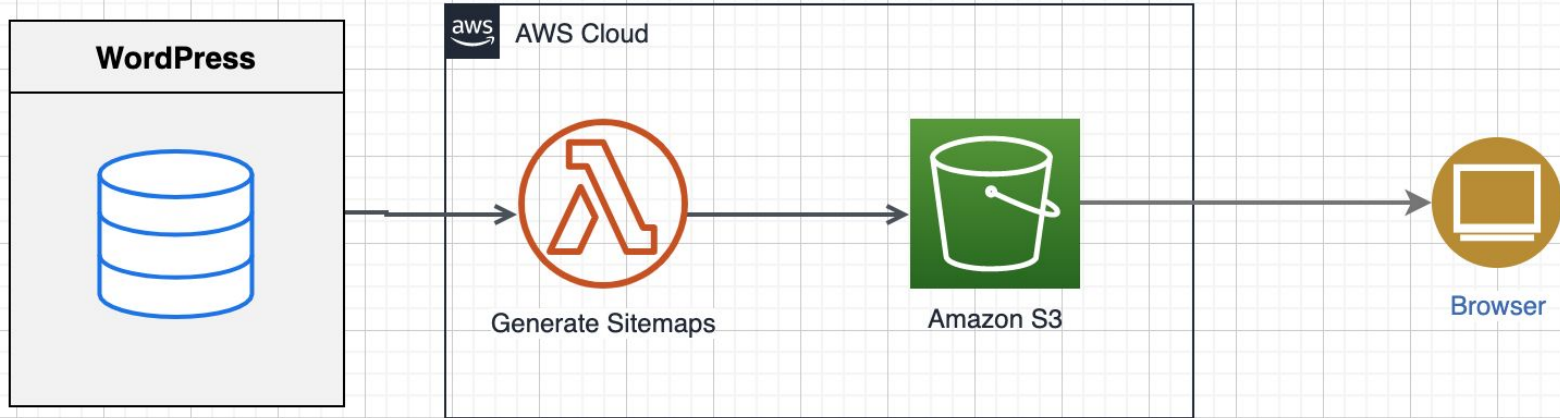
```
"dependencies": {
  "aws-sdk": "^2.395.0",
  "lodash": "^4.17.13",
  "restler": "^3.4.0",
  "s3-upload-stream": "^1.0.7",
  "sitemap": "^2.1.0",
  "string-to-stream": "^1.1.1",
  "underscore": "^1.9.1",
  "when": "^3.7.8",
  "xmlbuilder": "^10.1.1"
}
```

Source: frankadrian/lambda-sitemap-generator


```
const upload = async function (content, filename) {
  return await when.promise(function(resolve, reject, notify){
    s3Stream = require('s3-upload-stream')(new AWS.S3()); // Upload sitemap to s3
    let read = str(content); // Create the stream
    let compress = zlib.createGzip();
    let upload = s3Stream.upload({
      'Bucket' : process.env.BUCKET,
      'Key': filename
    });

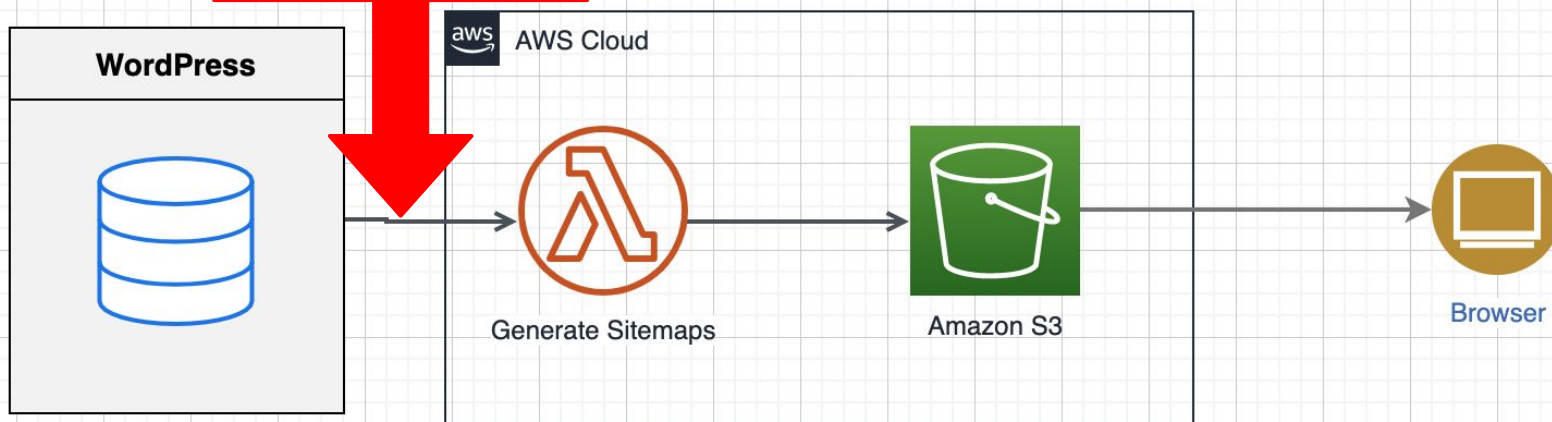
    upload.maxPartSize(20971520); // 20 MB
    upload.concurrentParts(5);
    upload.on('error', function (error) {
      console.log('error: ', error);
      reject(error);
    });
    upload.on('uploaded', function (details) {
      console.log('uploaded: ', details);
      resolve();
    });
    // Pipe the incoming stream through compression, and up to S3.
    read.pipe(compress).pipe(upload);
  });
};
```

Version 1.0

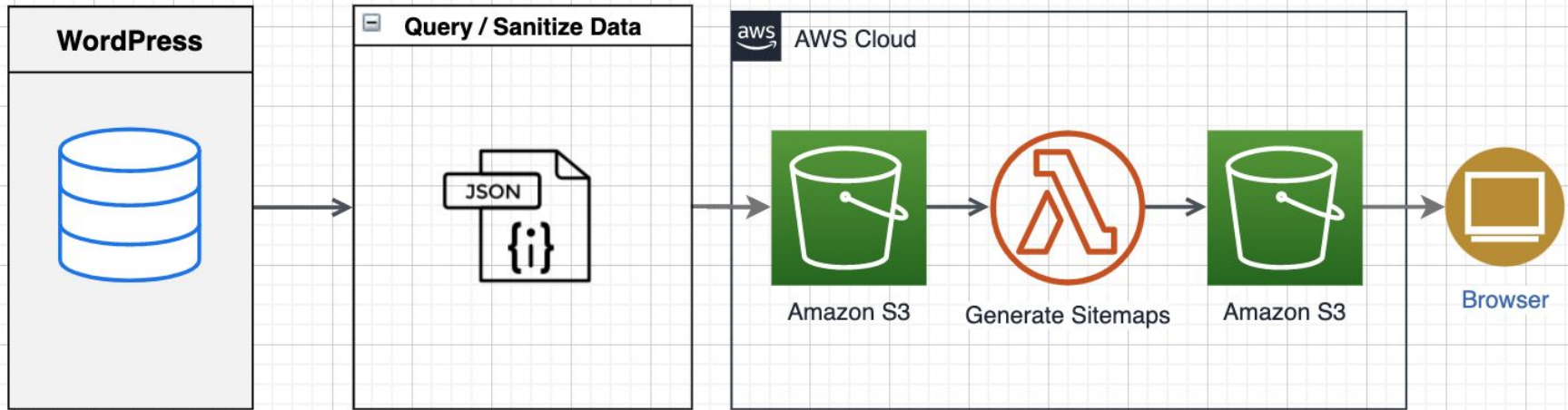


Version 1.0

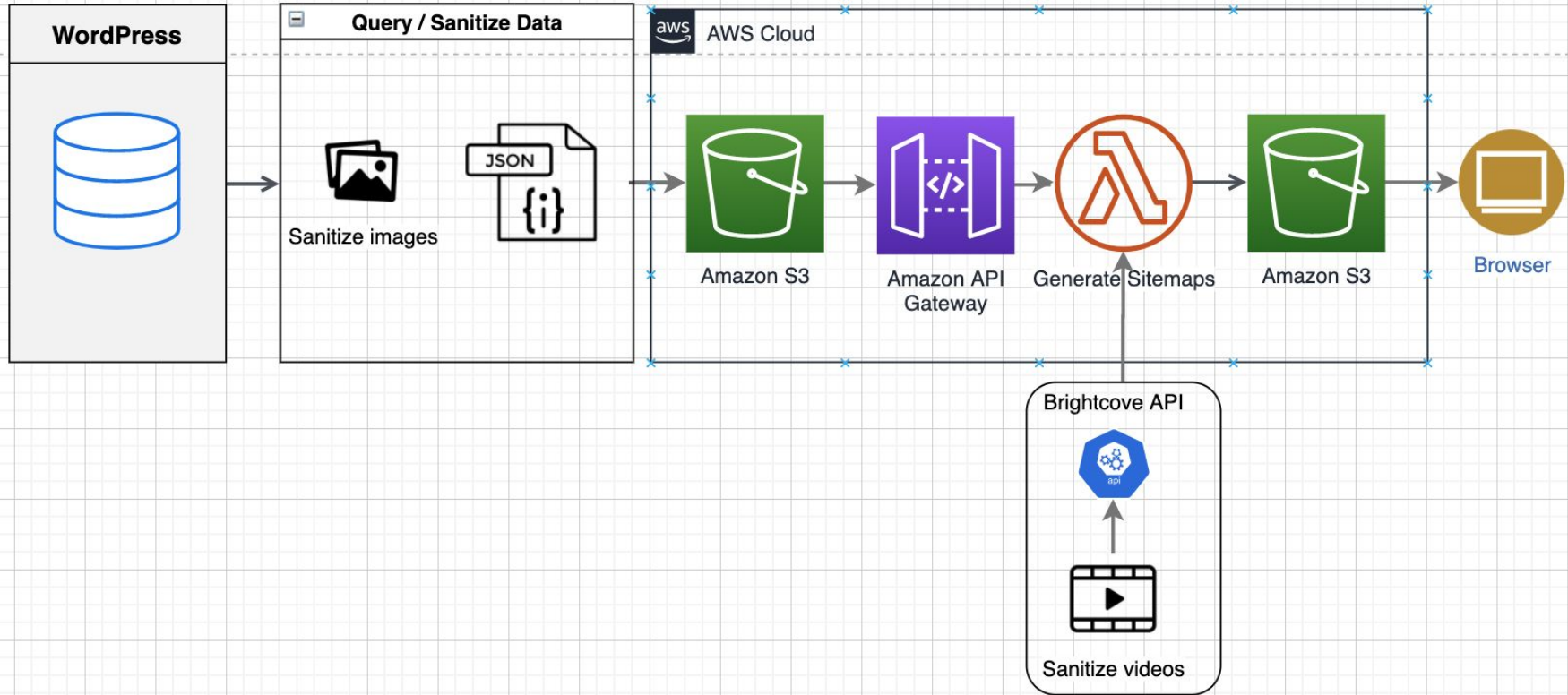
Bottleneck



Version 1.1

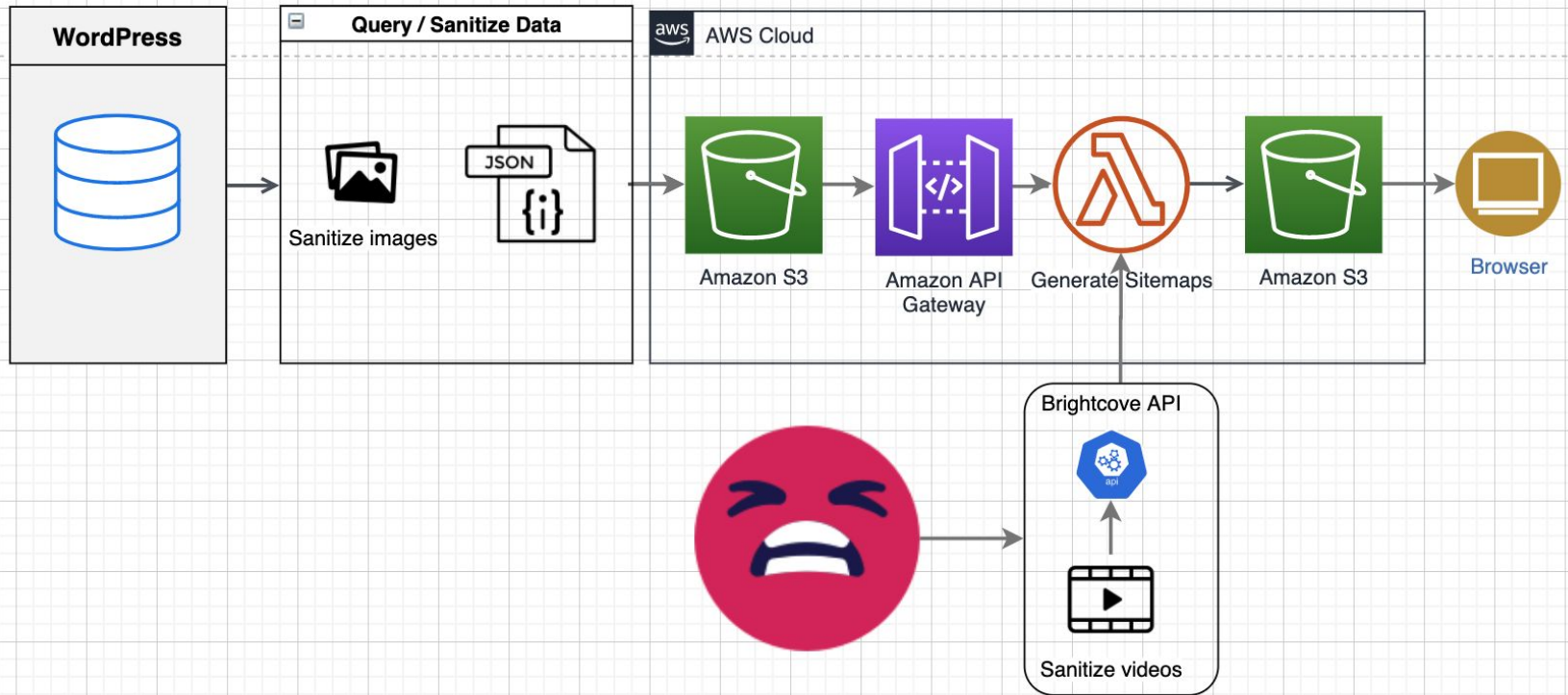


Version 1.2

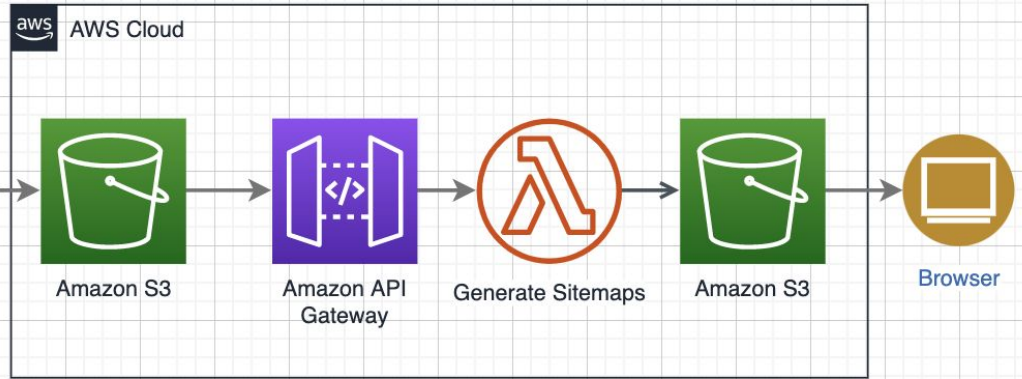
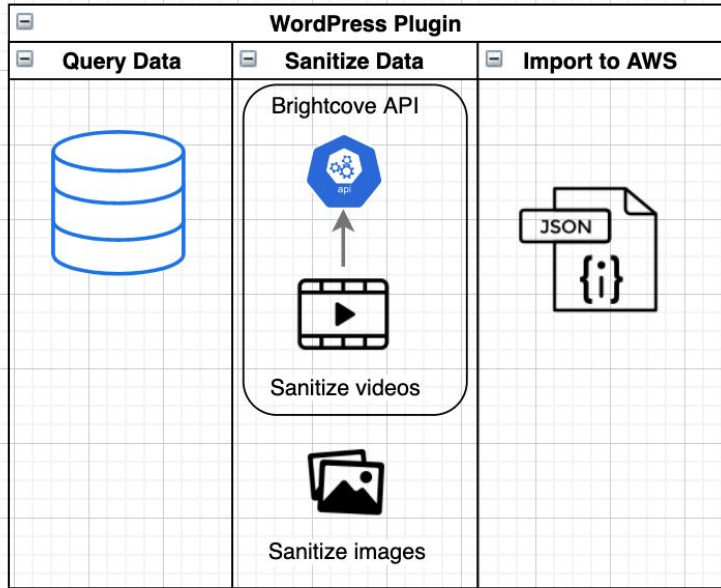


```
{
  "id":"1234",
  "type":"article",
  "url":"https://www.sample.com/videos/nene-leakes-kenya-moores-husband-marc-daly/",
  "images":[
    {
      "url":"https://www.sample.com/wp-content/uploads/2016/07/GettyImages-927369474-1568x2149.jpg"
    },
    {
      "url":"https://www.sample.com/wp-content/uploads/2020/03/NUP_188382_0727.jpg"
    }
  ],
  "videos":[
    {
      "brightcove_id":"6042451094001"
    },
    {
      "brightcove_id":"6077896689001"
    }
  ],
  "created":819951320000,
  "modified":819951320000
}
```

Version 1.2



Version 1.3




```
{
  id: '527134',
  type: 'article',
  url: 'https://www.sample.com/news/ppp-relief-aid-package-fails-black-businesses/',
  created: 1587553093,
  modified: 1587553097,
  images: [
    {
      caption: 'Queens, NY business that should be able to benefit from the PPP',
      location: 'https://www.sample.com/wp-content/uploads/2020/04/GettyImages-1216652629-scaled.jpg'
    },
    {
      caption: 'Trump and Mnuchin discuss PPP',
      location: 'https://www.sample.com/wp-content/uploads/2020/04/GettyImages-1217449615-scaled.jpg'
    }
  ],
  videos: [
    {
      thumbnail_loc:
'https://cf-images.us-east-1.prod.boltdns.net/v1/static/716147494/e6fab89e-e4cf-444b-aad9-62174bf9e040/ebc99476-16e3-4c2f-a26b-4578dbfb6559/1280x720/match/image.jpg',
      content_loc:
'http://house-fastly-signed-us-east-1-prod.brightcovecdn.com/media/v1/pmp4/static/clear/716147494/e6fab89e-e4cf-444b-aad9-62174bf9e040/9efe923a-ae0-45db-98b1-8feccd15a423/main.mp4?fastly_token=NWViYmZlOWNFMTkwYzVlYWQ4NGMxNmQ5NzAyNDk4ODBkMTRkYzlkMGY4ODcyYTBiM2MwMTMyNThjODdlNmZkMjM2MzUyOTNhNV8vL2hvdXNlLWZhc3RseS1zaWduZWQtdXMtZWZzdC0xLXByb2QuYnJpZ2h0Y292ZWNBi5jb20vbWVkaWVkdjEvcG1wNC9zdGF0aWMyY2x1YXVnZmE2MTQ3NDk0L2U2ZmFiODl1LWU0Y2YtNDQ0Yi1hYWQ5LTlYMTc0YmY5ZTA0MCM4SjZlOTIzYS1hZWUwLTQ1ZGI0OTIhMS04ZmVjY2QxNWU0MjMvbmVpbi5tcDQ%3D',
      title: ' Q&A on the #CARESact with John Stanford and Ron Busby',
      description: 'Essence chats with John Stanford, Managing Partner at Prism Group & Co-Founder Small Business Roundtable and Ron Busby President & CEO, U.S. Black Chambers, Inc. for a Q&A on the #CARESact and its effect on small businesses.',
      duration: 1409728,
      publication_date: '2020-04-13T19:36:45.369Z'
    }
  ]
}
```

Create sitemap Index

```
// Create sitemap index
let root = builder.create('sitemapindex', {encoding: 'UTF-8'}).att('xmlns',
'http://www.sitemaps.org/schemas/sitemap/0.9');

// add in each sitemap
_.each(chunks, function (chunk, index) {
  let sitemap = root.ele('sitemap');
  sitemap.ele('loc', site_url + '/' + sitemapGeneratedPrefix + (index + 1) + '.xml.gz');
  sitemap.ele('lastmod', new Date().toISOString());
});










let xmlString = root.end({
  pretty: true,
  indent: '  ',
  newline: '\n',
  allowEmpty: false
});

// Upload Master index sitemap
await upload(xmlString, sitemapIndex);
```

[Upload](#) [+ Create folder](#) [Download](#) [Actions](#) ▾

US East (N. Virginia) 

< Viewing 1 to 9 >

<input type="checkbox"/>	Name ▾	Last modified ▾	Size ▾	Storage class ▾
<input type="checkbox"/>	 sitemap_1.xml.gz	Apr 28, 2020 1:39:26 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_2.xml.gz	Apr 28, 2020 1:39:30 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_3.xml.gz	Apr 28, 2020 1:39:32 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_4.xml.gz	Apr 28, 2020 1:39:34 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_5.xml.gz	Apr 28, 2020 1:39:35 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_6.xml.gz	Apr 28, 2020 1:39:37 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_7.xml.gz	Apr 28, 2020 1:39:39 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemap_8.xml.gz	Apr 28, 2020 1:39:41 PM GMT-0400	1.2 MB	Standard
<input type="checkbox"/>	 sitemapindex.xml.gz	Apr 28, 2020 1:39:41 PM GMT-0400	219.0 B	Standard

< Viewing 1 to 9 >

```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:news="http://www.google.com/schemas/sitemap-news/0.9"
xmlns:xhtml="http://www.w3.org/1999/xhtml" xmlns:mobile="http://www.google.com/schemas/sitemap-mobile/1.0"
xmlns:image="http://www.google.com/schemas/sitemap-image/1.1" xmlns:video="http://www.google.com/schemas/sitemap-video/1.1">
  <url>
    <loc>https://www.essence.com/news/project-runways-korto-momolu-runway-show/</loc>
    <lastmod>2008-10-23</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.5</priority>
  </url>
  <url>
    <loc>https://www.essence.com/videos/overexplainer-reacts-julia-roberts-harriet-tubman/</loc>
    <lastmod>2019-11-23</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.5</priority>
  <video:video>
    <video:thumbnail_loc>https://cf-images.us-east-1.prod.boltdns.net/v1/static/716147494/88585400-b4ca-4bcb-8e6e-26e14911f528/7080a980-a317-
4135-9f37-6f32333819e9/1280x720/match/image.jpg</video:thumbnail_loc>
    <video:title>
      <![CDATA[ The OverExplainer Reacts: Julia Roberts As Harriet Tubman ]]>
    </video:title>
    <video:description>
      <![CDATA[ It's been revealed that a Hollywood exec once suggested casting Julia Roberts as Harriet Tubman. This was not a joke. Watch The
OverExplainer react to the news. ]]>
    </video:description>
    <video:content_loc>http://house-fastly-signed-us-east-1-prod.brightcovecdn.com/media/v1/pmp4/static/clear/716147494/88585400-b4ca-4bcb-
8e6e-26e14911f528/3cd1cfc0-f8da-4e3e-aa65-a55c5d960b80/main.mp4?
fastly_token=NWVnNTk0MjFfZjB1NjhiNWVmMGQ2ZDY2YTNmOTg4MzE3YmVlOTF1YmVhNTIyYmJhZGUwMzYzYzQxNzcxOWUwNTBhZDQwODBiOV8vL2hvdXNlLWZh3RseS1zaWduZW
<video:duration>28799</video:duration>
    <video:publication_date>2019-11-22T20:24:43.023Z</video:publication_date>
  </video:video>
</url>
  <url>
    <loc>https://www.essence.com/love/men-who-cook-alexis-aquino-2/</loc>
    <lastmod>2009-03-27</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.5</priority>
  </url>
  <url>
    <loc>https://www.essence.com/news/donald-trump-new-york-ventilators-andrew-cuomo/</loc>
    <lastmod>2020-03-31</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.5</priority>
  </url>
  <url>
    <loc>https://www.essence.com/lifestyle/editors-pick-phone-cleaner/</loc>
    <lastmod>2020-02-14</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.5</priority>
  <image:image>
    <image:loc>https://www.essence.com/wp-content/uploads/2020/02/Screen-Shot-2020-02-13-at-2.53.08-PM.png</image:loc>
  </image:image>
  <image:image>
    <image:loc>https://www.essence.com/wp-content/uploads/2020/02/5c8a6db6f73f5c31a272c02c.jpeg</image:loc>
  </image:image>
  <image:image>
    <image:loc>https://www.essence.com/wp-content/uploads/2020/02/Screen-Shot-2020-02-13-at-2.52.43-PM.png</image:loc>
  </image:image>
  <image:image>
```

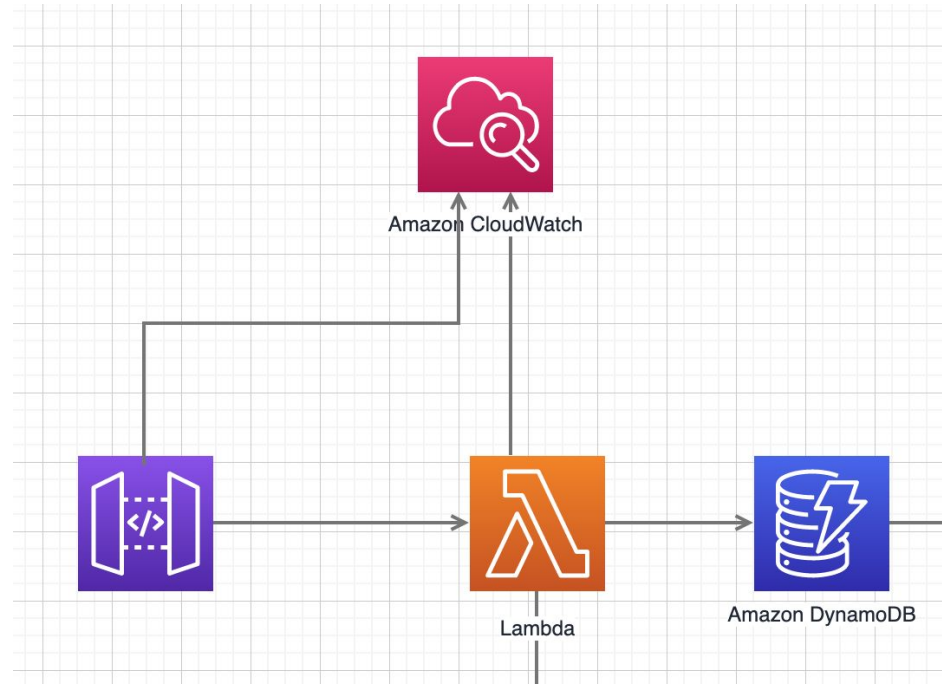
```
▼<sitemapindex xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_1.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_2.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_3.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_4.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_5.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_6.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_7.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
  ▼<sitemap>
    <loc>https://www.essence.com/sitemap_8.xml.gz</loc>
    <lastmod>2020-04-28T17:39:40.615Z</lastmod>
  </sitemap>
</sitemapindex>
```

Sitemaps: The Needs

- / Basic XML sitemaps ✓
- / Better promote rich media contents with image and video sitemaps ✓
- / Reduce amount, increase size of each sitemap file ✓
- / Sitemap update upon content updates
- / Sitemaps available without timing out
- / Submit to search engines upon sitemaps generation

Content updates / Clean up

Import contents to dynamoDB



API Endpoints

Resources

Actions ▾

- ▾ /
 - ▾ /items
 - GET
 - OPTIONS
 - POST
 - PUT
 - ▾ /{id}
 - DELETE
 - GET
 - OPTIONS
 - PUT

/ Create (POST): import one item

/ Import (PUT): bulk import

/ Update (PUT): update one item

/ Delete (DELETE): delete one item

/ Get (GET): get one item (Debug)

/ List (GET): get all items (Debug)

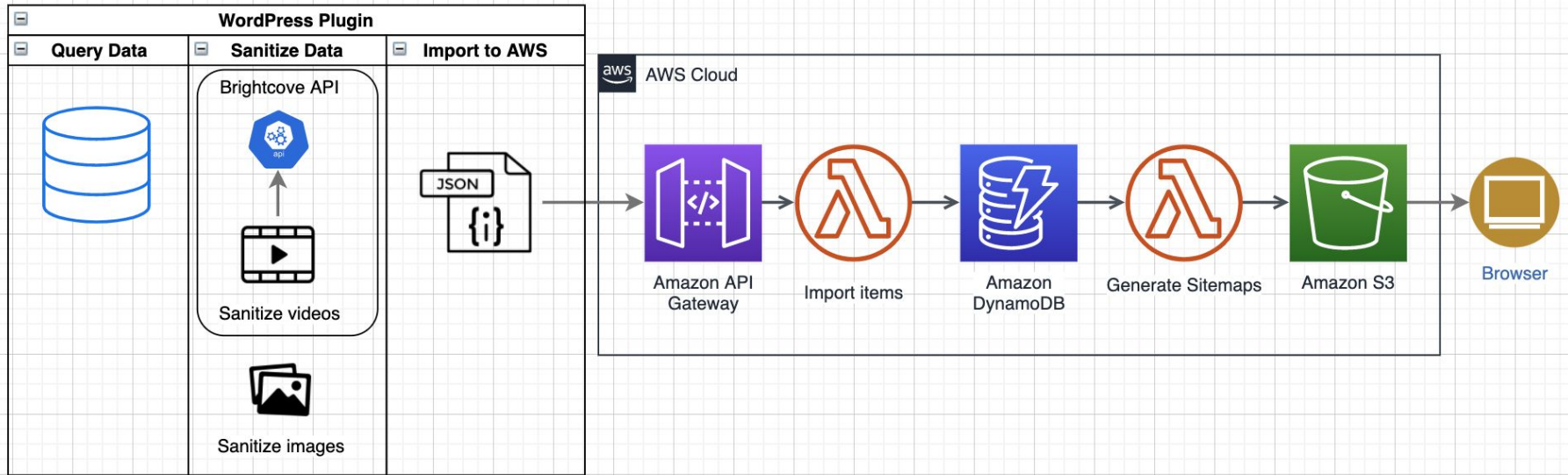
Example (create)

```
let params = {
  TableName: process.env.DYNAMODB_TABLE,
  Item: {
    id: data.id,
    type: data.type,
    url: data.url,
    images: data.images? data.images : [],
    videos: data.videos? data.videos : [],
    createdAt: data.created,
    updatedAt: data.modified
  },
};
```

```
dynamoDb.put(params, (error) => {
  if (error) {
    console.error(error);
    let response = {
      statusCode: error.statusCode || 501,
      headers: { 'Content-Type':
'text/plain' },
      body: 'Couldn\'t create the sitemaps
item.',
    });
  }
  response = {
    statusCode: 200,
    body: JSON.stringify(params.Item),
  };
  callback(null, response);
})
```

Sample code from Lambda function for create

Version 1.4



WordPress Plugin

/ Custom WP CLI

// One time bulk import

// Partial imports

/ Run heavy lifting processes from the import process

/ Propagate content updates

/ Retrieve sitemaps from S3

/ Update robots.txt

WP CLI (Import)

```
public function import_data( $args, $assoc_args ) {
    $is_reimport = false;
    $limit       = 1000;
    if ( isset( $assoc_args['action'] ) ) {
        $is_reimport = ( 'update' == $assoc_args['action'] ) ? true : false;
    }

    if ( isset( $assoc_args['type'] ) ) {
        $type = $assoc_args['type'];
    }
    if ( isset( $assoc_args['limit'] ) ) {
        $limit = $assoc_args['limit'];
    }
    if ( isset( $assoc_args['offset'] ) ) {
        $offset = $assoc_args['offset'];
        $start  = time();
        $ret    = Import_Data::import_data( $type, $is_reimport, $offset, $limit );
        $end    = time();
        if ( is_wp_error( $ret ) ) {
            WP_CLI::error( $ret );
        } else {
            $duration = $end - $start;
            $message  = 'cli ran. Duration (sec): ' . $duration;
            WP_CLI::success( $message );
        }
    }
}
```

```
else {
    $stats = Import_Data::get_stats( $types );
    foreach ( $stats as $type => $stat ) {
        if ( $stat['parts'] > 0 ) {
            $progress = \WP_CLI\Utils\make_progress_bar( 'Importing ' . $type, $stat['parts'] );
            for ( $i = 0; $i < $stat['parts']; $i++ ) {
                $ret = Import_Data::import_data( $type, $is_reimport, $i, $limit );
                $progress->tick();
            }
            $progress->finish();
        } else {
            WP_CLI::line( 'Importing ' . $type . ' ' . $stat['total'] . ' items.' );
            $start = time();
            $ret = Import_Data::import_data( $type, $is_reimport, 0, $limit );
            $end = time();
            if ( is_wp_error( $ret ) ) {
                WP_CLI::error( $ret );
            } else {
                $duration = $end - $start;
                $message = 'cli ran. Duration (sec): ' . $duration;
                WP_CLI::success( $message );
            }
        }
    }
}
}
```

```
$ wp lambda_sitemaps import_data --offset=15 --type=article --limit=5000
4995 out of 5000 succeeded.Success: cli ran. Duration (sec): 1445
$ wp lambda_sitemaps import_data --offset=70 --type=article --limit=1000
998 out of 1000 succeeded.Success: cli ran. Duration (sec): 287
```

```
$ wp lambda_sitemaps import_data
Importing article 1 % [==>
Importing article 2 % [====>
Importing article 4 % [=====>
Importing article 5 % [=====>
Importing article 6 % [=====>
Importing article 8 % [=====>
Importing article 9 % [=====>
Importing article 10 % [=====>

Importing article 89 % [=====>
Importing article 90 % [=====>
Importing article 100% [=====>] 514:05 / 559:11
```

```
public static function import_data( $type, $is_reimport = false, $part = 0, $split_limit = 1000 ) {
    if ( empty( $type ) ) {
        $post_types = [ 'article', 'feature' ]; //, 'commerce', 'gallery' ];
    } else {
        $post_types = $type;
    }
    $posts = self::get_data( $post_types, $part, $split_limit );
    if ( ! empty( $posts ) ) {
        $counter = 0;
        $success = 0;
        foreach ( $posts as $post ) {
            $response = self::import_post( $post, $type, $is_reimport );
            if ( is_wp_error( $response ) ) {
                $error_message = $response->get_error_message();
                echo "Something went wrong: $error_message";
            } else {
                if ( '200' == $response['response']['code'] ) {
                    $success++;
                }
            }
            $counter++;
        }
        echo $success . ' out of ' . $counter . ' succeeded.';
    }
}
```

```
public static function import_post( $post, $type = '', $is_reimport = false ) {  
    $item      = self::sanitize_post( $post, $type );  
    $data      = \wp_json_encode( $item );  
    $endpoint  = $api_base . '/items';  
    if ( true === $is_reimport ) {  
        $endpoint .= '/' . $post->ID;  
    }  
    $options = [  
        'timeout'      => 600,  
        'redirection' => 5,  
        'sslverify'    => false,  
        'data_format' => 'body',  
        'body'         => $data,  
        'headers'      => [  
            'Content-Type' => 'application/json',  
            //'Authorization' => 'Basic ' . base64_encode( $username . ':' . $password ), // TODO  
        ],  
    ];  
    $response = wp_remote_post( $endpoint, $options );  
    return $response;  
}
```

```
public static function sanitize_post( $post, $post_types ) {  
    $item = [];  
    $images = [];  
    $videos = [];  
    if ( ! empty( $post ) ) {  
        $post_id = $post->ID;  
        $post_content = $post->post_content;  
        $url = \get_permalink( $post_id );  
        $brightcove_ids = get_post_meta( $post_id, 'article_brightcove_ids', true );  
        $images = array_values( self::get_images( $post_content, $post_id ) );  
        $post_type = $post->post_type;  
        if ( ! empty( $brightcove_ids ) ) {  
            $videos = array_values( self::get_video_details( $post, $brightcove_ids ) );  
        }  
        $item = [  
            'id' => $post_id,  
            'type' => $post_type,  
            'url' => $url,  
            'created' => strtotime( $post->post_date ),  
            'modified' => strtotime( $post->post_modified ),  
        ];  
        if ( ! empty( $images ) ) {  
            $item['images'] = $images;  
        }  
        if ( ! empty( $videos ) ) {  
            $item['videos'] = $videos;  
        }  
    }  
    return $item;  
}
```

```

public static function get_video_details( $post, $brightcove_ids ) {
    require_once( BRIGHTCOVE_PATH . 'includes/api/class-bc-api.php' );
    require_once( BRIGHTCOVE_PATH . 'includes/class-bc-errors.php' );
    $bc_cms_api = new \BC_CMS_API();
    $videos     = [];
    if ( ! empty( $brightcove_ids ) ) {
        $bc_video_ids = maybe_unserialize( $brightcove_ids );
        foreach ( $bc_video_ids as $video_id ) {
            if ( ! isset( $videos[ $video_id ] ) ) {
                $bc_video      = $bc_cms_api->video_get( $video_id );
                $thumbnail_loc = '';
                if ( ! empty( $bc_video['images']['poster']['src'] ) ) {
                    $thumbnail_loc = $bc_video['images']['poster']['src'];
                }
                if ( empty( $thumbnail_loc ) && ! empty( $bc_video['images']['thumbnail']['src'] ) ) {
                    $thumbnail_loc = $bc_video['images']['thumbnail']['src'];
                }
                $video['thumbnail_loc'] = $thumbnail_loc;
                $video_sources         = $bc_cms_api->video_get_sources( $video_id );
                if ( isset( $video_sources[4]['src'] ) ) {
                    $content_loc = $video_sources[4]['src'];
                } elseif ( isset( $video_sources[0]['src'] ) ) {
                    $content_loc = $video_sources[0]['src'];
                }
                $video['content_loc']      = $content_loc;
                $video['title']            = substr( $bc_video['name'], 0, 100 );
                $video['description']      = $bc_video['description'];
                $video['duration']        = $bc_video['duration'];
                $video['publication_date'] = $bc_video['published_at'];
                if ( ! empty( $video['thumbnail_loc'] ) && ! empty( $video['description'] ) && ! empty( $video['title'] ) ) { // if the required fields are empty do not import.
                    $videos[ $video_id ] = $video;
                }
            }
        }
    }
    return $videos;
}

```

Propagating updates

```
add_action( 'save_post', [ $this, 'lambda_sitemap_save_action' ], 10, 3 );
add_action( 'wp_trash_post', [ $this, 'lambda_sitemap_trash_action' ], 10, 1 );
add_action( 'delete_post', [ $this, 'lambda_sitemap_trash_action' ], 10, 1 );

public function lambda_sitemap_save_action( $post_id, $post, $update ) {
    if ( empty( $_POST ) ) {
        return;
    }

    if ( isset( $post->post_type ) && ! ( in_array( $post->post_type, [ 'article', 'feature' ] ) ) ) {
        return;
    }

    if ( defined( 'DOING_AUTOSAVE' ) && DOING_AUTOSAVE ) {
        return $post_id;
    }

    $is_reimport = true;
    if ( 'publish' == $post->post_status ) {
        Import_Data::import_post( $post, $is_reimport );
    }
}

public function lambda_sitemap_trash_action( $post_id ) {
    Import_Data::delete_post( $post_id );
}
```

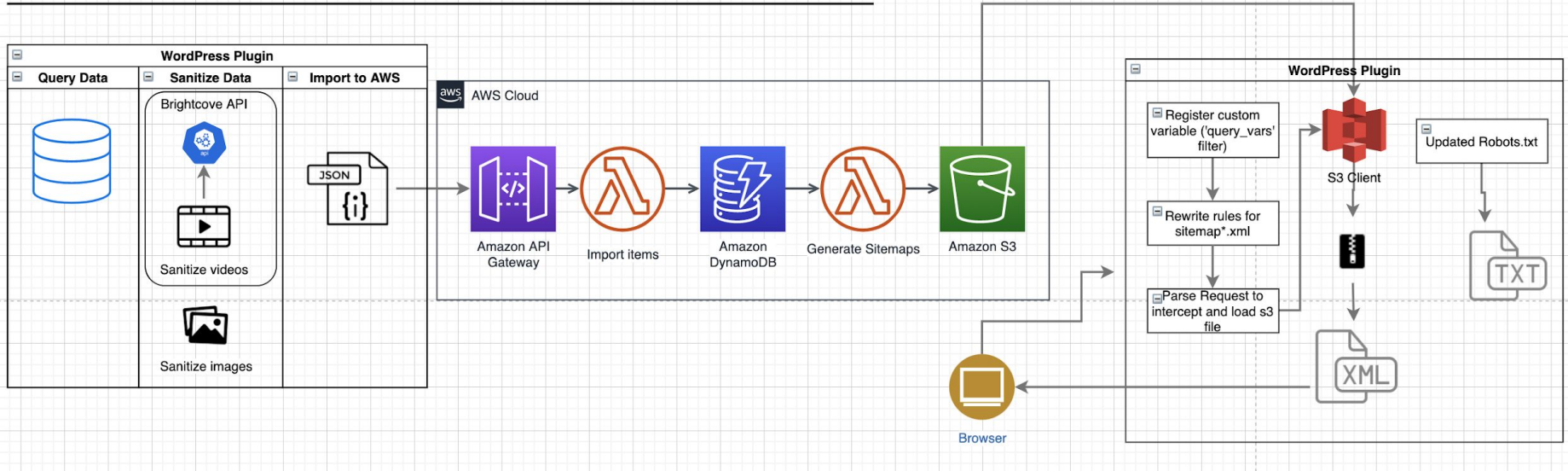
WP Hooks on any content updates to call the api endpoints to sync updates on dynamoDB.

Sitemaps: The Needs

- / Basic XML sitemaps ✓
- / Better promote rich media contents with image and video sitemaps ✓
- / Reduce amount, increase size of each sitemap file ✓
- / Sitemap update upon content updates ✓
- / Sitemaps available without timing out
- / Submit to search engines upon sitemaps generation

Retrieving the sitemaps

Version 1.5



```
add_filter( 'rewrite_rules_array', [ $this, 'lambda_sitemap_rewrite_rules' ], 9 );  
add_filter( 'query_vars', [ $this, 'lambda_sitemap_insert_query_vars' ] );  
add_action( 'parse_request', [ $this, 'lambda_sitemap_request_sitemap' ] );  
add_filter( 'robots_txt', [ $this, 'lambda_sitemap_do_robots' ], 1000, 2 );
```

```
public function lambda_sitemap_insert_query_vars( $vars ) {  
    array_push( $vars, 'ls_module' );  
    return $vars;  
}
```

```
public function lambda_sitemap_request_sitemap( $wp_query ) {  
    if ( isset( $wp_query->query_vars['ls_module'] ) ) {  
        $module = $wp_query->query_vars['ls_module'];  
    }  
    if ( ! empty( $module ) ) {  
        $this->_load_sitemap( $module );  
    }  
}
```

```
public function lambda_sitemap_rewrite_rules( $rules ) {  
    $rewrite_rules = array(  
        'sitemapindex\.xml$'           => 'index.php?ls_module=sitemapindex',  
        'sitemap\.xml$'               => 'index.php?ls_module=sitemap',  
        'sitemap_([a-z0-9_-]+)\.xml$' => 'index.php?ls_module=sitemap_${matches[1]}',  
        'news\.xml$'                  => 'index.php?ls_module=news',  
    );  
  
    $custom_rules = apply_filters( 'lambda_sitemaps_rewrite_rules', array() );  
    $rules        = array_merge( $custom_rules, $rewrite_rules, $rules );  
  
    return $rules;  
}  
  
public function lambda_sitemap_do_robots( $output, $public ) {  
    global $blog_id, $wpdb;  
  
    if ( '0' == $public ) {  
        return $output;  
    }  
  
    $output .= "\n";  
  
    $output .= 'Sitemap: ' . home_url( 'sitemapindex.xml' );  
  
    $output .= "\n";  
  
    return $output;  
}
```

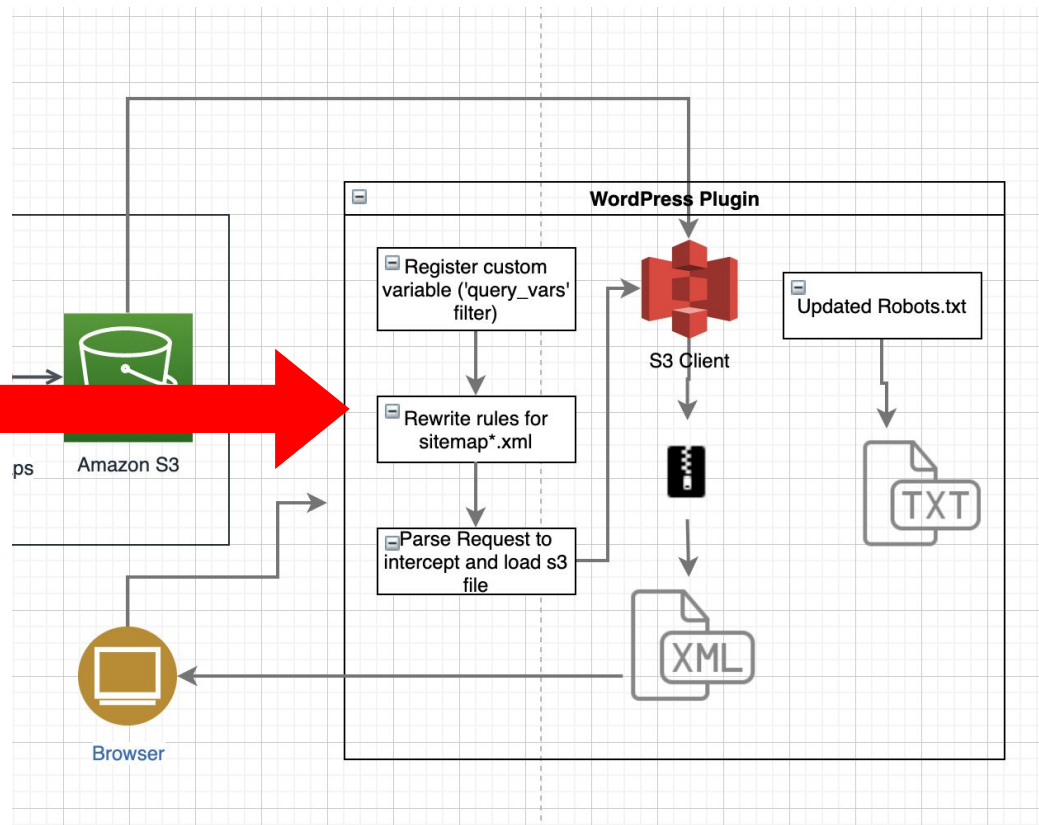
```
private function _load_sitemap( $module ) {  
    $module      = stripslashes( $module );  
    $part        = 0;  
    $module_name = '';  
    $sitemap_name = ( 0 < $part ) ? $module . '_' . $part : $module;  
    $module_name = $module;  
  
    $aws_s3 = AWS_S3::get_instance();  
    if ( $aws_s3->load_sitemap_from_s3( $module_name, $sitemap_name ) ) {  
        exit;  
    }  
}  
}
```

```
public function load_sitemap_from_s3( $module_name, $sitemap_name ) {
    $s3 = $this->s3();
    $s3->registerStreamWrapper();

    $size = filesize( 's3://' . $this->get_s3_bucket() . '/' . $sitemap_name . '.xml.gz' );
    if ( $stream = fopen( 's3://' . $this->get_s3_bucket() . '/' . $sitemap_name . '.xml.gz', 'r' ) ){
        while ( ! feof( $stream ) ) {
            $d = fread( $stream, $size );

            header( 'Content-Encoding: gzip' );
            header( 'Content-type: text/xml' );
            echo $d . PHP_EOL;
        }
        fclose( $stream );
    }
}
```

OVERKILL!



WordPress Plugin

/ Custom WP CLI

// One time bulk import

// Partial imports

/ Run heavy lifting processes from the import process

/ Propagate content updates

~~/ Retrieve sitemaps from S3~~

~~/ Update robots.txt~~

```
// Create sitemap index
let root = builder.create('sitemapindex', {encoding: 'UTF-8'}).att('xmlns',
'http://www.sitemaps.org/schemas/sitemap/0.9');

// add in each sitemap
_.each(chunks, function (chunk, index) {
  let sitemap = root.ele('sitemap');
  sitemap.ele('loc', 'https://' + process.env.BUCKET, + '.s3.amazonaws.com/' + sitemapGeneratedPrefix +
(index + 1) + '.xml.gz');
  sitemap.ele('lastmod', new Date().toISOString());
});

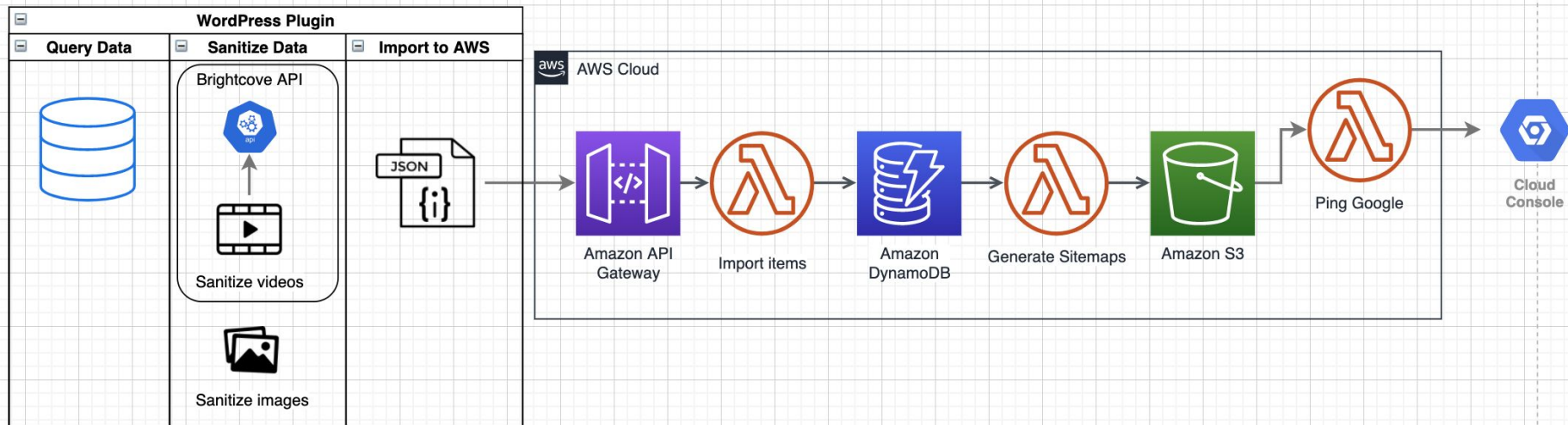
let xmlString = root.end({
  pretty: true,
  indent: ' ',
  newline: '\n',
  allowEmpty: false
});

await upload(xmlString, sitemapIndex);
```

Make sure these files are public

```
await when.promise(function (resolve, reject, notify) {
  // For Bing
  // rest.get('http://www.bing.com/ping?sitemap=' + event.site_url + '/' + sitemapIndex)
  rest.get('http://google.com/ping?sitemap=' + site_url + '/' + sitemapIndex)
    .on('success', function (data, response) {
      console.log('Google Ping: ' + data);
      resolve();
    })
    .on('fail', function (data, response) {
      console.log('Google Ping Error:', data);
      resolve();
    });
});
```

Version 1.6



Sitemaps: The Needs

- / Basic XML sitemaps ✓
- / Better promote rich media contents with image and video sitemaps ✓
- / Reduce amount, increase size of each sitemap file ✓
- / Sitemap update upon content updates ✓
- / Sitemaps available without timing out ✓
- / Submit to search engines upon sitemaps generation ✓

Next steps...

/ SQS Queue for bulk import

/ WordPress admin page for more control

/ Scheduled job for auto-generation

/ News sitemap

/ More...

Useful Links

/ <https://support.google.com/webmasters/answer/183668?hl=en>

/ <https://serverless.com/>

/ <https://aws.amazon.com/serverless/>

/ <https://github.com/frankadrian/lambda-sitemap-generator/>

Questions?

/ contact@ndevr.io

/ <https://ndevr.io>

/ Follow us on Twitter: [@ndevrinc](https://twitter.com/ndevrinc)

/ LinkedIn: <https://www.linkedin.com/company/ndevr>

/ Slides: To come...

Thank You