

adaptTo()

APACHE SLING & FRIENDS TECH MEETUP
2 - 4 SEPTEMBER 2019

Swagger AEM - An OpenAPI Specification For AEM
Cliffano Subagio and Michael Bloch, Shine Solutions Australia



About

- Cliffano Subagio and Michael Bloch
Consultants at Shine Solutions, Australia
AEM OpenCloud Core Contributors



Early challenges with AEM provisioning



Our first ‘AEM on the cloud’ project

- AEM 5.6 on AWS circa 2013-2014
- Infrastructure as code
- Cloud: auto-provisioning, auto-recoverable
- AEM tooling largely ‘Java-focused’
- Sysops teams use of non-Java tools



We were advised to use cURL

List of common AEM cURL commands

Search Adobe

Adobe AEM curl commands

Environment

AEM 6.x

Using cURL with AEM

Adobe CQ5 AEM cURL Commands

AEM: cURL commands

User Administering User Guide > Operations

Select an article...

AEM Tutorials

AEM Curl commands

3 min read

0 3 years ago Raj



Share on Face

Using curl command line when it comes to maintain want to update blueprint once again and some changes

What is Curl ?

Curl is a command line

comma

delete

-F":c

curl Upload a package AND install

Asked 7 months ago Active 7 months ago Viewed 280 times

Use curl to create, copy, move and update AEM pages

The Adobe Experience Manager (AEM) REST API makes it easy to script content operations from the terminal shell. It's not appropriate for production content but is tremendously useful for website content migration; for tasks like a bulk import of simple articles it can save a metric buttload of manual effort.

all curl command,

name="name of page.jsp



Integrate this harmless shell script with cURL!

```
set -o errexit
result=$(curl ...)
if [[${result} = *“error” ]]; then
    exit 123
fi
```



Is it really harmless?

```
set -o errexit <- Unreliable exit code  
result=$(curl ...)  
if [[${result} = *“error” ]]; then <- Fragile error checking  
    exit 123 <- Uninformative error result  
fi  
<- No success result
```



Problems with AEM Web ‘API’

- cURL is not always available everywhere
- Inconsistent response payload types
JSON, XML, HTML
- Unreliable status code
Response 200 on change password error
- Lack of integration with various tech stacks
The world has been polyglot for a while

What did we do to solve
this?



We built Swagger AEM



What is Swagger AEM?

- API specification for AEM endpoints
- OpenAPI version 3
- OpenAPI Generator
- API clients for 30+ languages



OpenAPI specification - metadata

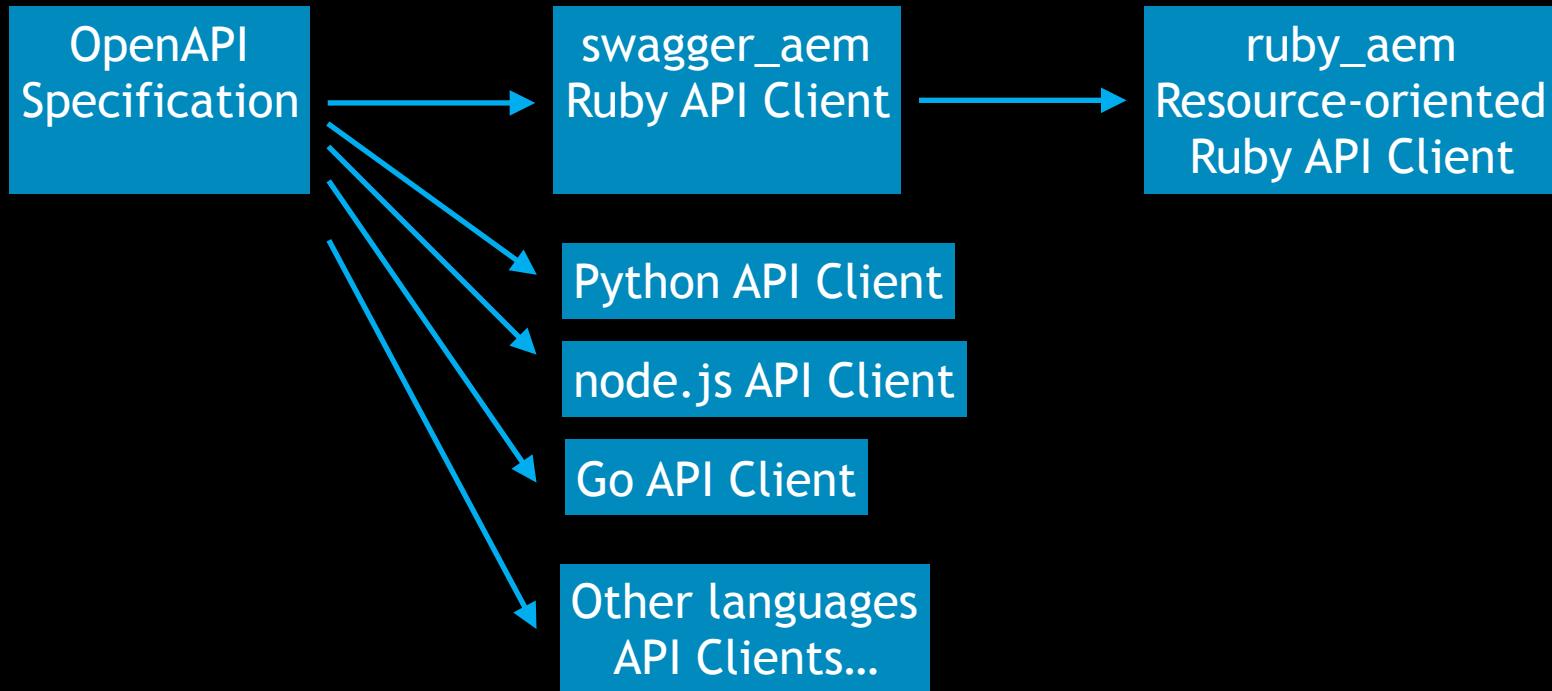
```
openapi: 3.0.0
info:
  title: Adobe Experience Manager (AEM) API
  description: Swagger AEM is an OpenAPI specification for Adobe
  version: 3.2.0-pre.0
  contact:
    name: Shine Solutions
    url: http://shinesolutions.com
    email: opensource@shinesolutions.com
security:
  - aemAuth: []
```



OpenAPI specification - path

```
paths:  
  "/crx/packmgr/service/.json/{path}":  
    post:  
      operationId: postPackageServiceJson  
      parameters:  
        - name: path  
          in: path  
          required: true  
          schema:  
            type: string  
        - name: cmd  
          in: query  
          required: true  
          schema:  
            type: string  
        - name: groupName  
          in: query  
          required: false  
          schema:  
requestBody:  
  content:  
    multipart/form-data:  
      schema:  
        type: object  
        properties:  
          package:  
            type: string  
            format: binary  
responses:  
  default:  
    description: Default response  
  content:  
    application/json:  
      schema:  
        type: string  
tags:
```

API Clients Generation





What is Ruby AEM?

- Resource-oriented design Ruby API
- Further abstraction from AEM endpoints
- Error and response objects



Result handling

```
package = aem.package('mygroup', 'mypkg', '1.2.3')
```

```
opts = { force: true }  
result = package.upload('/tmp', opts)
```

```
puts result.message  
puts result.response.status_code  
puts result.response.body  
puts result.response.headers
```



Error handling

```
begin
  package = aem.package('mygroup', 'mypkg', '1.2.3')
  opts = { force: true }
  result = package.upload('/tmp', opts)
rescue RubyAem::Error => err
  puts err.message
  puts err.result.response.status_code
  puts err.result.response.body
  puts err.result.response.headers
end
```



What else can Swagger AEM and Ruby AEM do?

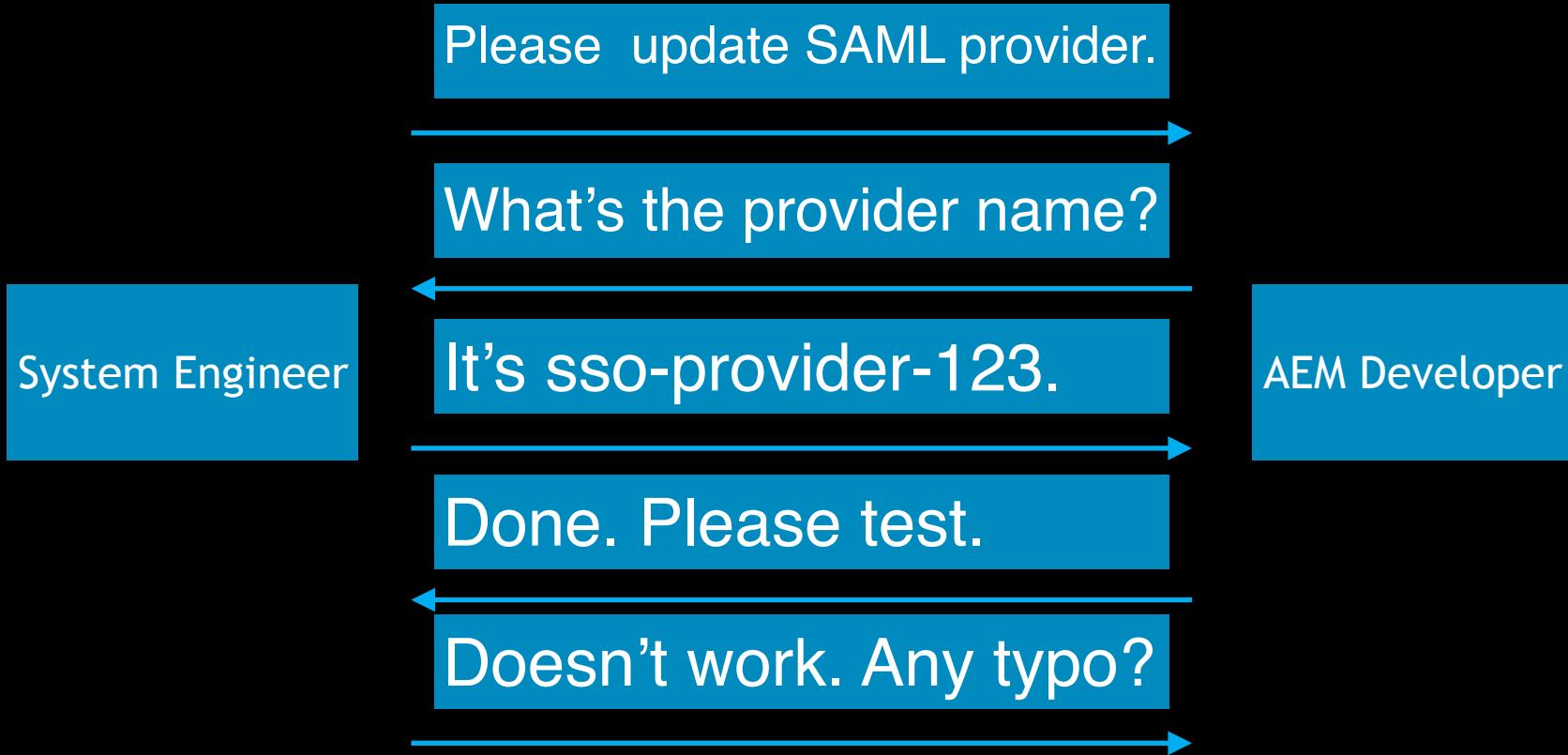
- Replication, flush, reverse replication agents
- Packages, users, groups, paths, nodes
- Certificates, truststores, keystores
- Resource readiness checks with retries
- Many more to add in the future!



We also built Swagger AEM OSGI



SAML config with AEM package





How about cURL?

- curl -v -u admin:admin -F 'jcr:primaryType=sling:OsgiConfig' -F 'keyStorePassword="password"' -F 'service.ranking=I"5002"' -F 'idpHttpRedirect=B"true"' -F 'createUser=B"true"' -F 'defaultRedirectUrl="/sites.html"' -F 'userIDAttribute="NameID"' -F 'defaultGroups=["content-authors"]' -F 'idpCertAlias="certalias____1471833774937"' -F 'addGroupMemberships=B"true"' -F 'path=["/"]' -F 'synchronizeAttributes=["givenName=profile/givenName,familyName=profile/familyName,mail=profile/email"]' -F 'clockTolerance=I"60"' -F 'groupMembershipAttribute="http://temp/variable/aem-groups"' -F 'idpUrl="https://accounts.google.com/o/saml2/idp?idpid=xyz"' -F 'logoutUrl="https://accounts.google.com/logout"' -F 'serviceProviderEntityId="AEMSSO"' -F 'handleLogout=B"true"' -F 'spPrivateKeyAlias=""' -F 'useEncryption=B"false"' -F 'nameIdFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"'

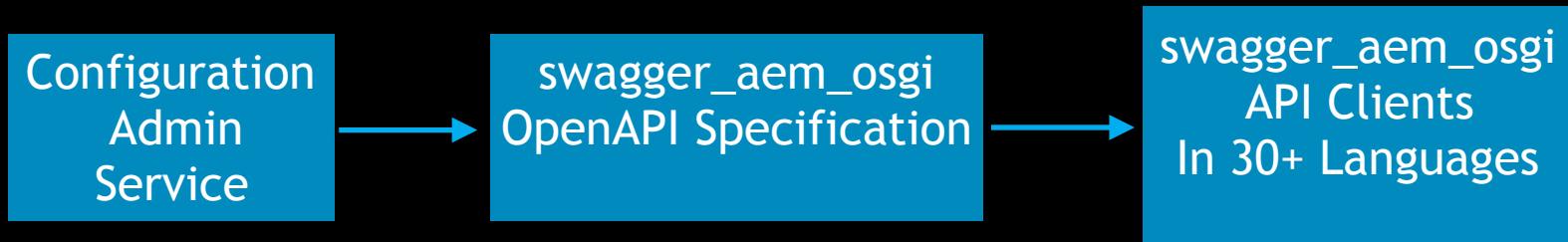


What is Swagger AEM OSGI?

- API specification for AEM OSGI configurations
- Converted from Configuration Admin Service
- 71k+ lines of generated OpenAPI v3 spec
- 12m+ lines of generated API clients
- Regenerated for each AEM version



API Clients Generation





Apache Felix Web Console

Adaptive Form and Interactive Communication Web Channel Configuration.name

Adaptive Form and Interactive Communication Web Channel Configuration.description

Adaptive Form and Interactive Communication

Web Channel

Configuration.showPlaceholder.name



⚠ Adaptive Form and Interactive Communication Web Channel Configuration.showPlaceholder.description (showPlaceholder)

Adaptive Form and Interactive Communication

100

Web Channel

Configuration.maximumCacheEntries.name



⚠ Adaptive Form and Interactive Communication Web Channel Configuration.maximumCacheEntries.description (maximumCacheEntries)

Adaptive Form and Interactive Communication

None

Web Channel

Configuration.af.scripting.compatversion.name



⚠ Adaptive Form and Interactive Communication Web Channel Configuration.af.scripting.compatversion.description (af.scripting.compatversion)

Adaptive Form and Interactive Communication



Web Channel

Configuration.makeFileNameUnique.name



⚠ Adaptive Form and Interactive Communication Web Channel Configuration.makeFileNameUnique.description (makeFileNameUnique)

Adaptive Form and Interactive Communication



Web Channel

Configuration.generatingCompliantData.name



⚠ Adaptive Form and Interactive Communication Web Channel Configuration.generatingCompliantData.description (generatingCompliantData)

Configuration Information

Persistent Identity (PID)

Adaptive Form and Interactive Communication Web Channel Configuration

Configuration Binding

Unbound or new configuration

Cancel

Reset



Converted into OpenAPI specification

```
paths:  
  "/system/console/configMgr/Adaptive Form":  
    post:  
      parameters:  
        {...}  
        - name: showPlaceholder  
          in: query  
          schema:  
            type: boolean  
        - name: maximumCacheEntries  
          in: query  
          schema:  
            type: integer  
        - name: af.scripting.compatversion  
          in: query  
          schema:  
            type: string  
        - name: makeFileNameUnique  
          in: query  
          schema:  
            type: boolean  
        - name: generatingCompliantData  
          in: query  
          schema:  
            type: boolean  
  
      operationId: adaptiveFormAndInteractiveCommunicationWebChannelConfigur  
      tags:  
        - configmgr  
      responses:  
        '200':  
          description: Successfully retrieved configuration parameters  
          content:  
            application/json:  
              schema:  
                "$ref": "#/components/schemas/adaptiveFormAndInteractiveComm  
        '302':  
          description: Default response  
          content:  
            text/plain:  
              schema:  
                type: string  
        default:  
          description: Default response  
          content:  
            application/json:  
              schema:  
                type: string
```



Converted into Ruby API client code

```
def adaptive_form_and_interactive_communication_web_channel_configuration_with_http_info(opts = {})
  if @api_client.config.debugging
    @api_client.config.logger.debug 'Calling API: ConfigmgrApi.adaptive_form_and_interactive_communication_web_channel_configuration'
  end
  # resource path
  local_var_path = '/system/console/configMgr/Adaptive Form and Interactive Communication Web Channel Configuration'

  # query parameters
  query_params = opts[:query_params] || {}
  query_params[:'post'] = opts[:'post'] if !opts[:'post'].nil?
  query_params[:'apply'] = opts[:'apply'] if !opts[:'apply'].nil?
  query_params[:'delete'] = opts[:'delete'] if !opts[:'delete'].nil?
  query_params[:'action'] = opts[:'action'] if !opts[:'action'].nil?
  query_params[:'$location'] = opts[:'location'] if !opts[:'location'].nil?
  query_params[:'propertylist'] = @api_client.build_collection_param(opts[:'propertylist'], :csv) if !opts[:'propertylist'].nil?
  query_params[:'showPlaceholder'] = opts[:'show_placeholder'] if !opts[:'show_placeholder'].nil?
  query_params[:'maximumCacheEntries'] = opts[:'maximum_cache_entries'] if !opts[:'maximum_cache_entries'].nil?
  query_params[:'af.scripting.compatversion'] = opts[:'af_scripting_compatversion'] if !opts[:'af_scripting_compatversion'].nil?
  query_params[:'makeFileNameUnique'] = opts[:'make_file_name_unique'] if !opts[:'make_file_name_unique'].nil?
  query_params[:'generatingCompliantData'] = opts[:'generating_compliant_data'] if !opts[:'generating_compliant_data'].nil?

  # header parameters
  header_params = opts[:header_params] || {}
  # HTTP header 'Accept' (if needed)
  header_params['Accept'] = @api_client.select_header_accept(['application/json', 'text/plain'])
```



Demo: SAML provisioning

Code flow





Demo: AEM package installation via voice



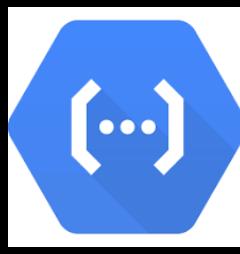
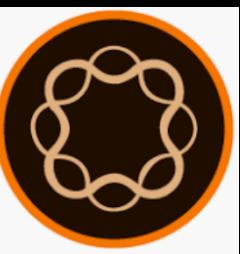
D

```
{ "queryText": "Install package ABC",  
  "parameters": {  
    "packageName": "ABC"  
  },  
  "intent": {  
    "action": "com.googleassistant.intent.actionSTALL-PACKAGE"
```

/crx/packmgr/service/.json/{path}

s.intent.TEXT",

'OICE",
"Install package ABC" ge ABC"



200

```
{  
  "fulfillmentText": "Pack  
  "payload": { "google":  
    "simpleResponse": {  
      "textToSpeech": "P  
      "displayText": "Pac
```

```
  "expectedInputs": [  
    {  
      "inputPrompt": {  
        "richInitialPrompt": {  
          "items": [ { "simpleResponse": {  
            "textToSpeech": "Package ABC has been installed"
```

"Package ABC has been installed"



In conclusion...



What have we learned?

- API specifications enable API clients
- API clients enable richer integration
- AEM provisioning doesn't have to be Java-focused
- Think outside the box!
Lots of DevOps tools to integrate



Most important of all...

You are welcome to join this effort
and contribute to benefit
the wider AEM community!



Thank you and any questions ?



Links

- <https://github.com/shinesolutions/adaptto-2019-demo>
- <https://github.com/shinesolutions/swagger-aem>
- <https://github.com/shinesolutions/swagger-aem-osgi>
- https://github.com/shinesolutions/ruby_aem
- <https://aemopencloud.io/>
- <https://www.openapis.org/>
- <https://openapi-generator.tech/>