Shuttle swagger service specification

# Structure des appels

Pour le module natif rest

* <server>/rest/cloudapi/xxxxxxx

Pour le service CURL :

* TASKNAME= ***MyRepo***.**MyDatamart**.**MyBackendRest**
* ${SHUTTLE\_URL}/rest/backend/***MyRepo***/**MyDatamart/MyBackendRest/myObject**

# Wrapper : Service rest / Service Curl

*Gestion de la doc : openapi.json*

***Côté client rest***

curl -v --http1.1 -X **GET** -L ${SHUTTLE\_URL}/rest/backend/***MyRepo***/**MyDatamart/MyBackendRest/openapi.json**-H 'Authorization: Basic YWRtaW46cGFzc3dvcmQ= ' \  
-H 'Accept: application/json'

***Côté wrapper java***

TASKNAME= ***MyRepo***.**MyDatamart**.**MyBackendRest branchée sur la fonction jexl backend**

jsonCurl = {  
 event : **get.openapi.json**  
}

curl -v --http1.1 -F user=${SHUTTLE\_USER} -F password=${SHUTTLE\_PASSWORD} -F waitCompletion=yes -F action=execute -F task=${TASKNAME} -F json="$jsonCurl" ${SHUTTLE\_URL}/curl

renvoie le openapi.json

***Coté Backend Jexl***

* **Function backend**

var **backend**(json) -> {

//valid event vs hashmap

//call du endpoint identifié avec le reste du json

}

* **hashMap des endpoints**

entrypoints = {

" get.openapi.json" : getOpenAPI

};

*Gestion du backend pour les autres fonctions*

***Exemple : create repository***

***Côté client rest***

jsonRest = {  
 "name": "filesystemRepo",

"implementation": "filesystem",

"keepSource": true  
}

curl -v --http1.1 -X **POST** -L ${SHUTTLE\_URL}/rest/backend/***MyRepo***/**MyDatamart/MyBackendRest/repository** \  
-H 'Authorization: Basic YWRtaW46cGFzc3dvcmQ= ' \  
-H 'Content-Type: application/json' \  
-H 'Accept: application/json' \  
-d $jsonRest

***Côté wrapper java***

jsonCurl = {  
 event : post.repository,  
 "name": "filesystemRepo",

"implementation": "filesystem",

"keepSource": true  
}

TASKNAME= ***myRepo***.**MyDatamart**.**MyBackendRest**

curl -v --http1.1 -F user=${SHUTTLE\_USER} -F password=${SHUTTLE\_PASSWORD} -F waitCompletion=yes -F action=execute -F task=${TASKNAME} -F json="$jsonCurl" ${SHUTTLE\_URL}/curl

***Coté Backend Jexl***

* **Function backend**

var **backend**(json) -> {

//valid event vs hashmap

//call du endpoint identifié avec le reste du json

}

* **hashMap des endpoints**

entrypoints = {

" get.openapi.json" : getOpenAPI

,"post.repository”, postRepository

};