

## **Guide MQ du 08/04/21**

- **Migration OS**
- **Inventaire dynamique**

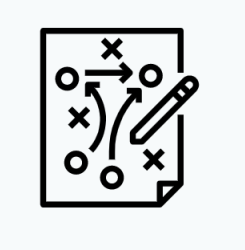


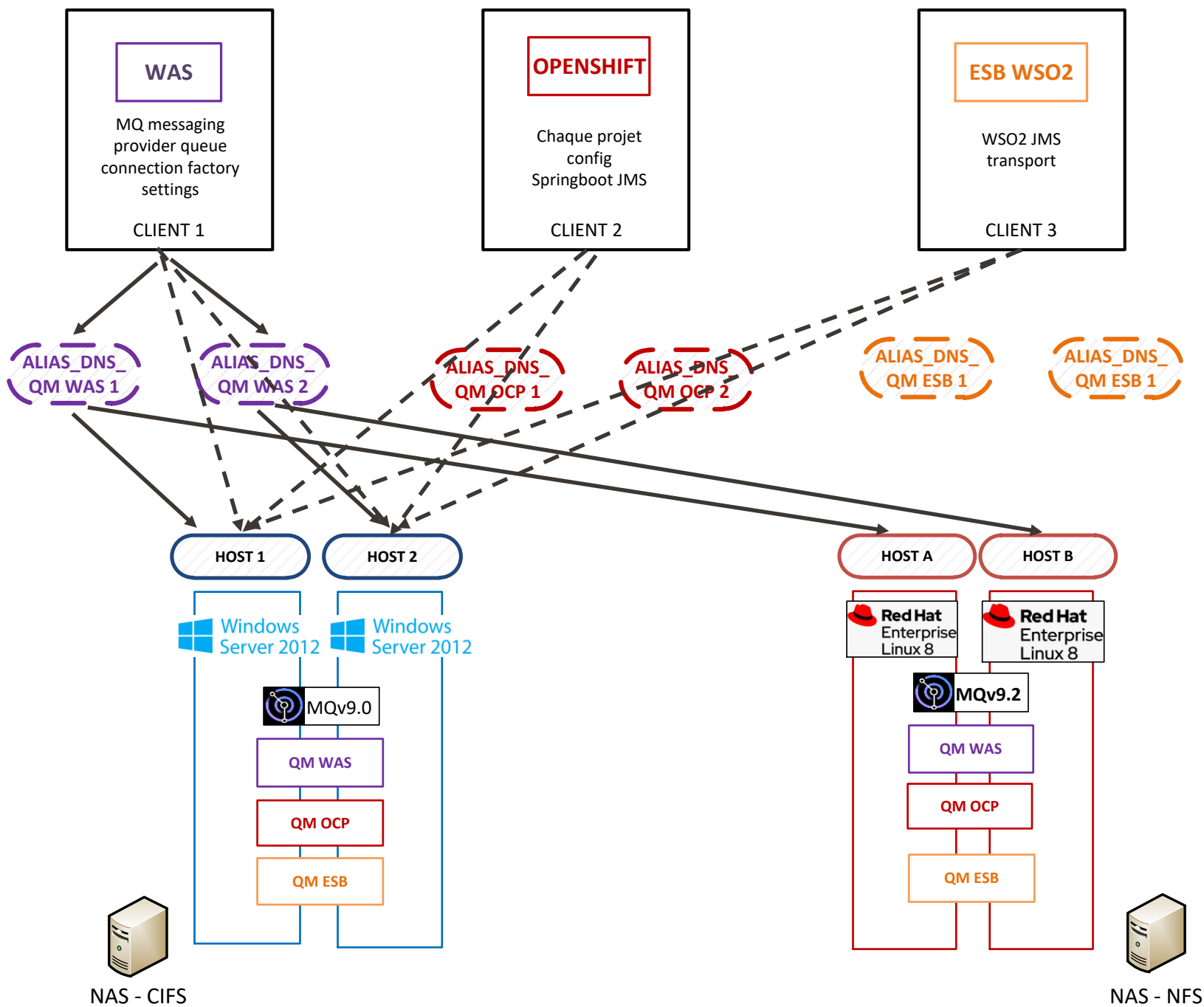
IBM MQ

# 01.

**Migration OS =>  
Reconstruction d'un Queue  
Manager**

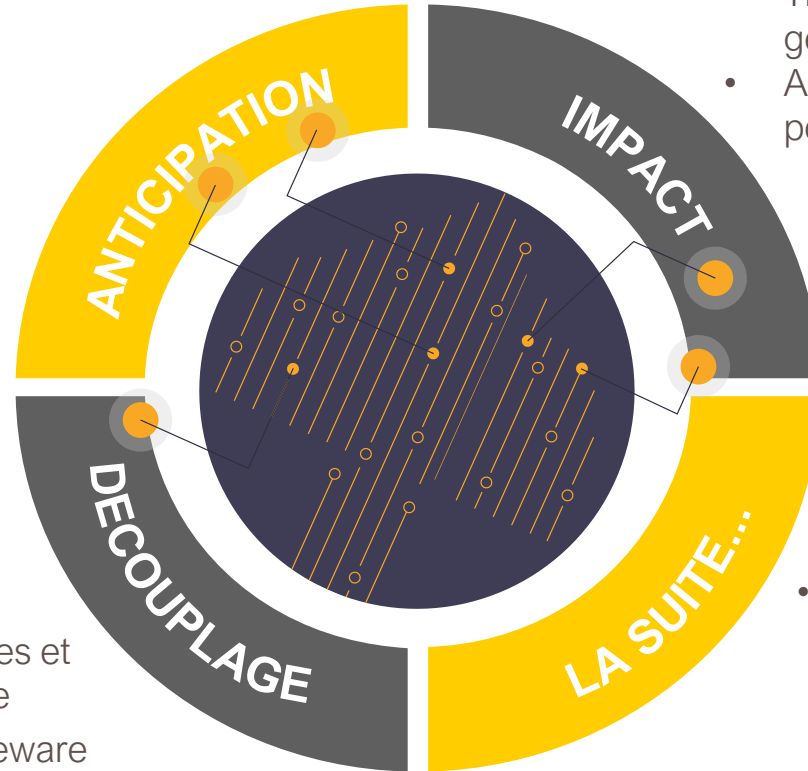
# Cycle de vie





# Alias DNS par queue manager

- Les admin MQ peuvent installer les serveurs en avance
- Les Clients peuvent migrer progressivement sans synchronisation de planning



- Transparent pour les projets, ne pas gérer de planning de synchro
- Approche pas à pas limitant les impacts pour chaque migration

- On dissocie les adhérences et la mutualisation éventuelle
- MAJ Projet vs MAJ Middleware

- Préparation des prochaines migrations, Anonymisation de la couche système, vers MQ Container ?

# Points d'attention reconstruction queue manager

- Identification de tous les clients

```
dis chstatus(ws02)
  1 : dis chstatus(ws02)
AMQ8417I: Display channel status details.
CHANNEL (WS02)                CHLTYPE (SVRCONN)
CONNAME (10.10.119.19)        CURRENT
STATUS (RUNNING)              SUBSTATE (RECEIVE)
```

- Anticiper les créations d'Alias DNS par QM
- Installation / Création queue manager
- Reprise des configurations MQ : dmpmqcfg + qm.ini, configs spécifiques (ex : MAXUMSGS, MAXMSGL)
- Certificats TLS
- Conserver les ports

## Post Bascule :

- Désynchronisation channel / réalignement ou RESET des séquences

⇒ cf site+doc Luc-Michel Demey :

[https://demey-consulting.fr/wp-content/uploads/2019/12/backup\\_mq.pdf](https://demey-consulting.fr/wp-content/uploads/2019/12/backup_mq.pdf)

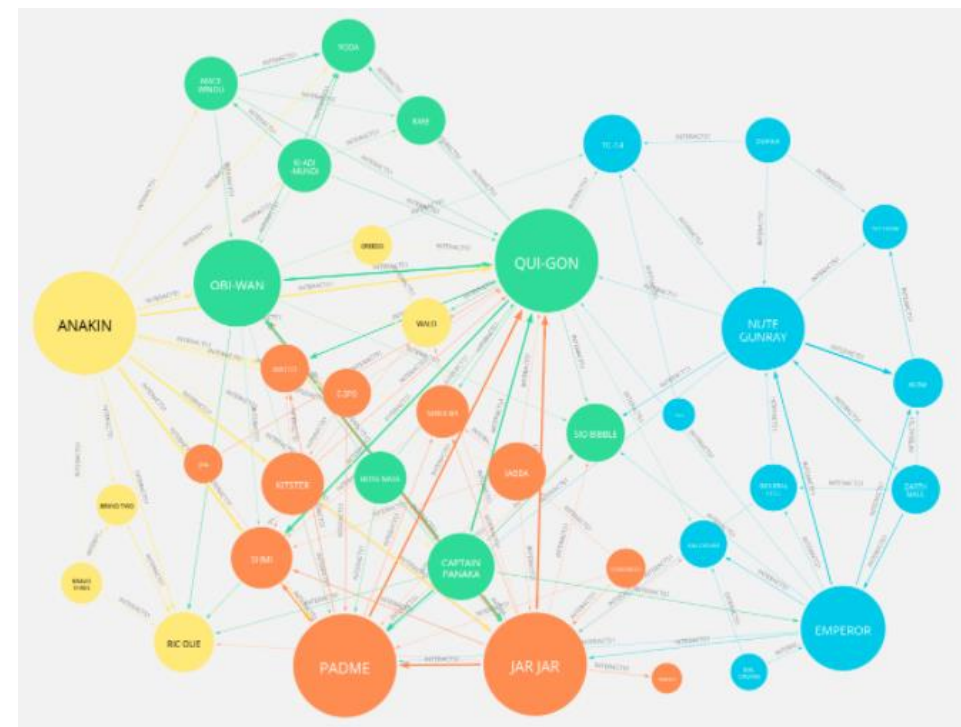
# 02.

## Inventaire dynamique

# Besoin d'un inventaire dynamique

- Avoir une vue complète des dépendances entre les serveurs, les socles et les applications
- Dynamique : Eviter une CMDB à mettre à jour
- Personnalisé et complet du SI
- Fichiers json, kafka, API Rest

Vue dynamique des dépendances avec Neo4j



=> Export format JSON pour configs MQ



# Comment récupérer les infos de mon infra MQ ?

Exemples de commandes en API REST

- Obtenir la version MQ installée :

```
[ansible@████████ ~]$ curl -k https://████████:9443/ibmmq/rest/v1/admin/installation -X GET -u ██████████
Enter host password for user '████████':
{"installation": [{
  "name": "Installation1",
  "platform": "unix",
  "version": "9.2.0.1"
}]}
```

- Lister les queues manager :

```
[ansible@████████]$ curl -k https://████████:9443/ibmmq/rest/v1/admin/qmgr -X GET -u ██████████
Enter host password for user '████████':
{"qmgr": [
  {
    "name": "MQ████████01",
    "state": "running"
  },
  {
    "name": "MQ████████02",
    "state": "running"
  },
  {
    "name": "████████",
    "state": "running"
  }
]}
```

## API REST :

- Interrogeable à distance (pas besoin de connexion aux serveurs)
- Format JSON facile à parser

# ANSIBLE – rôles – template jinja2

- Utilisation de **roles** et de template **jinja2**
- Récupération des informations via **uri\_module**
- user/password technique dans **Ansible vault**

```
uri:
  url: https://{{inventory_hostname}}:9443/ibmmq/rest/v1/admin/installation
  user: "{{mq_rest_user}}"
  password: "{{mq_rest_passwr}}"
  method: GET
  validate_certs: false
  force_basic_auth: yes
  register: response
  ignore_errors: yes

- debug: var=response

- set_fact:
  version_mq: "{{ response.json.installation[0].version }}"
```

# Exemple JSON généré

```
{
  "resource":{
    "vendor":"IBM",
    "hosting":"ON-PREM",
    "environments":[
      "ppd1"
    ],
    "id":"ppd1_9.1.0.6_MYQMGR",
    "platform":"MQ",
    "state":"running",
    "version":"9.1.0.6",
    "dependencies":{
      "to":[
        "myhost"
      ]
    },
    "type":"queuemanager",
  ]
}
```

# Difficultés

- Cas du Multi instance à gérer : « Running as standby »

```
{"qmgr": [  
  {  
    "name": "MYQMGR01",  
    "state": "runningAsStandBy"  
  }  
],  
...}
```

- Pas possible de relevé d'information, mais dépendance « Serveur » à afficher

```
[ansible@██████████] $ curl -k https://██████████:9443/ibmmq/rest/v1/admin/qmgr/██████████ -X GET -u A██████████  
Enter host password for user '██████████':  
{"error": [{  
  "action": "Use the serialized exception in the 'exception' field, if any, to get more information about the problem. Rectify the problem, which might require o  
perations to be performed on the queue manager, and resubmit the request.",  
  "completionCode": 2,  
  "explanation": "The REST API caught an exception when communicating with the queue manager. This might indicate that there was a problem with the queue manager  
, or that a parameter that was passed was incorrect.",  
  "message": "MQWB0004E: An internal error occurred while communicating with the queue manager. The root MQ reason code was 2543 : MQRC_STANDBY_Q_MGR.",  
  "msgId": "MQWB0004E",  
  "reasonCode": 2543,  
  "type": "pcf"}]}
```

**MERCI pour votre attention.**