



# Messagerie

Janvier 2019 - KP-1

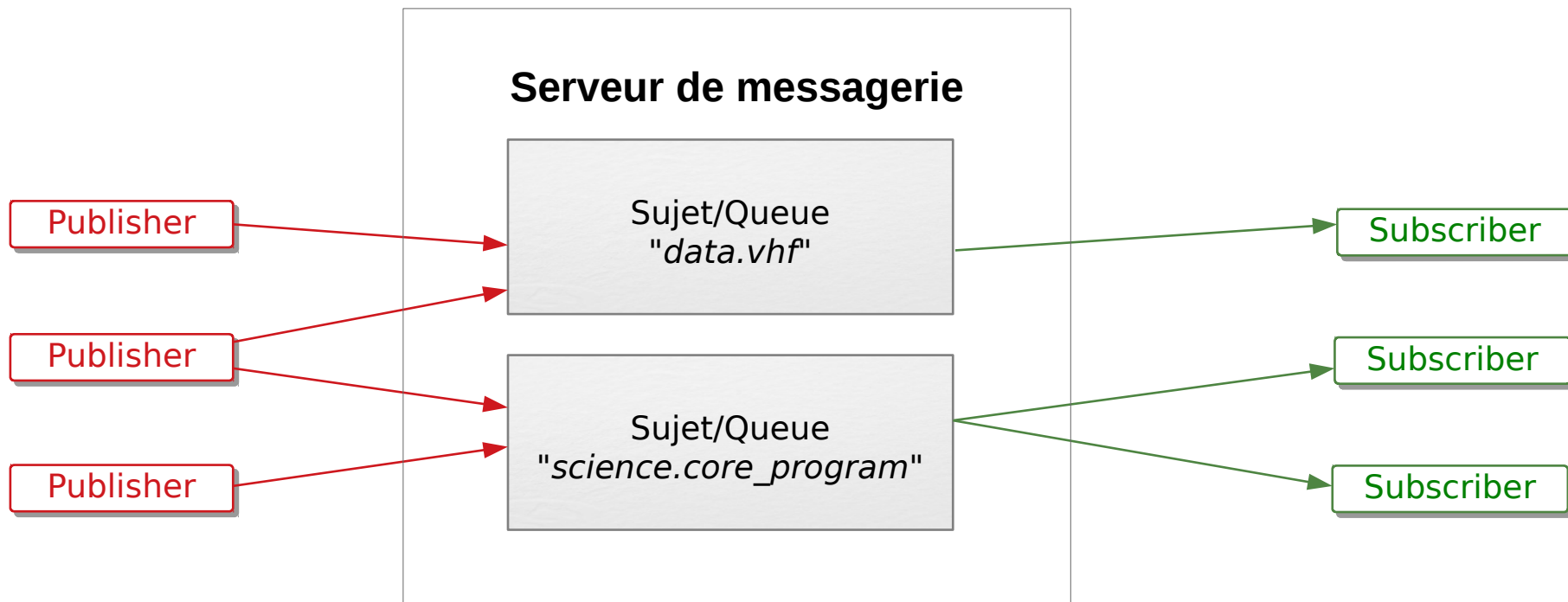




# La communication entre services

| ONE-TO-ONE   | ONE-TO-MANY   |
|--|---|
| <ul style="list-style-type: none"><li>- Le service communique directement avec un autre service</li><li>- Le destinataire est connu précisément</li><li>- Une réponse est attendue</li><li>- Exemples : Demande de mise à disposition de données,</li></ul> <p style="text-align: center;">→ <b>REST</b></p> | <ul style="list-style-type: none"><li>- Un service communique une information susceptible d'intéresser plusieurs autres services</li><li>- Publication de message sans se préoccuper de l'existence de destinataires</li><li>- Exemple : Notification d'arrivées de nouvelles données, diffusion d'informations d'états du service,...</li></ul> <p style="text-align: center;">→ <b>MESSAGERIE</b></p> |

## Architecture publish-subscribe



- › Accessibilité du serveur à tous (?) les services
- › Contrôle des queues
- › Uniformisation du format des messages

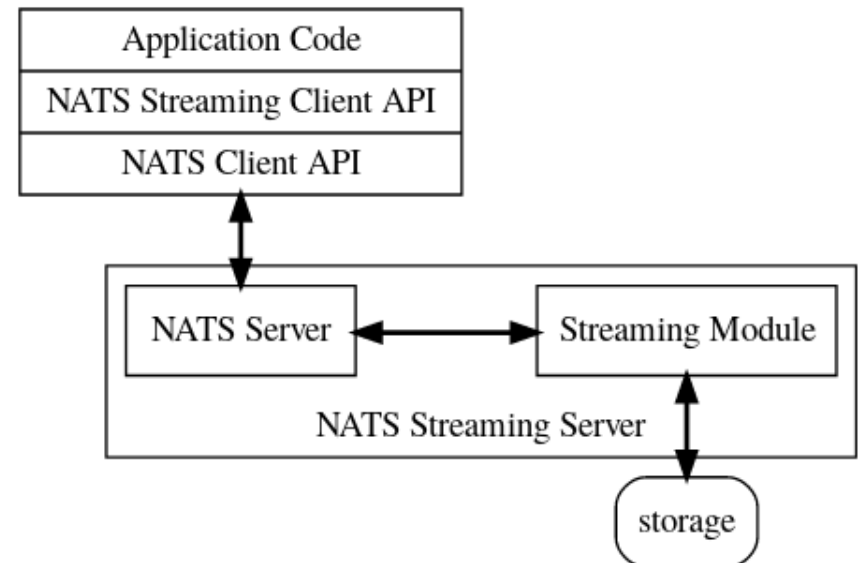


## Choix de NATS comme système de messagerie

- Open-source, performant, *scalable* et simple d'utilisation
- Clients synchrones et asynchrones disponibles pour Python et Java

## Utilisation de NATS Streaming

- NATS + Module Streaming
- La possibilité de souscriptions *durables*
- Le contrôle des queues





## Structure à deux niveaux : *domain.program*

- Possibilité de souscrire à plusieurs queues en utilisant des wildcards (e.g. *domain.\**)
- Queues définies au 13/12/2018 :

| domaine             | programme   |
|---------------------|---|
| activity<br>science | calibration<br>core_program<br>general_program<br>infrastructure<br>too |
| data                | mdb<br>vhf<br>xband   |



## Exemples de messages

### ServiceStatus message

```
{
  "ServiceStatus": {
    "descriptor": {
      "name": "WEBSOCKET_GATEWAY",
      "creation_date": "2019-01-09T14:20:01",
      "uri": "svom://natsgateway",
      "instrument": "ANY",
      "mode": "ANY",
      "version": 1.0
    },
    "activity": "Running",
    "date": "2019-01-09T14:24:09"
  }
}
```



## DataNotification message

```
{
  "DataNotification": {
    "source": "VHF",
    "data_type": "ECLALERTL1",
    "date": "2018-12-04T14:59:04.535Z",
    "message": "{\\"CcsdsTmHeadr\\": {\\"CcsdsApid\\": 576, \\"CcsdsGFlag\\": 3,
      \\"CcsdsCounter\\": 1, \\"CcsdsPLength\\": 87}, \\"VhfTmHeader\\":
      {\\"DPacketTimeS\\": 61}, \\"IObservId\\": {\\"IObsType\\":
      0, \\"NObsNumber\\": 1}, \\"Alert_MsgCount\\": 0,
      \\"Alert_TimeTbAbsSeconds\\": 325, \\"DalertTBAbF\\": 4410433,
      \\"Alert_TimeTb\\": 1279611476, \\"Alert_SlewRequest\\": 76,
      \\"Alert_SkyTheta\\": 3211333, \\"Alert_SkyPhi\\": 4410433,
      \\"Alert_CatSrcNum\\": 19525, \\"Alert_SkyYpix\\": 82,
      \\"Alert_SkyZpix\\": 84, \\"Alert_SkyVar\\": 82,
      \\"Alert_Quality\\": 76, \\"Alert_TriggerCriterium\\": 6716,
      \\"Alert_DetCounts\\": 4543060, \\"Alert_DetBkg\\": 4993280,
      \\"Sat_PositionX\\": 3124.7658, \\"Sat_PositionY\\": 5172.0,
      \\"Sat_PositionZ\\": 46399.0, \\"Sat_AttitudeQ0\\": 204.25,
      \\"Sat_AttitudeQ1\\": 33.2685, \\"Sat_AttitudeQ2\\": 1.8e-09,
      \\"Sat_AttitudeQ3\\": 0.0, \\"Sat_AttQuality\\": 0,
      \\"Sat_AttRef\\": 0, \\"ZVhfPEC\\": 52688}",
    "resource_locator": "32fef21a2cea43652fa28346de70b8"
  }
}
```



## Pour DC0

- Serveur + **serveur Streaming**
- Définition des **queues**
- **Librairie python** pour faciliter l'utilisation de NATS Streaming :

### Exemple d'utilisation de **natsio.py**

```
from svom.messaging.natsio import NatsIo

nats_client = NatsIo(host='127.0.0.1', port=4222, user='svom', password='secret')
nats_client.publish('data.vhf', 'This is not a valid message')
```

- Le projet *messaging* sur gitlab :  
<https://drf-gitlab.cea.fr/svom/common-services/messaging>



## Pour la suite

- Accès au serveur (SSL? Nécessité d'accès depuis l'extérieur?)
- Accès non contrôlé aux queues
- Protection contre la chute du serveur de messagerie (solution : **cluster**)

