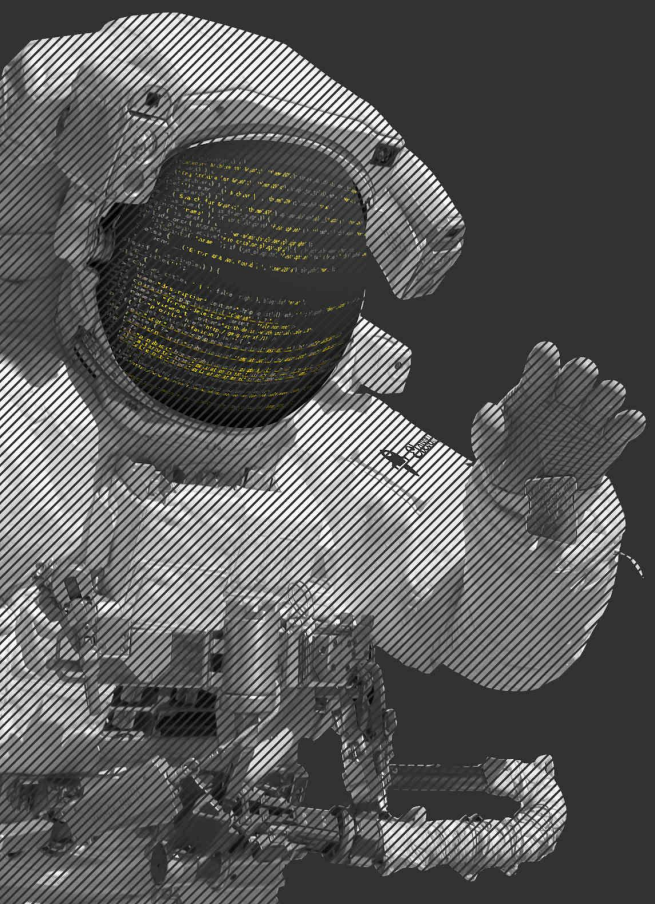


Eleven labs
présente

REX MICROSERVICE DE SYNCHRONISATION VIA RABBITMQ

#sfpot

QUI SUIS-JE ?

Thierry Thuon, aka lepiaf

Développeur chez Eleven-Labs au sein du studio

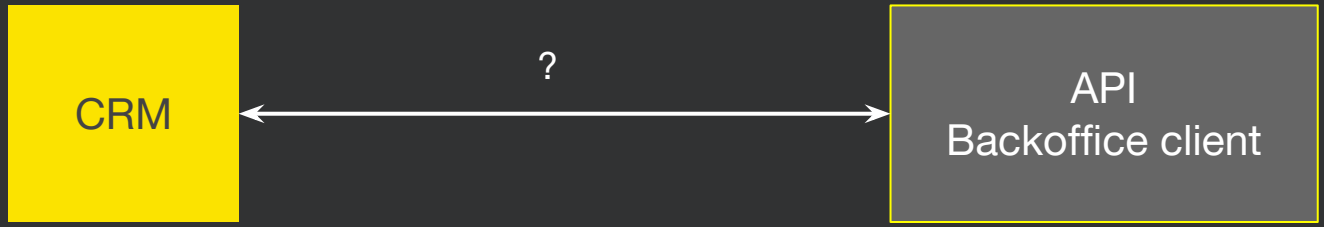
github.com/lepiaf

INTRODUCTION

- Données dispersées à plusieurs endroits.
- Connecter les systèmes.
- Cas concret : synchronisation d’un compte d’une entreprise entre le CRM et la plateforme eCommerce.

POSONS LE DÉCORS

Comment synchroniser des données entres les deux domaines ?



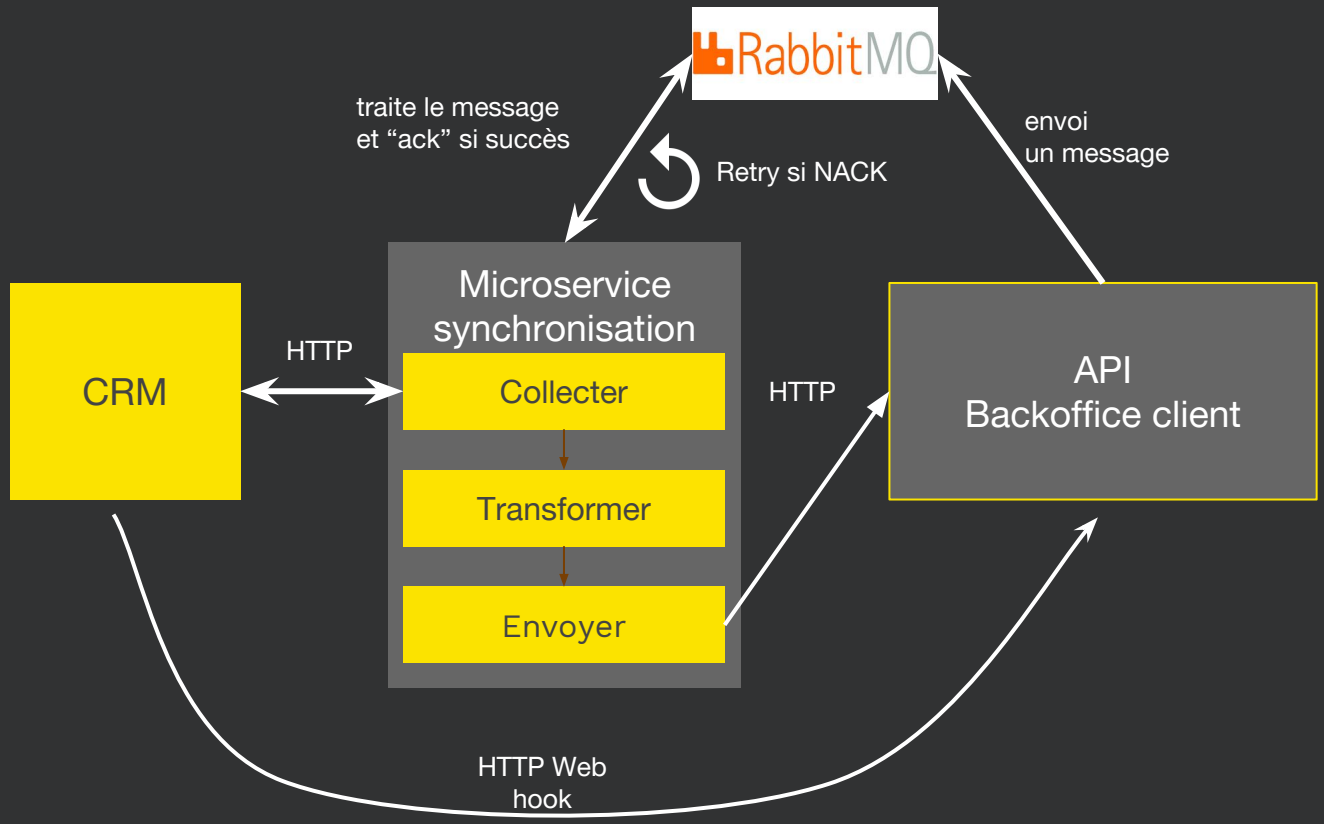
UN MICRO-SERVICE DE SYNCHRONISATION

Sa mission :

- **collecter** les données depuis le CRM
- **transformer** les données
- **envoyer** le résultat à l'API client

Et inversement.

ARCHITECTURE

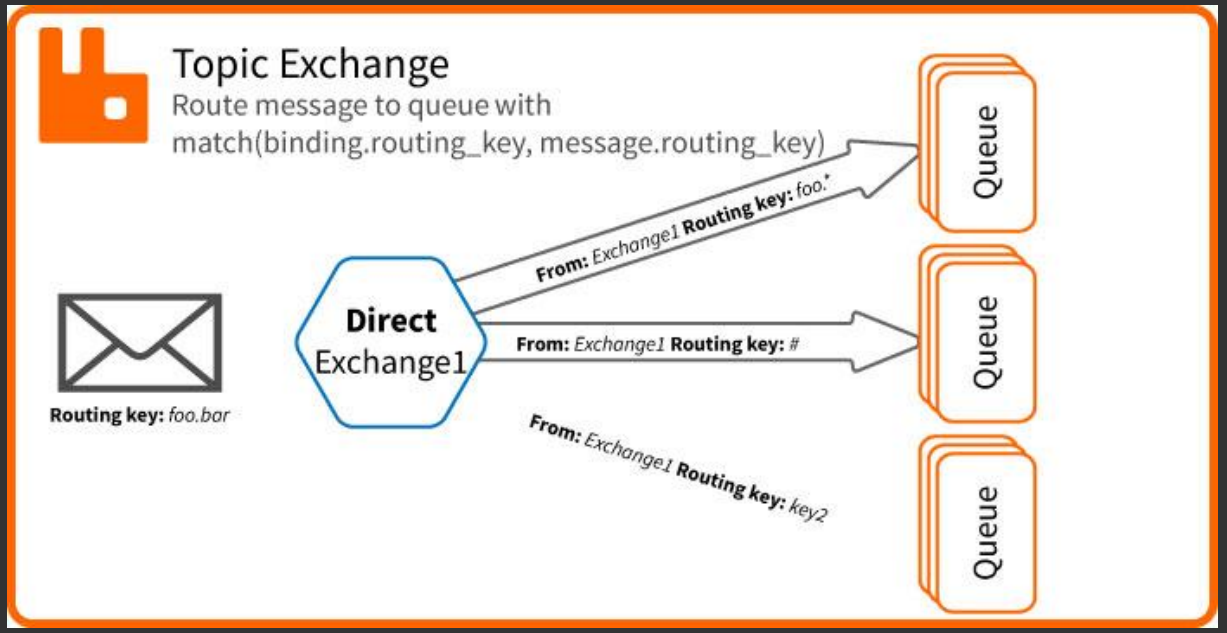


COMPOSANTS

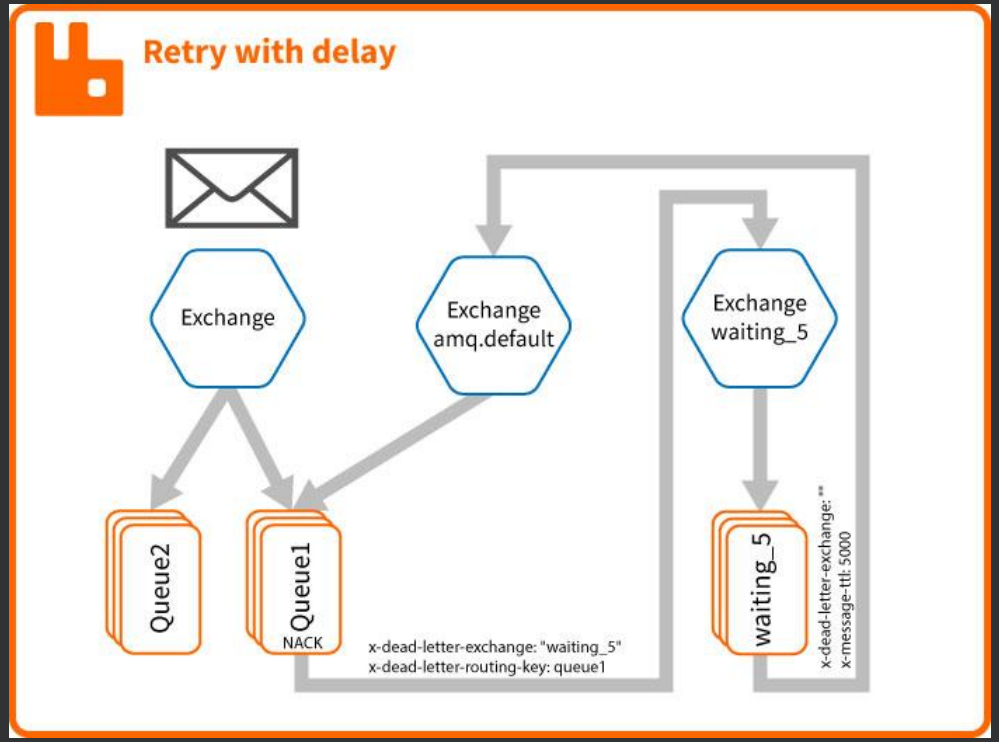
```
{
    "symfony/symfony": "^3.1",
    "swarrot/swarrot-bundle": "~1.5",
    "symfony/monolog-bundle": "~3.1",
    "guzzlehttp/guzzle-services": "^0.5.0|~1.1",
    "csa/guzzle-bundle": "^1.3",
    "graylog2/gelf-php": "~1.5",
    "odolbeau/rabbit-mq-admin-toolkit": "^4.0"
}
```

CONTRÔLE DES MESSAGES

Analogie avec un contrôleur aérien



RÉESSAIE PLUS TARD

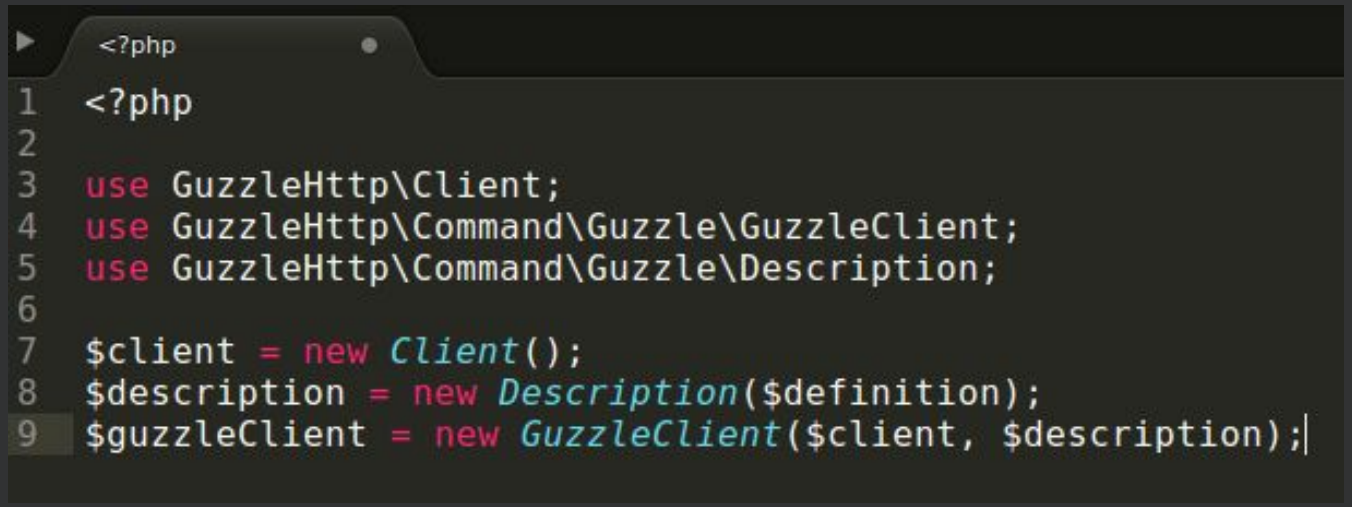


COLLECTER - GUZZLE SERVICE

```
parameters:
     salesforce_definition:
     baseUrl: https://cs83.salesforce.com/
     operations:
          updateAccount:
               httpMethod: PATCH
               uri: 'sobjects/Account/{accountId}'
               responseModel: getResponse
               parameters:
                    accountId:
                    type: string
                    location: uri
                    required: true
                    Name:
                    type: string
                    location: json
```

Client configué simplement en YAML et facilement lisible.

COLLECTER - GUZZLE SERVICE



COLLECTER - USAGE

```
$this->salesforceClient
  ->updateAccount(
    [
      'accountId' => 'gg-11',
      'Name' => 'Eleven-Labs',
    ]
  );
```

CORRESPONDANCE ET TRANSFORMATION

Données CRM

```
{
    "Name": "Eleven Labs",
    "Status__c": "CLIENT_ACTIVE",
    "Country__c": "FR"
}
```

Données attendues par API client

```
{
    "business": {
      "name": "Eleven Labs"
    },
    "status": 70,
    "country": {
      "id": 76
    }
}
```

CORRESPONDANCE ET TRANSFORMATION

```
parameters:
     salesforce_to_emo_mapping:
     business:
          reverse_data_transformer_ids: [app.transformer.address]
          fields:
               - from: "[Name]"
                 to: "[business][name]"
               - from: "[Status__c]"
                 to: "[business][businessStatus]"
                 reverse_data_transformer_id: app.transformer.business_status
```

Configuration simple en YAML

PROBLÈMES RENCONTRÉS

- Import initial des données
- Performance du driver AMQP
- Retry et dépendance aux services externes
- Gestion des événements du CRM

CONCLUSION

- Microservice indispensable dans un SI
- Facile à surveiller et à déployer
- Facilement maintenable et scalable
- Déclinable pour d’autre problématique asynchrone
- Déjà mis en oeuvre chez France Télévision et chez ETSGlobal