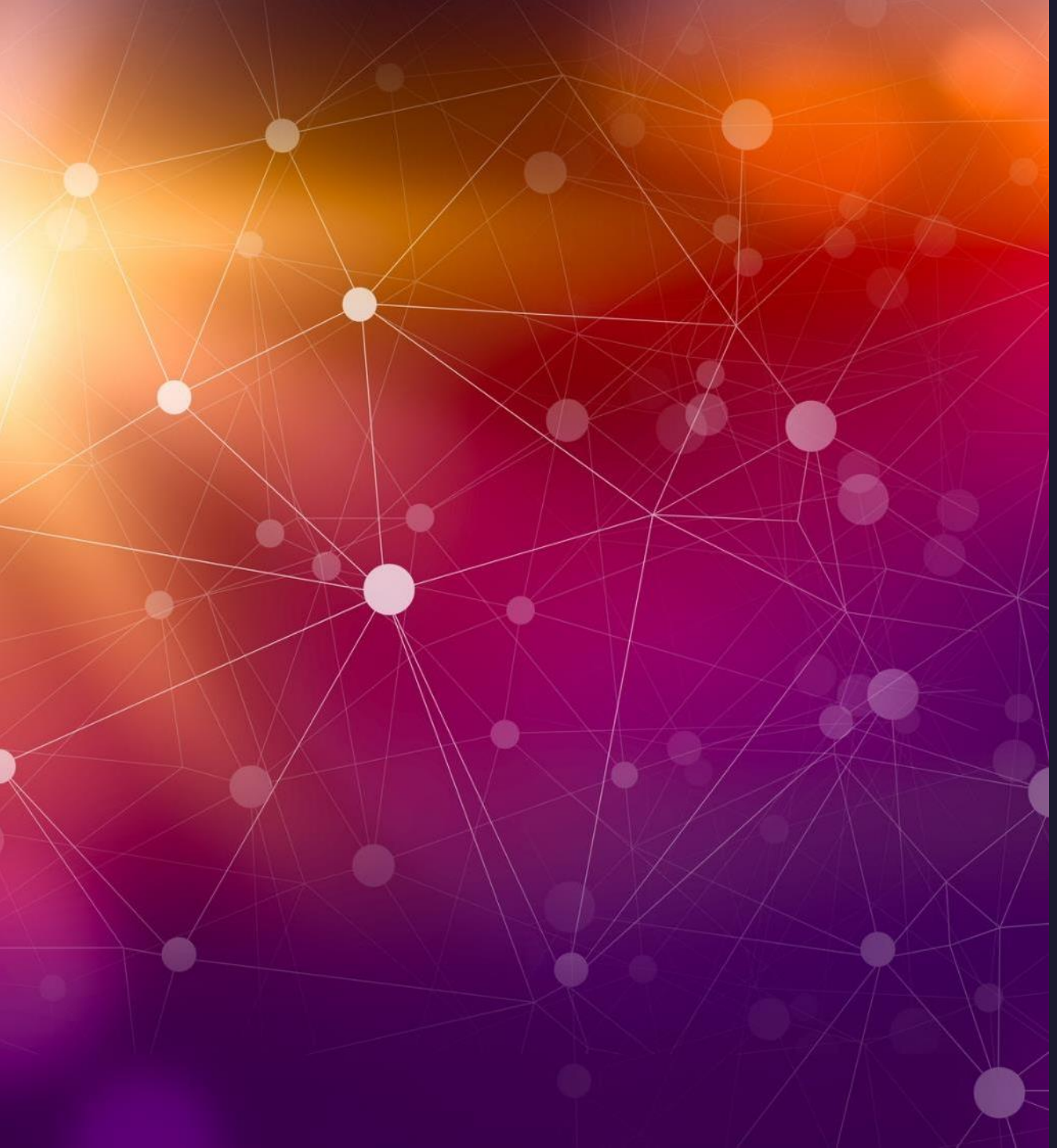


EventHub – Git et docker

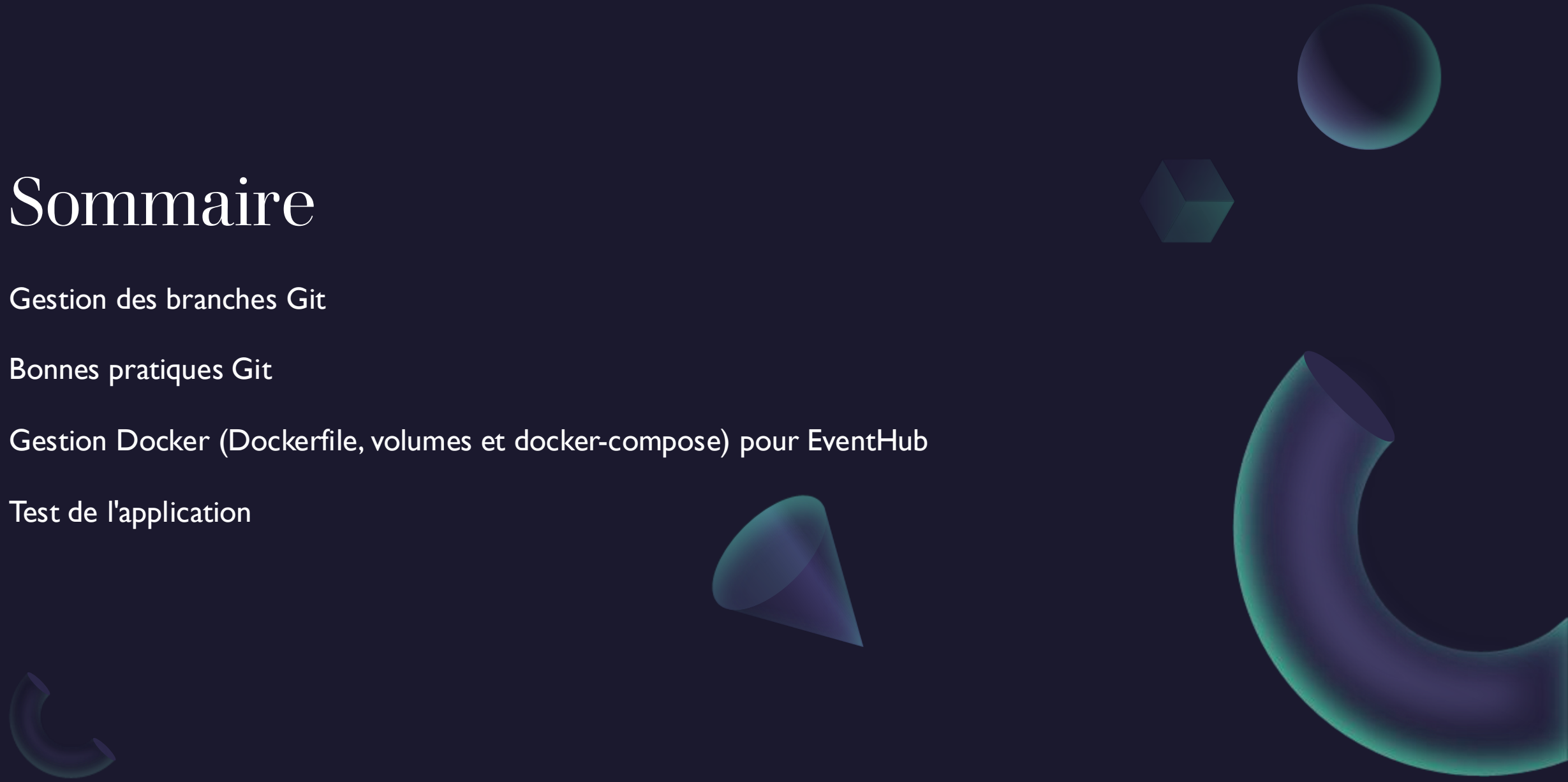
Sommaire

Gestion des branches Git

Bonnes pratiques Git

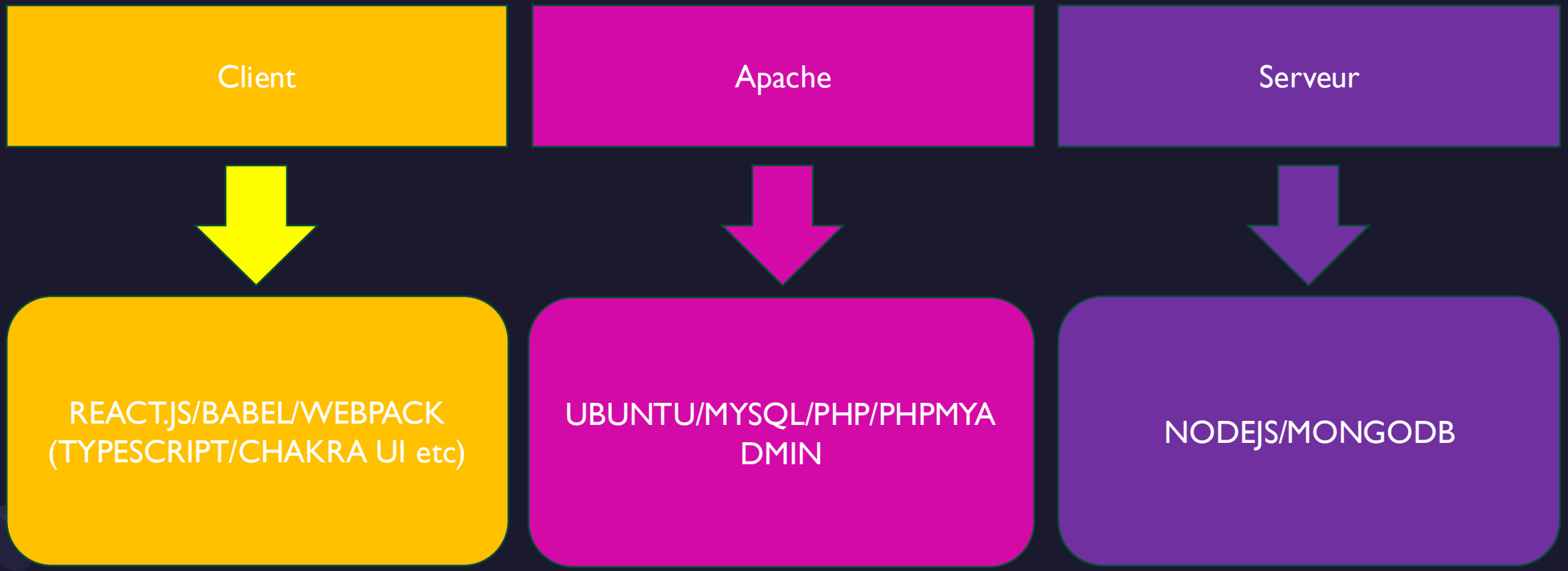
Gestion Docker (Dockerfile, volumes et docker-compose) pour EventHub

Test de l'application



Gestion Git

Gestion Git


Client
Apache
Serveur
REACT.JS/BABEL/WEBPACK
(TYPESCRIPT/CHAKRA UI etc)
UBUNTU/MYSQL/PHP/PHPMYA
DMIN
NODEJS/MONGODB

Gestion des branches Git

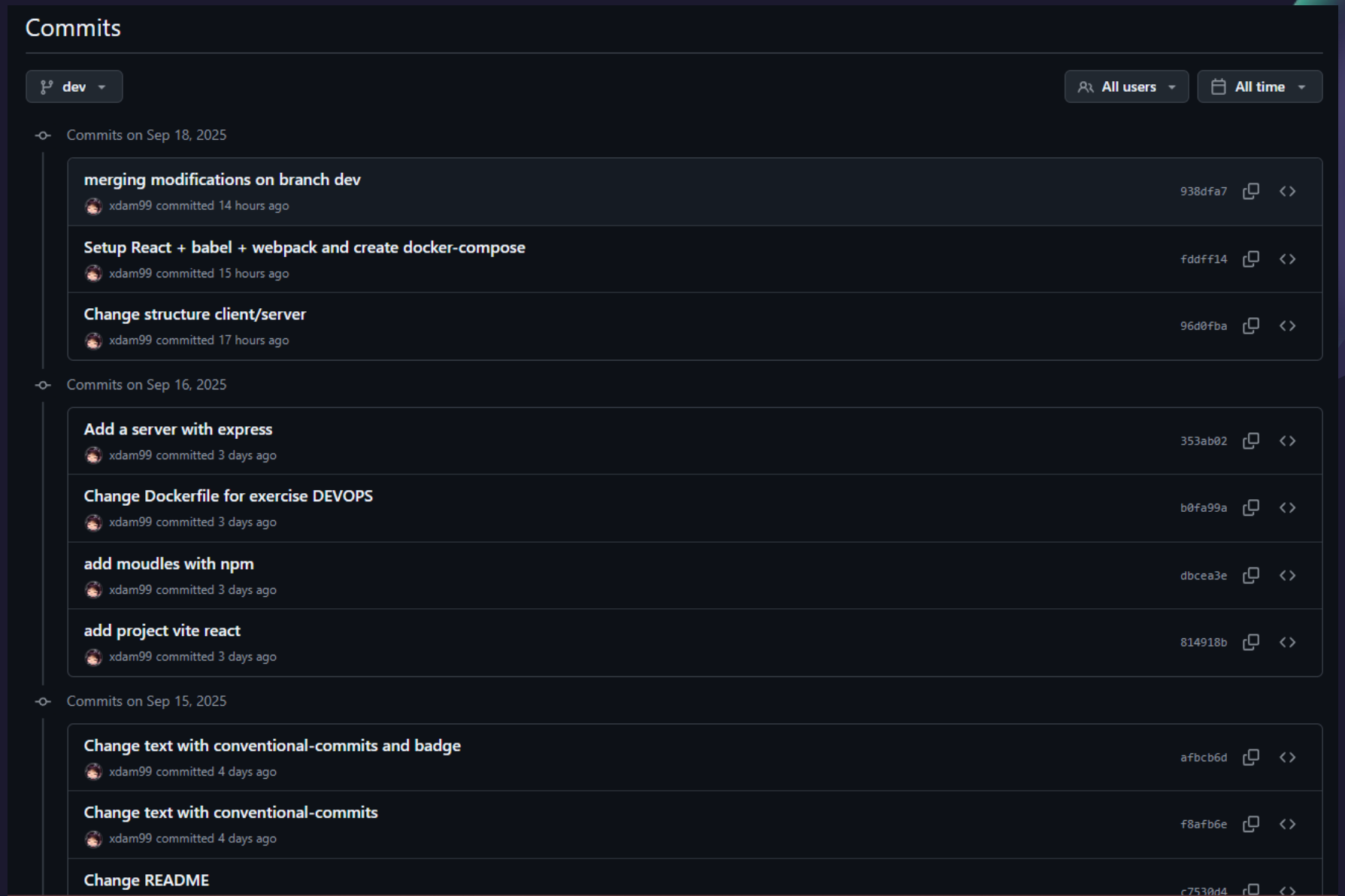
Workflow

Voici comment va se faire ma manière de travailler selon la méthode workflow :

Créations des branches :

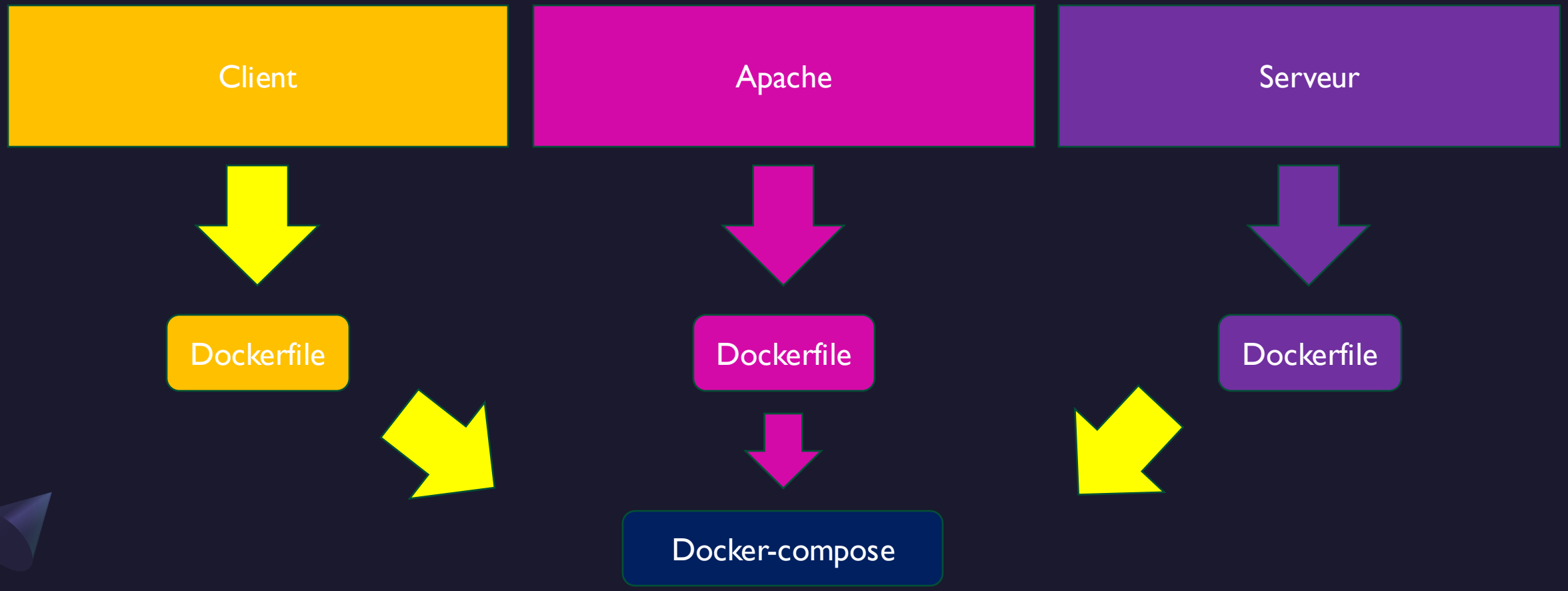
- Main ->
  - Dev ->
    - Front ->
      - Front-hotfixe
    - Back ->
      - Back-hotfixe

Bonnes pratiques Git



Gestion Docker

Gestion des dockerfiles



Pour le client react (même chose pour server node mais exposé sur le port 5000)

event-hub > client > Dockerfile

You, 15 hours ago | 1 author (You)

```
1  FROM node:18    The image contains 2 critical and 19 high vulnerabilities
2        You, 15 hours ago • Setup React + babel + webpack and create docker…
3  WORKDIR /app
4
5  COPY package.json package-lock.json ./
6
7  RUN npm install
8
9  COPY . .
10
11 EXPOSE 3000
12
13 CMD ["npm", "start"]
14
```

Pour le server Apache sur Ubuntu

event-hub > php > Dockerfile

You, 15 hours ago | 1 author (You)

```
1  FROM ubuntu:latest (last pushed 2 days ago)
2
3  ENV DEBIAN_FRONTEND=noninteractive
4
5  RUN apt-get update && apt-get install -y \
6      apache2 \
7      php \
8      libapache2-mod-php \
9      php-mysql \
10     php-gd \
11     php-xml \
12     php-mbstring \
13     php-zip \
14     php-curl \
15     && apt-get clean && rm /var/www/html/index.html
16
17 WORKDIR /var/www/html
18
19 COPY . /var/www/html
20
21 RUN chown -R www-data:www-data /var/www/html
22
23 RUN a2enmod rewrite
24
25 EXPOSE 80
26
27 CMD ["apachectl", "-D", "FOREGROUND"]
28
```

Les conteneurs créés pour le bon fonctionnement :

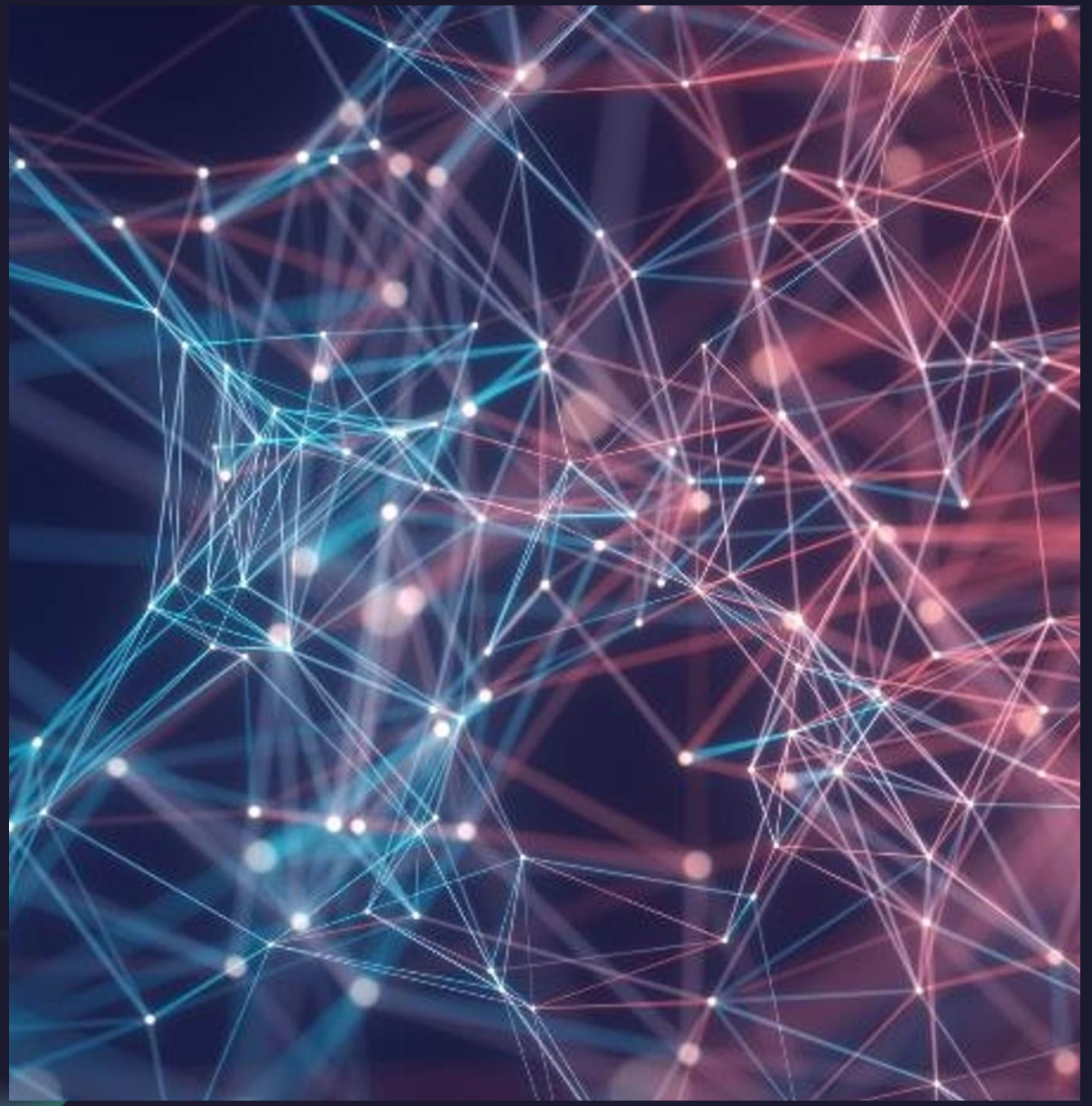
5 services :

- Client
- Server
- Mysql
- Mongo
- Phpmyadmin

3 networks :

- Frontend
- Backend
- Phpmyadmin-network

2 volumes (pour l'instant) :

- Mysql_data
- Mongo_data

Merci !