

# WIP : Connecter une BDD MongoDB à Metabase en local via Docker

## Metabase, qu'est-ce que c'est ?

Metabase est un outil de BI (Business Intelligence) open source qui nous permet de visualiser nos données grâce à des créations de Dashboard (ou autre) simplement. On peut se connecter avec différentes sources de données (PostgreSQL, MongoDB, voir CSV). Nous avons déjà connecter la BDD Postgre Compta à Metabase.

## Objectif :

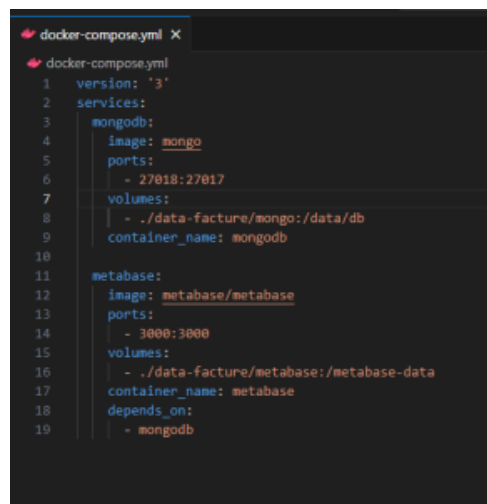
Connecter une BDD MongoDB à Metabase via Docker.

## Installation de Metabase et mongoDB via Docker :

Allez dans le projet *monaliasa-tools* et aller sur le dossier '*metabase*' (nom du dossier + emplacement du code temporaire)

Le dossier est d'un fichier : *docker-compose.yml* et d'un dossier: *data-facture* (le nom peut changer)

## docker-compose.yml



```
1 version: '3'
2 services:
3   mongodb:
4     image: mongo
5     ports:
6       - 27018:27017
7     volumes:
8       - ./data-facture/mongo:/data/db
9     container_name: mongodb
10
11  metabase:
12    image: metabase/metabase
13    ports:
14      - 3000:3000
15    volumes:
16      - ./data-facture/metabase:/metabase-data
17    container_name: metabase
18    depends_on:
19      - mongodb
```

Le fichier se compose de plusieurs couches et est composé de deux images : *mongodb* et *metabase*

mongodb :

le service nommée mongodb est composé de plusieurs couches :

- *image : mongo* => signifie qu'on va pull l'image la plus récente sur docker-hub.
- *ports : 27018:27017* avec à gauche le port de notre machine et à droite le port du container, ils ne sont pas pareils car 27017 sur notre poste fait également référence à notre serveur mongodb lancé en local donc pour éviter des surprises j'ai préféré le mettre sur un autre port.
- *volumes : ./data-facture/mongo:/data/db* avec à gauche le dossier présent sur notre machine et à droite le dossier présent dans le container nous permet de 'bind' et donc de garder les données lorsque le container s'arrête. Ici on garde en mémoire les collections mongodb.
- *container\_name: mongodb* => nous permet de donner le nom qu'on souhaite au container ( plus de lisibilité)

metabase :

- *image: metabase/metabase* signifie qu'on va pull l'image la plus récente sur docker-hub.

- *ports*: 3000:3000 pour avoir accès sur notre machine depuis le navigateur on peut faire : 'localhost:3000'
- *volumes*: ./data-facture/metabase:/metabase-data => nous permet de garder les données lorsque le container s'arrête
- *container\_name*: metabase pour plus de lisibilité
- *depends\_on* : mongodb => signifie qu'on attend que mongodb finisse de se lancer avant de run cette image pour permettre le chargement des données




## Lancer le docker-compose

Pour lancer ensuite ce fichier il suffit d'être dans le répertoire où se situe le fichier *yaml* et d'entrer la commande : *docker-compose up*

```
PS C:\Users\rpailler\Documents\docker> docker-compose up
[+] Running 13/15
✓ mongodb 8 layers [██████████] 0B/0B Pulled
- metabase 7 layers [██████ ] 10.16MB/43.46MB Pulling
```

Ici les images sont *pulled* depuis docker-hub

On peut vérifier que le container fonctionne en allant sur *Docker Desktop* :

<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	▼  <b>docker</b>		Running (2/2)	2.9%		44 seconds ago	■ : 🗑
<input type="checkbox"/>	 <b>mongodb</b> 64cab4ab538b 🗑	<a href="#">mongo</a>	Running	2.26%	<a href="#">27018:27017</a> 🔗	44 seconds ago	■ : 🗑
<input type="checkbox"/>	 <b>metabase</b> cd445690acd1 🗑	<a href="#">metabase/metabase</a>	Running	0.64%	<a href="#">3000:3000</a> 🔗	44 seconds ago	■ : 🗑

## Connecter une bdd sur Metabase

On peut maintenant utiliser le logiciel *Relationnal Migrator* (voir [doc](#)) et aussi mongoDb Compass (voir [doc](#)) pour ajouter une collection dans notre base de données mongoDB lancé via un container

Au lieu d'aller sur le terminal docker du container *mongodb*, on peut aller sur *mongoDb Compass* et faire new Connection avec le bon port (ici 27018) :

## New Connection

Connect to a MongoDB deployment

FAVORITE

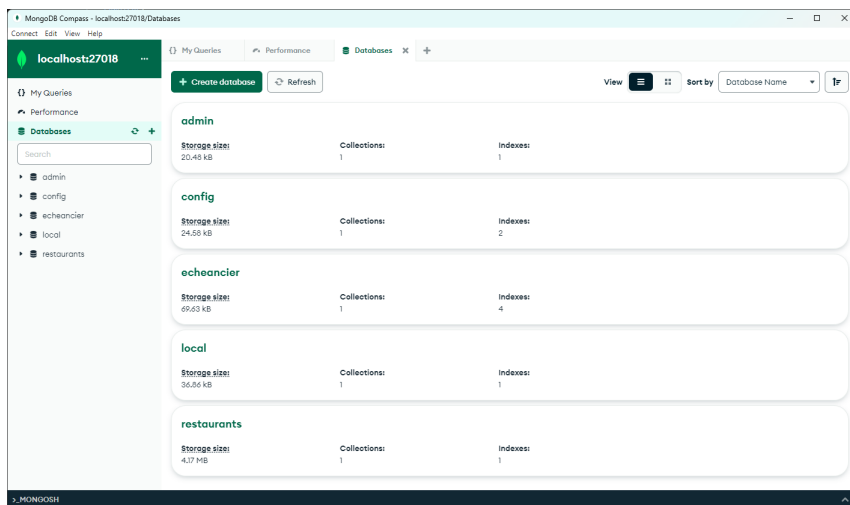
URI ⓘ Edit Connection String ☐

mongodb://localhost:27018/

> Advanced Connection Options

Save Save & Connect Connect

Vous aurez ensuite la liste de vos bdd, exemple :



Database	Storage size	Collections	Indexes
admin	20.48 KB	1	1
config	24.55 KB	1	2
echecancier	65.63 KB	1	4
local	35.86 KB	1	1
restaurants	4.17 MB	1	1

Pour ensuite ajouter une nouvelle collection migrée depuis notre base *postgres*, modifier uniquement l'url de connexion avec le bon port (ici 27018) et faites *start*

## New Migration Job

1 Connect Source DB 2 Connect Destination DB 3 Migration Options

Connect to the MongoDB destination database by entering the connection details.

Select a database Enter MongoDB URI

MongoDB connection string (URI) Documentation

mongodb://localhost:27018

Database echecancier Optional

Username Optional

Password Optional

☒ Save password for this session ⓘ

Back Cancel Test connection Connect

Lorsque la migration est 'Completed', allez sur *mongoDb Compass* et cliquez sur le symbole de refresh pour retrouver notre collection migrée :



## Metabase

Lançons désormais Metabase, soit via Docker Desktop en cliquant sur le lien bleu dans la colonne *ports* :

<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	docker		Running (2/2)	0.78%		13 minutes ago	
<input type="checkbox"/>	mongodb 64cab4ab538b	<a href="#">mongo</a>	Running	0.38%	<a href="#">27018:27017</a>	13 minutes ago	
<input type="checkbox"/>	metabase cd445690acd1	<a href="#">metabase/metabase</a>	Running	0.4%	<a href="#">3000:3000</a>	<a href="#">http://localhost:3000</a>	

- Cliquez sur '*C'est parti !*'
- Choisissez '*Français*'

2

Comment devons-nous vous appeler ?

Prénoms

Romain

Nom

PAILLER

Adresse électronique

romain.pailier@equasens.com

Nom de l'entreprise ou de l'équipe

Lessa

Créer un mot de passe

\*\*\*\*\*

Confirmez votre mot de passe

\*\*\*\*\*

Suivant

- Remplissez le formulaire :
- Choisissez 'Pas encore sûr' :

1



- 1

1

- Quand on va sur la partie : '*Parcourir les données*', on se retrouve avec notre base de données voulue (ici échancier pour l'exemple) :

