

---

# *Projet de Visualisation de simulation ARENA sur Grafana*

---

**Réalisé par :**

– Ali EL JARTI

**Encadrés par :**

– Mr . KAMACH OUALID

**Année universitaire 2023/2024**

## Tableaux de matière

I.	Architecture de la solution .....	3
II.	Connexion entre Arena et la base de données Microsoft Access.....	4
III.	Liaison ODBC entre Microsoft Access et MySQL.....	6
III.1	La configuration du Driver ODBC .....	6
III.2	Exportation de la base de données Access vers Mysql .....	6
III.3	Liaison de la table Mysql à la base de donnée Access.....	9

## Tableaux de figure :

Figure 1: Diagramme de l'architecture réalisée.....	3
Figure 2: Modél Arena.....	4
Figure 3: Configuration du module ReadWrite .....	5
Figure 4: Configuration du fichier .....	5
Figure 5: Configuration du Recordset.....	5
Figure 6: Configuration du Driver ODBC .....	6
Figure 7: ODBC DSN.....	7
Figure 8: Table crée dans la base de données SQL .....	8
Figure 9: Choix de la table de BDD MYSQL.....	11
Figure 10: Choix du Primary Key.....	11
Figure 11: Table liée .....	12
Figure 12: ajouter nouvelle base de donnees .....	12
Figure 13 : interface d'insertion de base de données.....	13
Figure 14 : création des différents outils de visualisation .....	13
Figure 15 : Dashboard grafana finale.....	14

# I. Architecture de la solution :

L'architecture présentée dans le diagramme illustre un système de visualisation et de suivi des données de simulation en utilisant plusieurs technologies et outils intégrés.

1. **Modélisation avec Arena:** Tout commence avec le modèle de simulation créé dans Arena, un logiciel de simulation de systèmes complexes.
2. **Base de données Access:** Les données générées par le modèle Arena sont stockées dans une base de données Microsoft Access (MDB). Cela permet de centraliser les données avant de les exporter vers d'autres systèmes.
3. **Exportation via ODBC vers MySQL:** Les données sont ensuite transférées de la base de données Access vers une base de données MySQL. Ce processus utilise une liaison ODBC (Open Database Connectivity), un standard pour accéder aux bases de données.
4. **Exportation continue:** Les données sont exportées de MySQL vers le dashboard Grafana toutes les 5 secondes, permettant une mise à jour quasi instantanée et continue des données visualisées.
5. **Visualisation avec Grafana:** Enfin, Grafana, qui est un outil puissant de visualisation et d'analyse, permet de créer des tableaux de bord dynamiques (dashboards) qui affichent les données de simulation en temps réel. Cela permet aux utilisateurs de suivre les performances et d'analyser divers paramètres de la simulation en cours.

Cette architecture permet non seulement de visualiser les données en temps réel mais aussi d'effectuer une analyse dynamique et détaillée grâce à la puissance de Grafana en matière de visualisation de données.

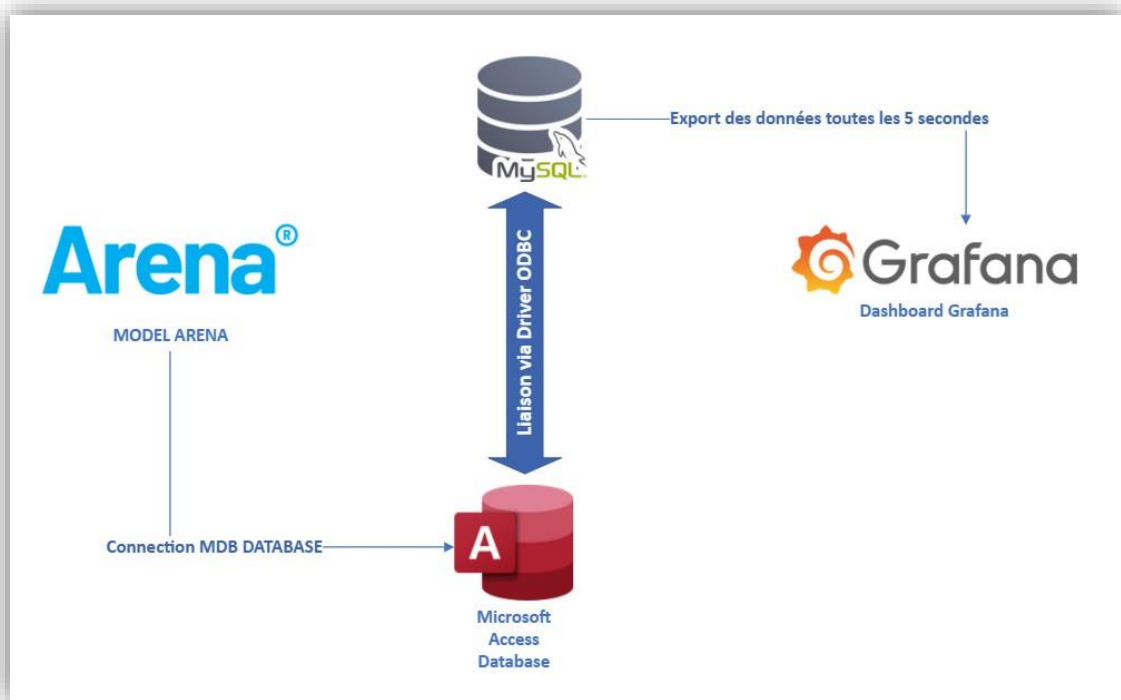


Figure 1: Diagramme de l'architecture réalisée

Dans la suite de notre documentation, nous détaillerons les trois connexions clés que nous avons réalisées dans cette architecture :

1. **Connexion entre Arena et la base de données Microsoft Access** : Nous expliquerons comment les données de simulation sont initialisées et stockées dans Access.
2. **Liaison ODBC entre Microsoft Access et MySQL** : Nous aborderons les spécificités de la configuration ODBC qui permet le transfert des données entre les deux bases de données.
3. **Exportation continue des données vers Grafana** : Nous décrirons le processus d'exportation des données de MySQL à Grafana toutes les 5 secondes, assurant une visualisation en temps réel.

Chacune de ces connexions sera examinée en détail pour comprendre le flux de données et les configurations techniques impliquées.

## II. Connexion entre Arena et la base de données Microsoft Access :

Pour enregistrer les données dans la base de données Microsoft Access depuis Arena, Il faut tout d'abord ouvrir le model Arena que nous avons réalisé :

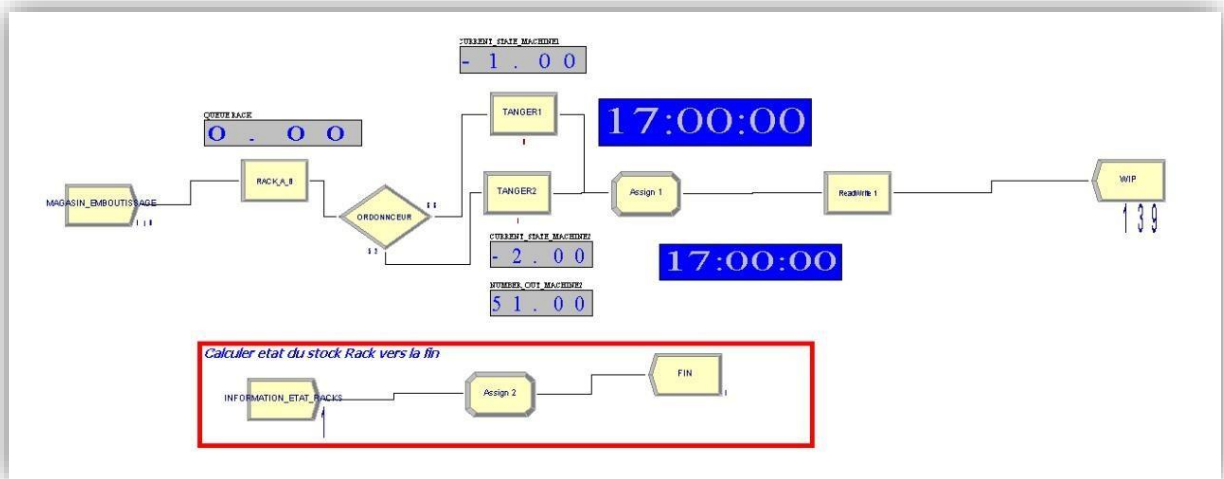


Figure 2: Modél Arena

Ensuite On configure le Block ReadWrite avec le nom de notre fichier et les variable/attribus qu'on veut exporter :

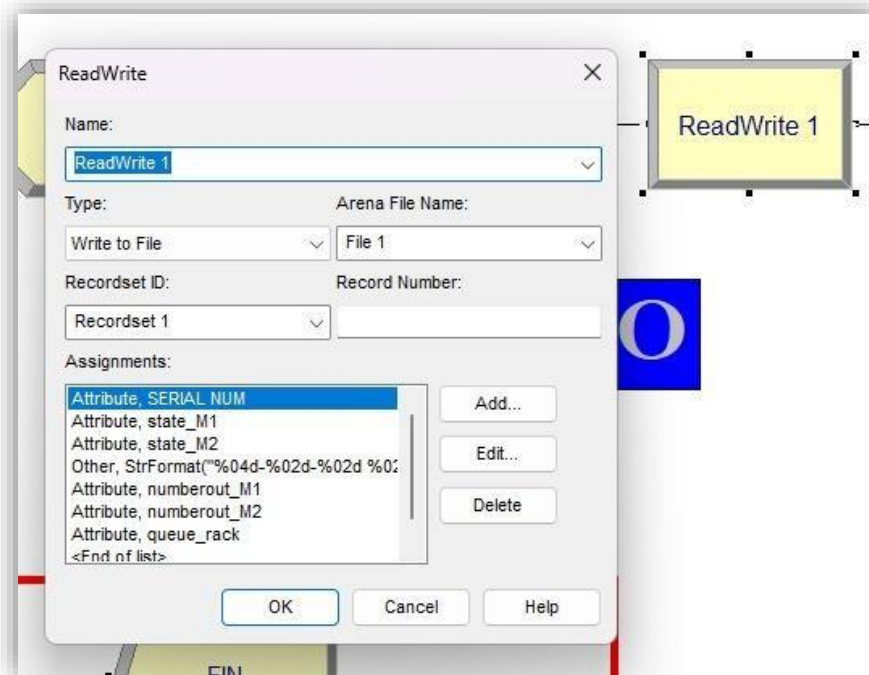


Figure 3: Configuration du module ReadWrite

Ensuite, nous devons configurer le fichier que nous avons utilisé en sélectionnant le type d'accès qui est, dans notre cas, **Microsoft Access 97-2003 (\*.mdb)**, puis charger le chemin du fichier de la base de données **.mdb**.

	Name	Access Type	Operating System File Name	End of File Action	Initialize Option	Recordsets
1	File 1	Microsoft Access 97-2003 (*.mdb)	C:\Users\HPIDocuments\MASTER TIUFS2\gestion industrielle\projet_arena\basededonnees\mydata.mdb	Dispose	Hold	1 rows
	Double-click here to add a new row.					

Figure 4: Configuration du fichier

Ensuite, nous devons saisir dans le **recordset** le nom de la table de notre base de données **Access** dans laquelle nous souhaitons enregistrer les données.

Recordsets		
	Recordset Name	Table Name
1	Recordset 1	arenadata1
Double-click here to add a new row.		

Figure 5: Configuration du Recordset

Après avoir lancé la simulation on remarque que les données sont bien enregistrées dans la table de la base de données Access.

### III. Liaison ODBC entre Microsoft Access et MySQL :

#### III.1 La configuration du Driver ODBC :

Nous avons configuré le pilote ODBC pour établir une connexion entre notre base de données Access et MySQL. Cette étape implique la spécification du serveur local, la sélection du port approprié, et l'entrée des identifiants de l'utilisateur pour sécuriser l'accès à la base de données MySQL.

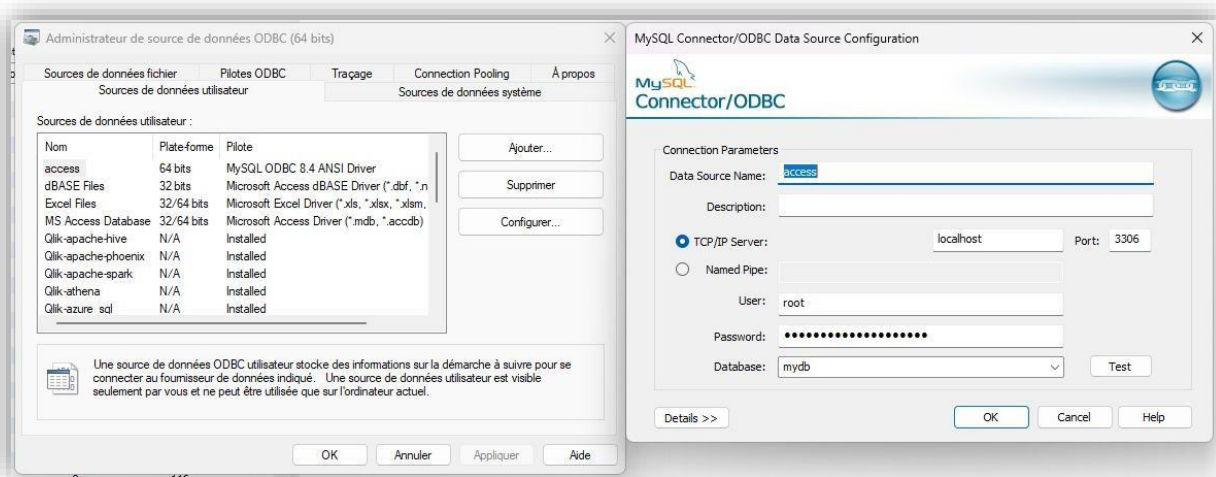
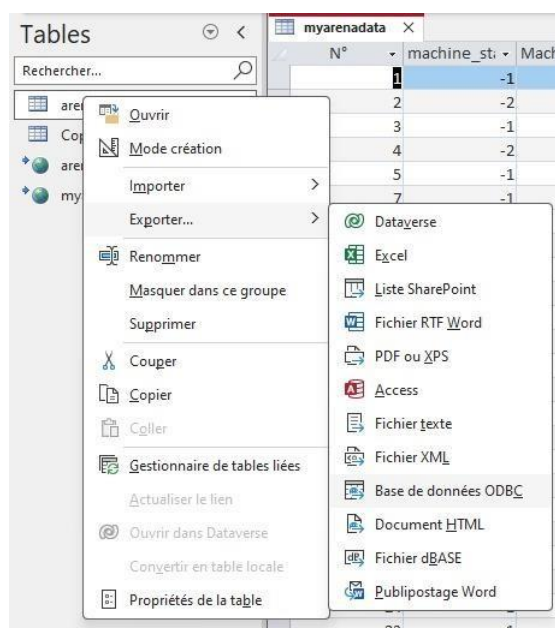


Figure 6: Configuration du Driver ODBC

#### III.2 Exportation de la base de données Access vers Mysql :

Nous devons exporter la base de données Access vers MySQL afin de créer une copie de cette dernière dans MySQL, pour ensuite la lier avec notre base de données Access.

Tout d'abord, nous choisirons l'option « exporter vers une base de données ODBC :



Ensuite on choisi le nom de la base de données qui serait crée dans mysql :



Ensuite nous choisissons la source que nous avons déjà créer :

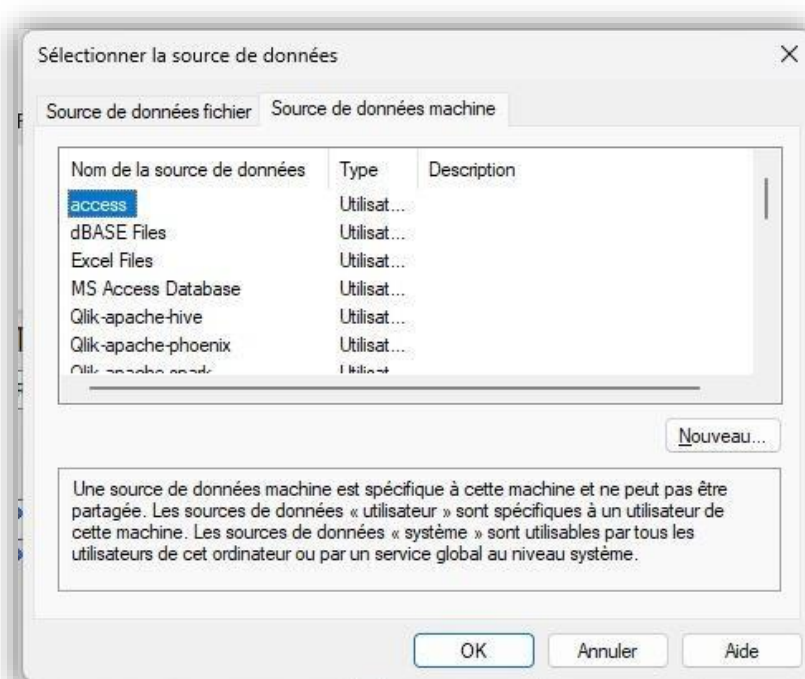
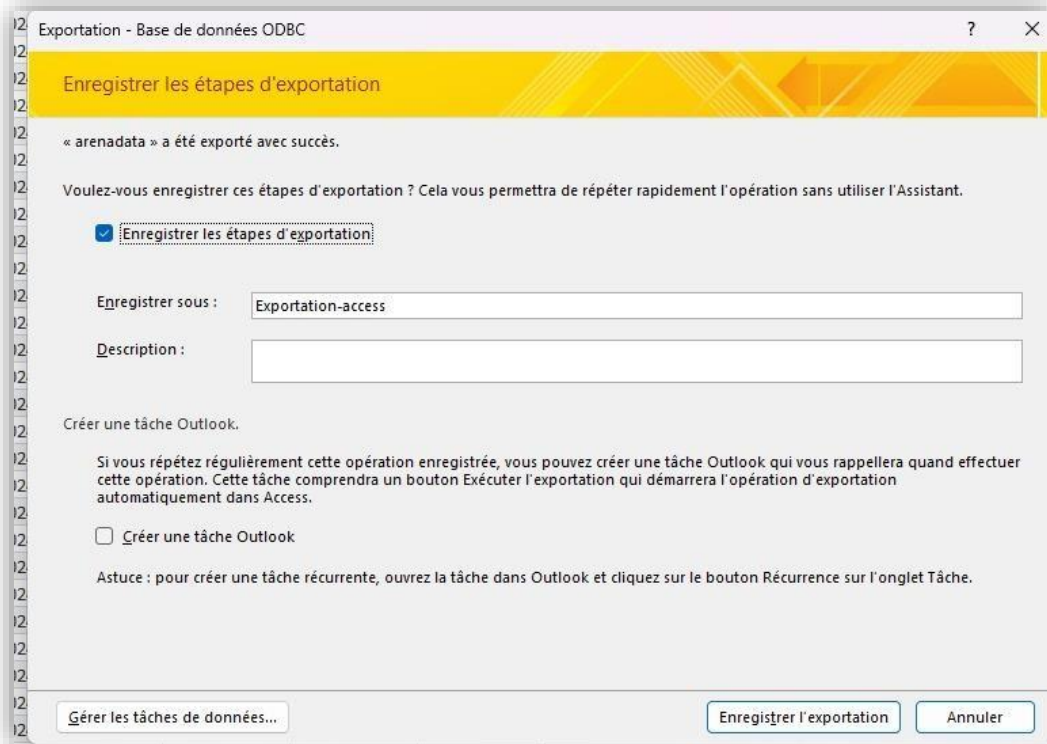


Figure 7: ODBC DSN

Ensuite On clique sue « Enregistrer l'exportation » :





Nous pouvons ensuite voir que la copie de notre table Access a bien été créée dans Mysql :

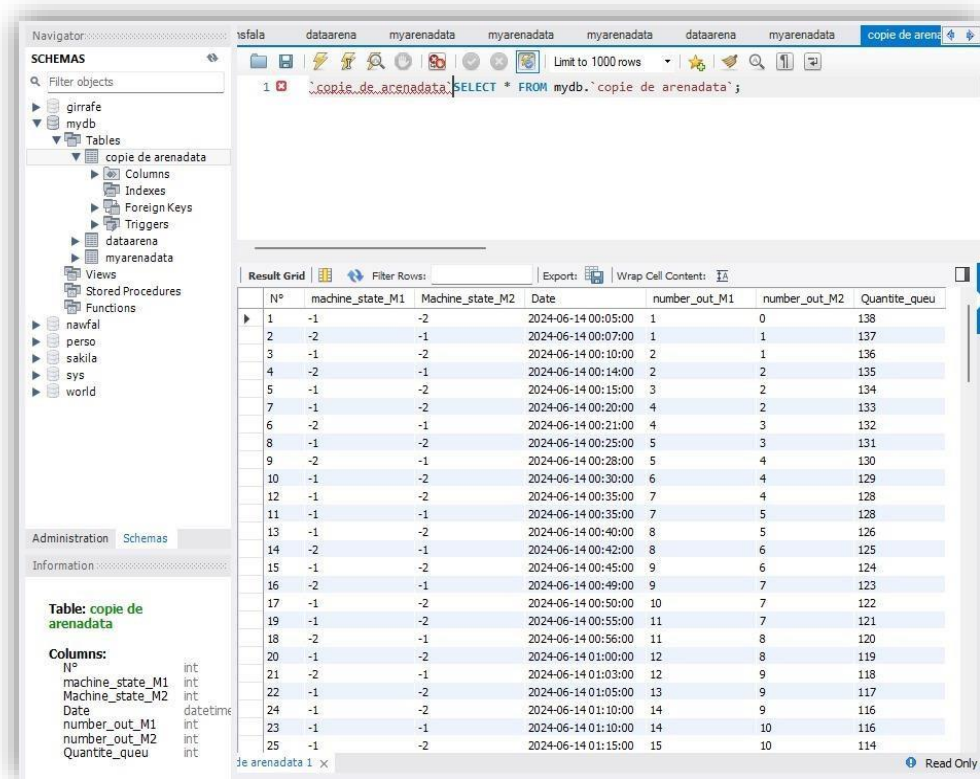
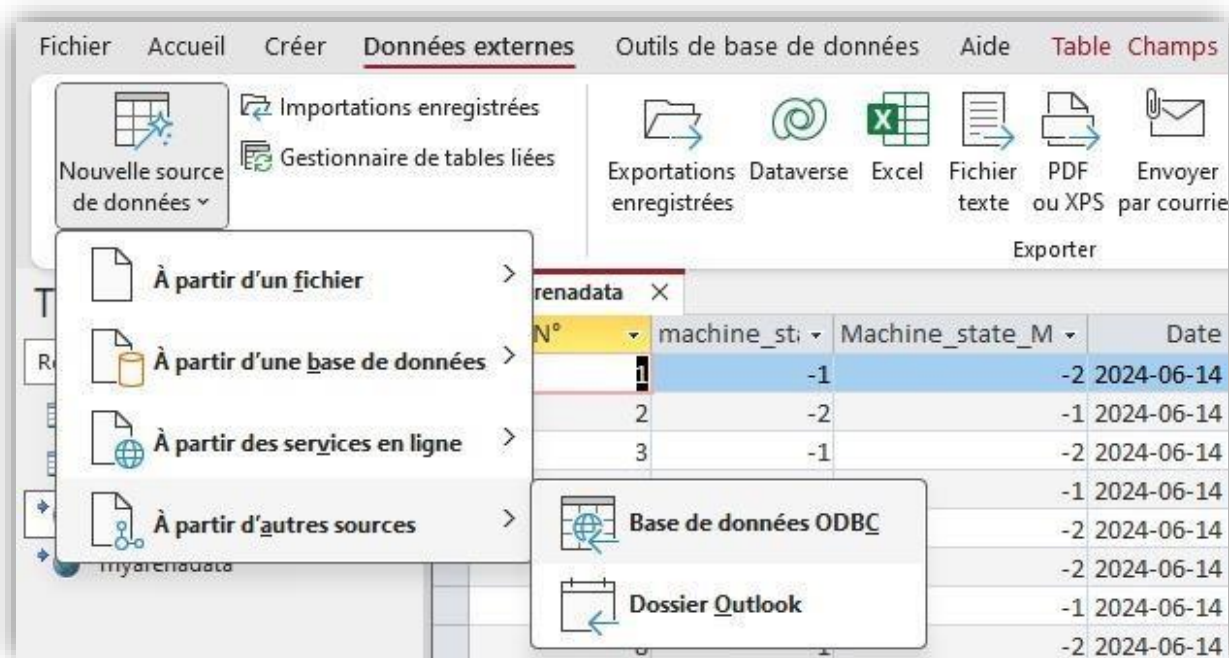


Figure 8: Table créée dans la base de données SQL

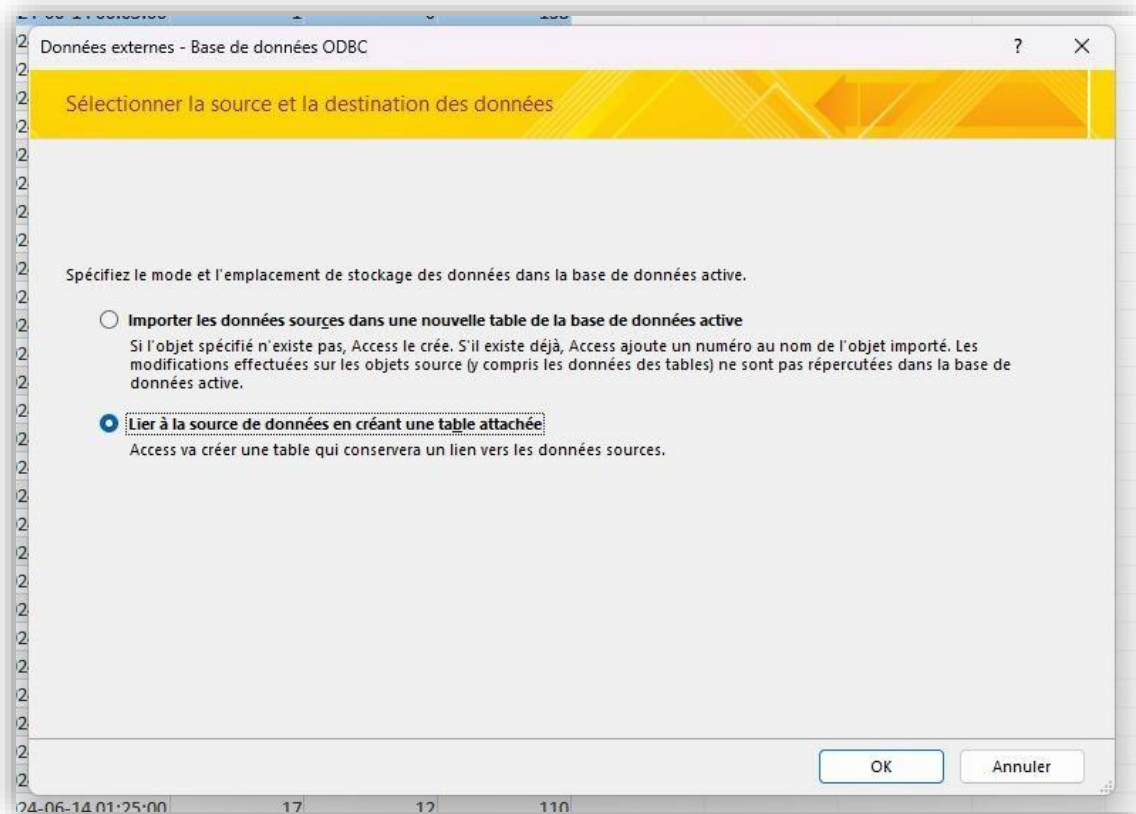


### III.3 Liaison de la table Mysql à la base de donnée Access :

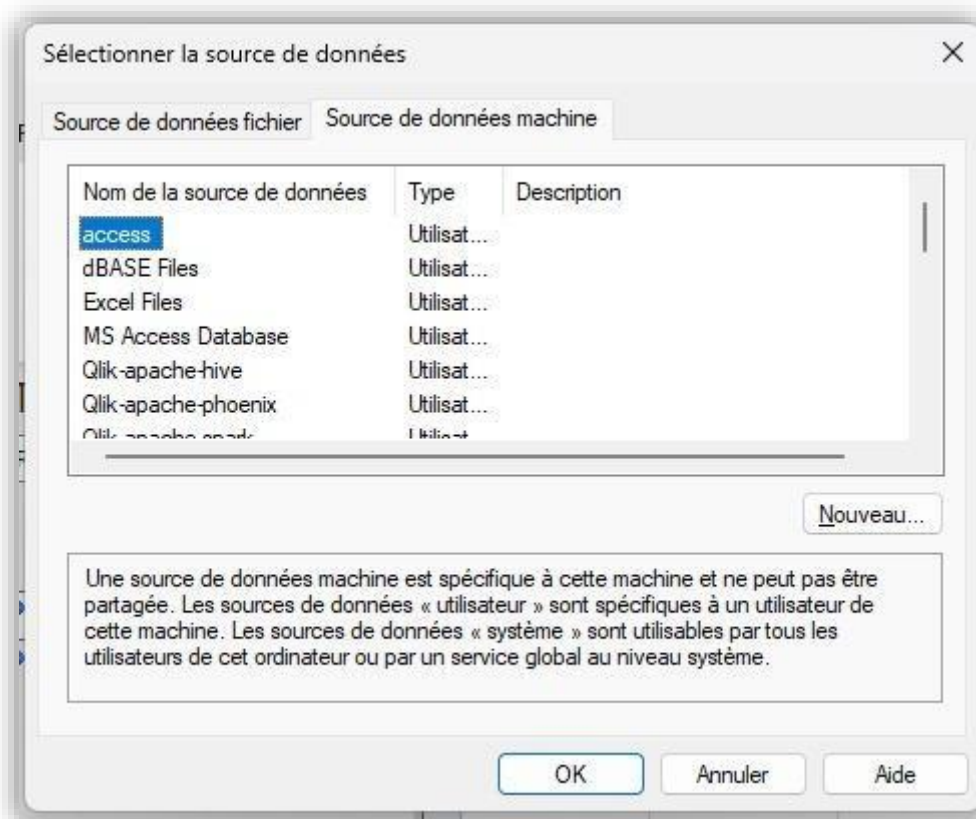
Après avoir créer une copie de notre table Access dans la base de données SQL il faut lier cette table SQL à notre base de données Access. Tout d'abord on clique sur **Données externes** >> **A partir d'autres sources** >> **Base de données ODBC**.



Nous choisissons l'option : **Lier à la source de données en créant une table attachée** :



Ensuite nous choisissons la source que nous avons déjà configuré dans la partie 1) :



Puis nous choisissons la table que nous voulons lié à nître base de données Access :

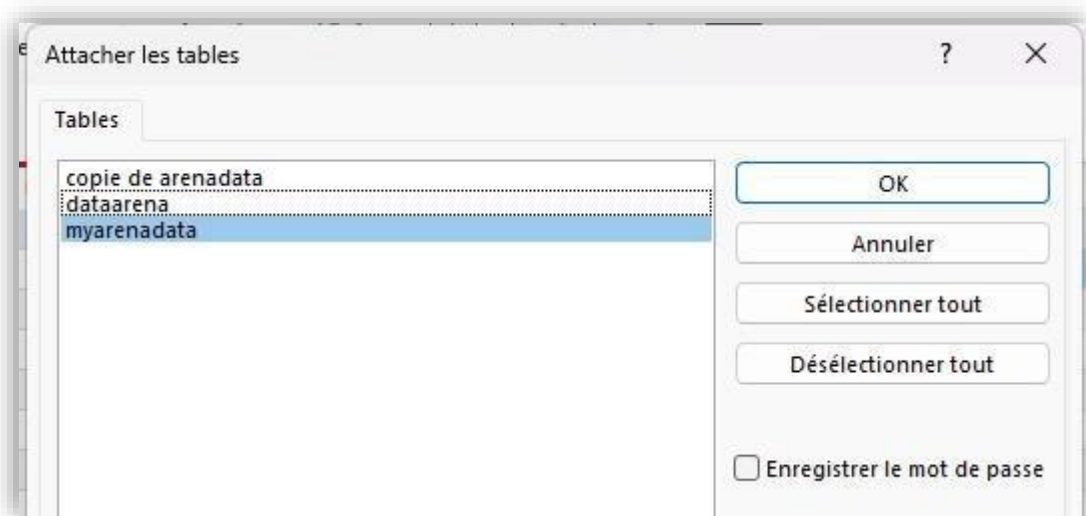


Figure 9: Choix de la table de BDD MYSQL

Nous choisissons le Primary Key de cette Table :

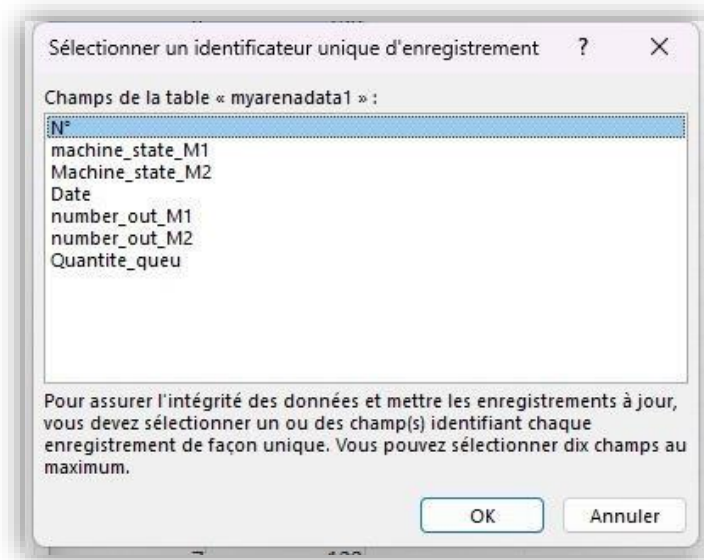


Figure 10: Choix du Primary Key

Nous validons en cliquant sur **OK** :

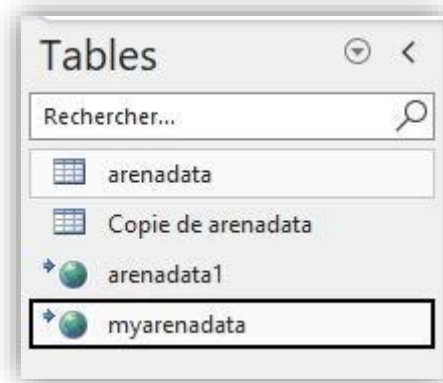


Figure 11: Table liée

Tout d'abord, nous passons à la visualisation du processus sur la plateforme Grafana.

On commence par ajouter la base de données MySQL avec laquelle nous allons nous connecter :

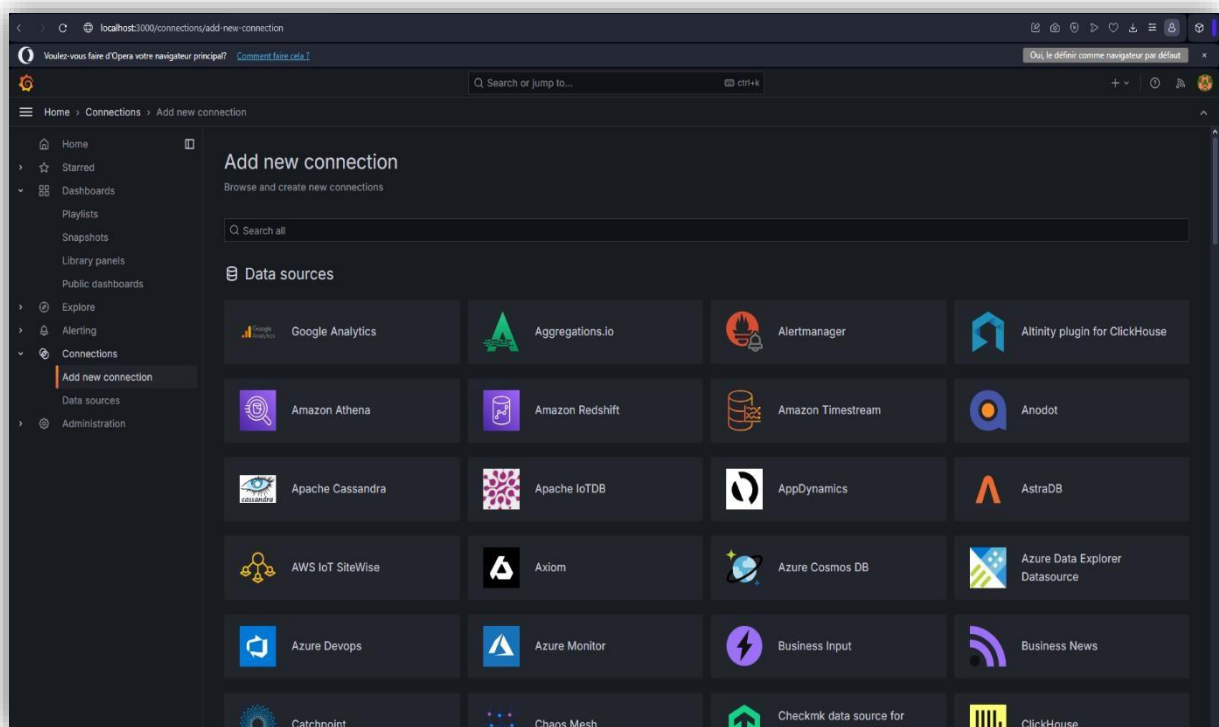


Figure 12: ajouter nouvelle base de données

Ensuite, nous connectons la base de données via MySQL Workbench :



→ Dans la partie création, nous devons sélectionner la base de données, puis la table dans laquelle les données transmises depuis Access ont été enregistrées. Chaque donnée de la table sera affichée à l'aide des différents outils disponibles sur la plateforme Grafana.

Voici la dashboard finale qu'on a créée :



Figure 15 : Dashboard grafana finale

→ Sur notre tableau de bord, nous avons créé différents widgets pour visualiser les états de nos machines Tanger1 et Tanger2, ainsi que le nombre de pièces produites par chaque machine. Toutes ces données sont visualisées en temps réel avec une fréquence de 5 secondes.