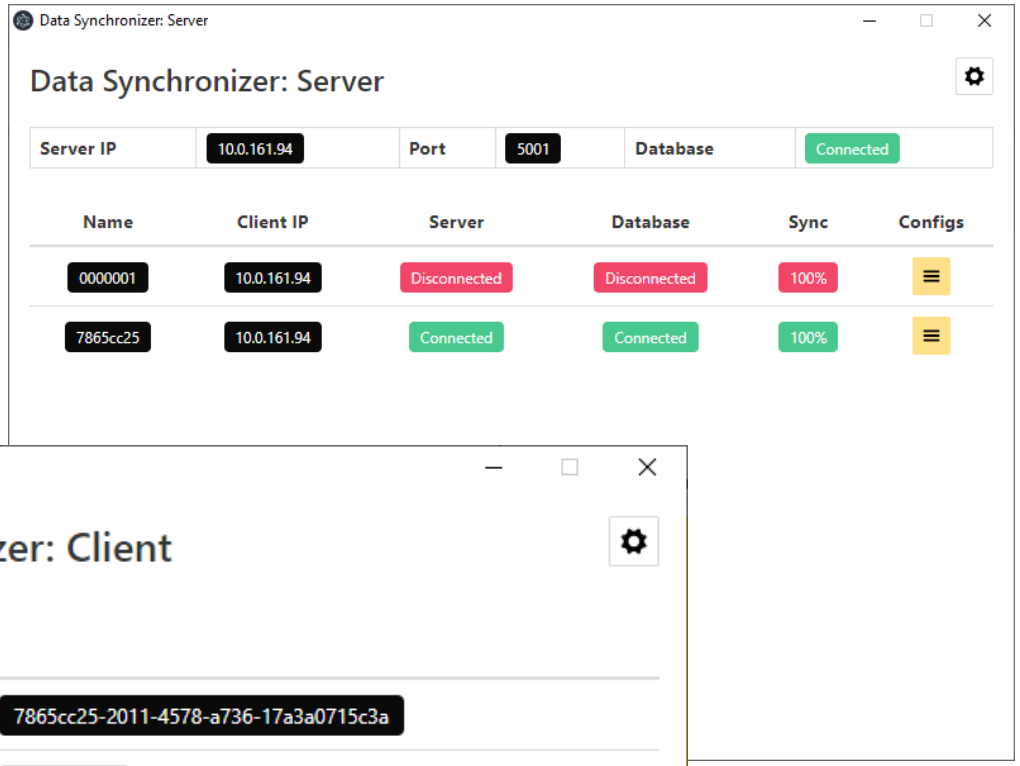
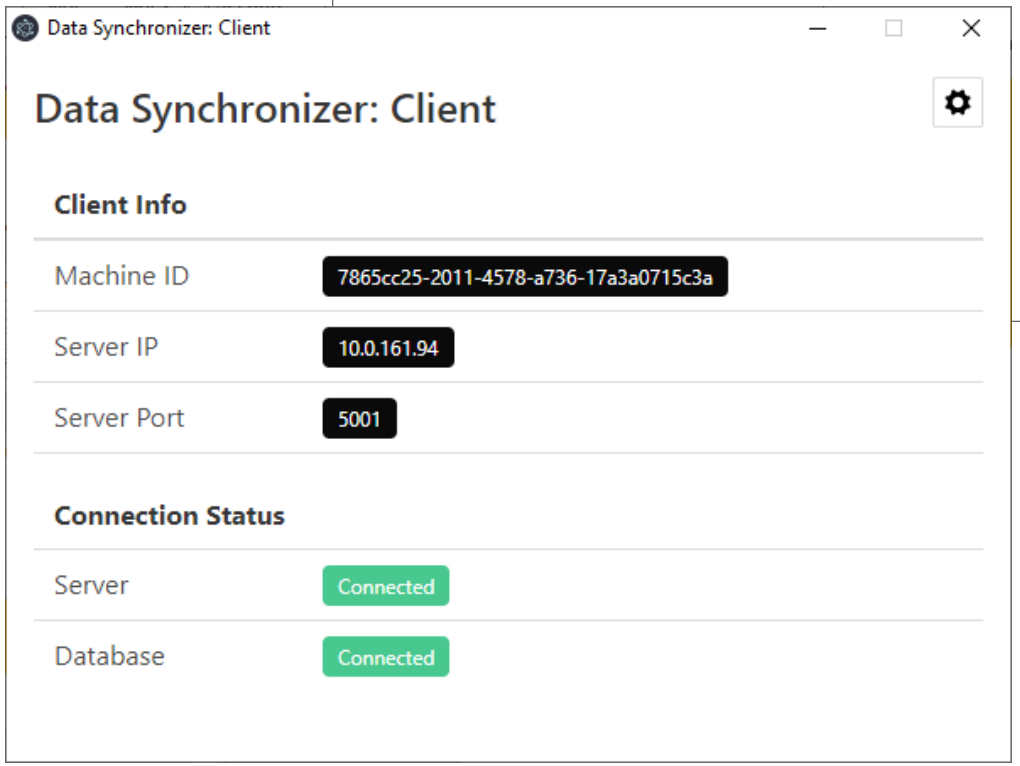
Data Synchronizer

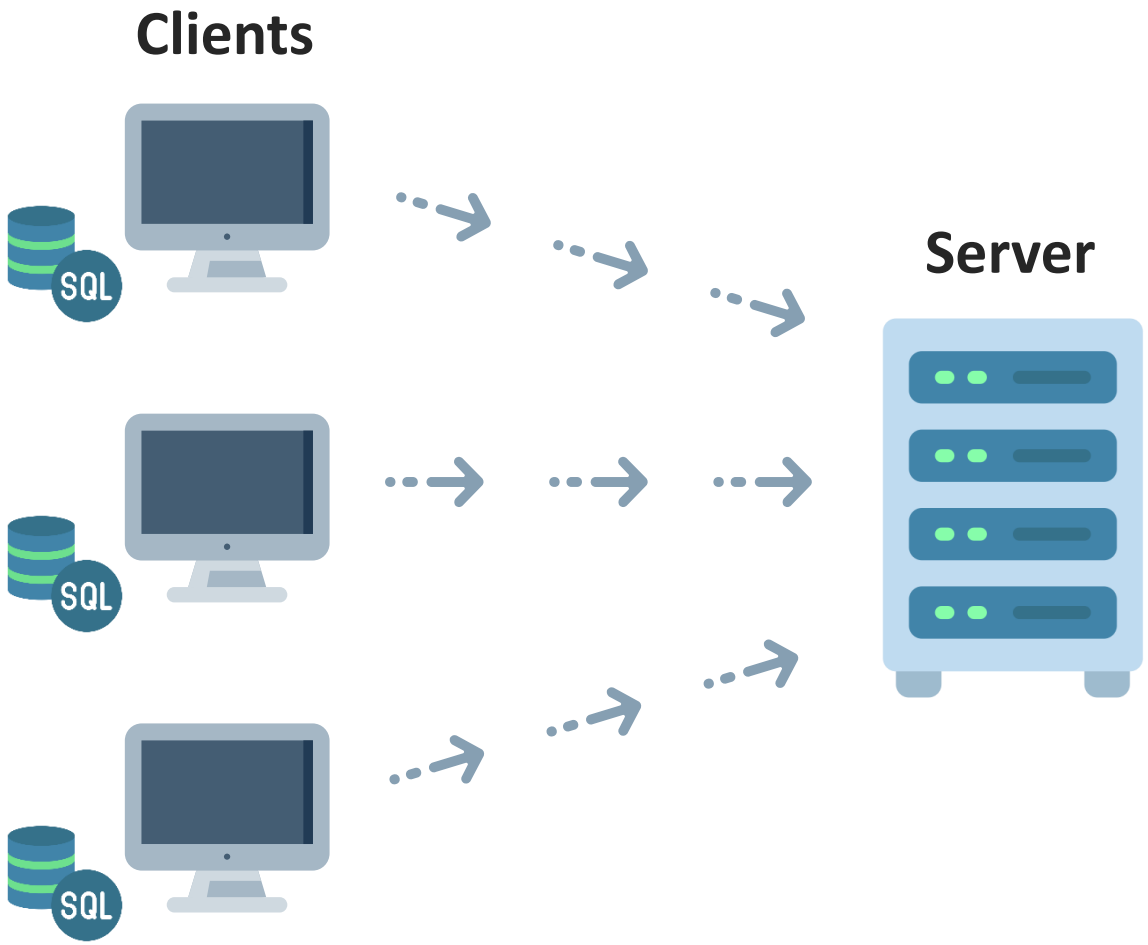




MySQL MariaDB

erman999

Introduction



Data Synchronizer is a Node.js Electron application that synchronizes MySQL/MariaDB data from Client database to chosen Server database.

Applications use Socket.io for connecting and emitting data continuously.

Clients repeatedly try to connect to server if disconnects and continues data transfer from where they left.

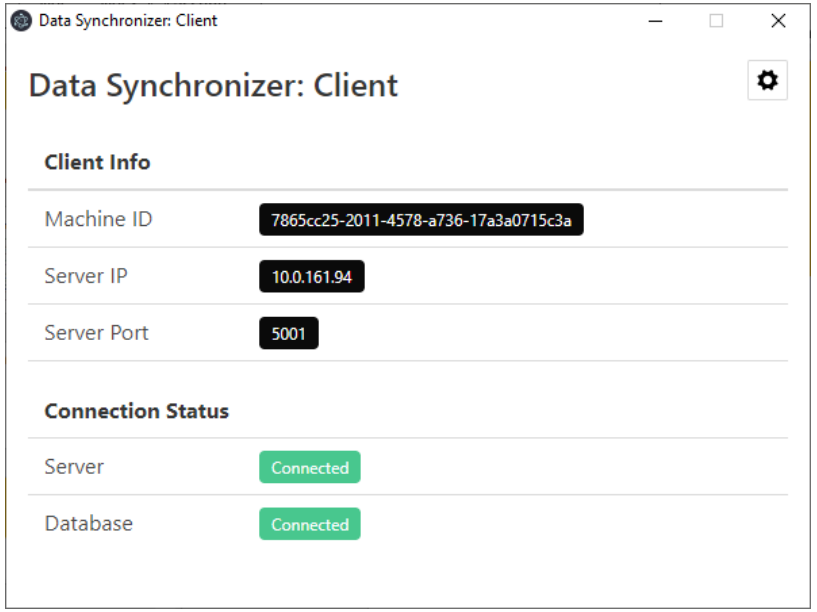
Applications

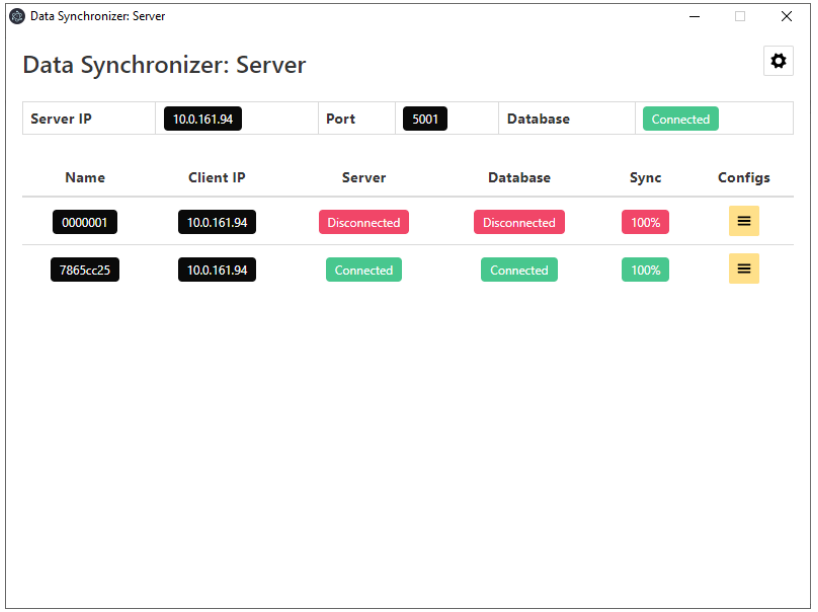
Client

Connects to SQL server, receives requests and sends back to server

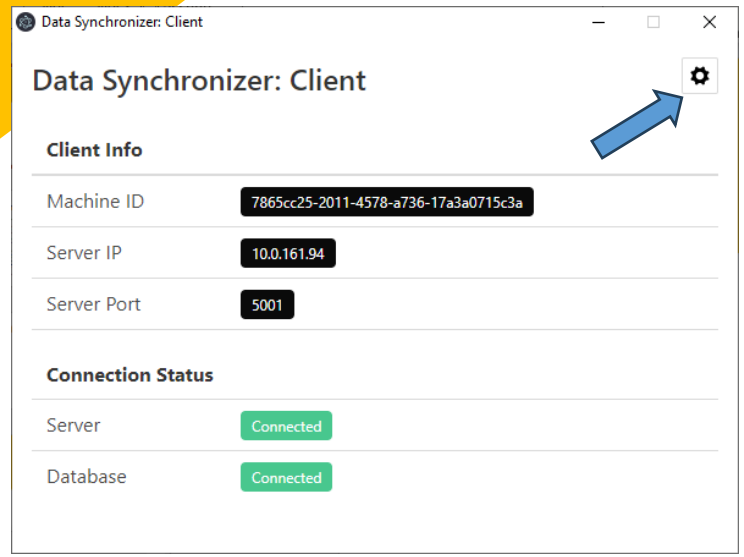
Server

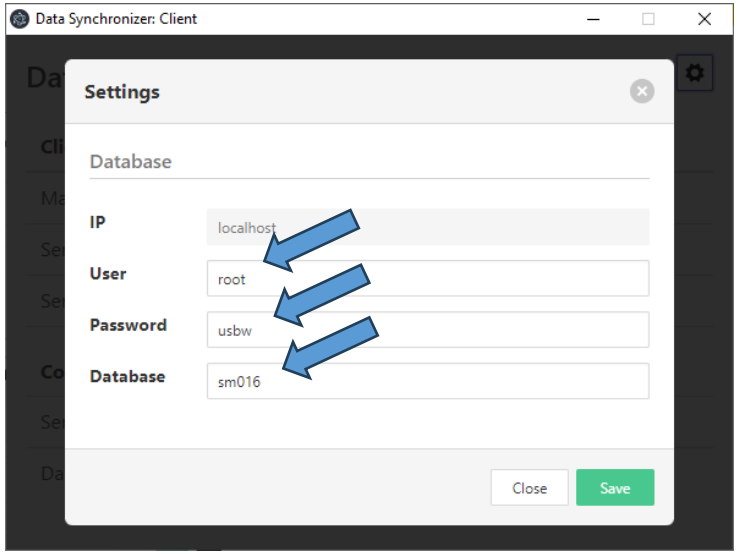
Collects data from clients, synchronize them to target databases

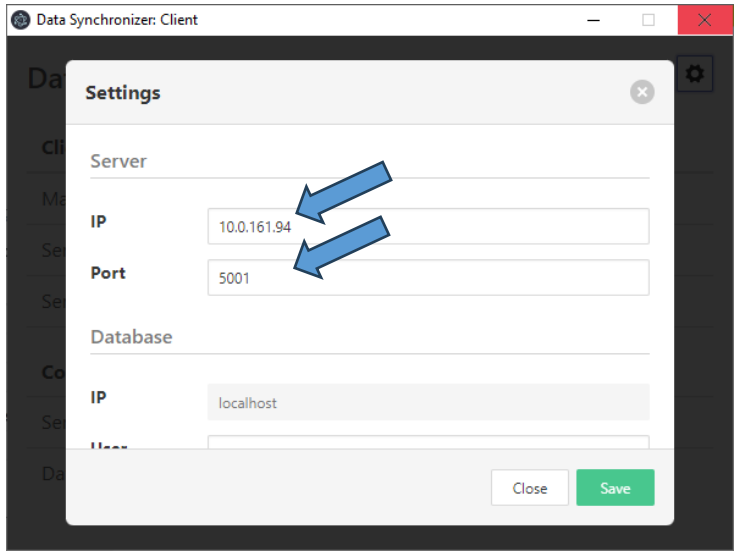


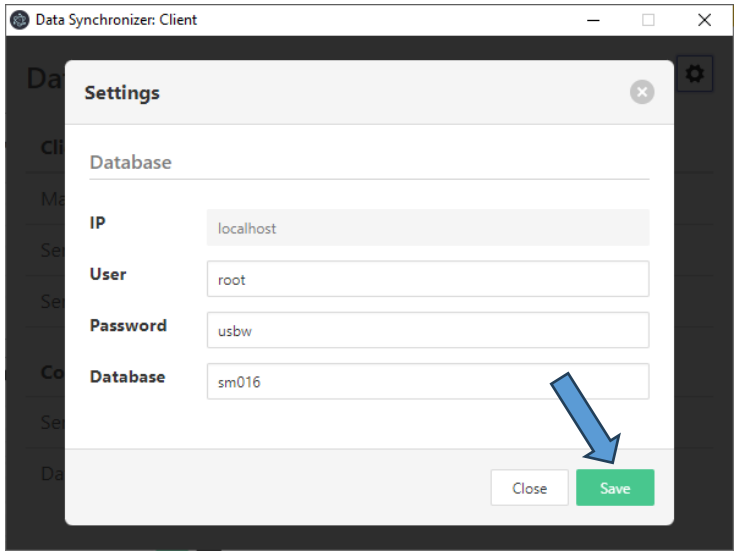


Client Settings



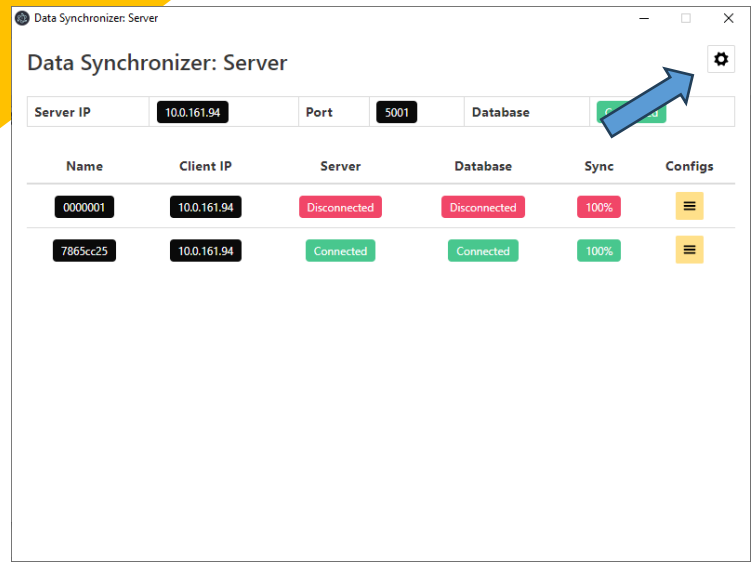


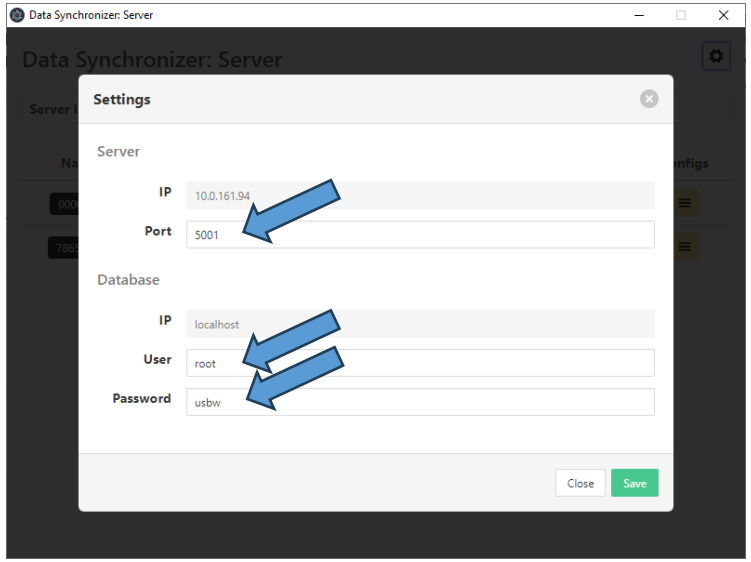


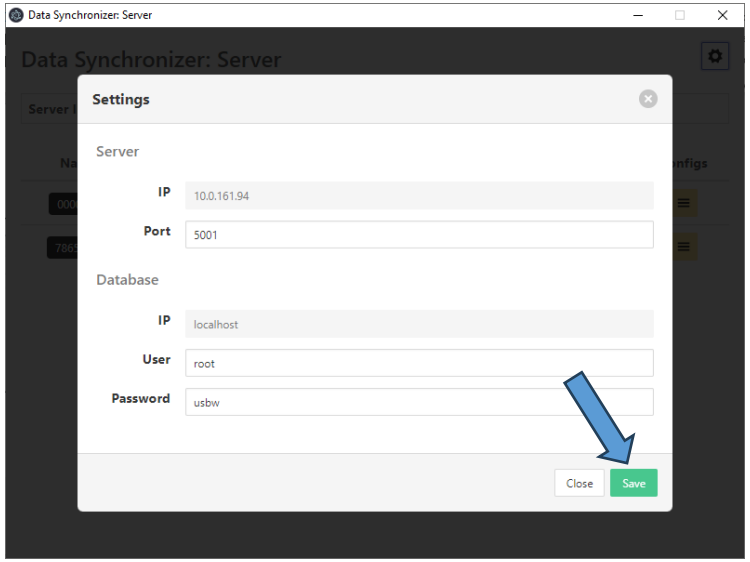


- Click **Cog** button to open settings.
- Enter Server IP address and Port credentials. *(These informations can also be collected from Server application)*
- Enter Database credentials. *(Database name is case-sensitive)*
- Click save button and restart application.
- Settings will be valid after restart.

Server Settings

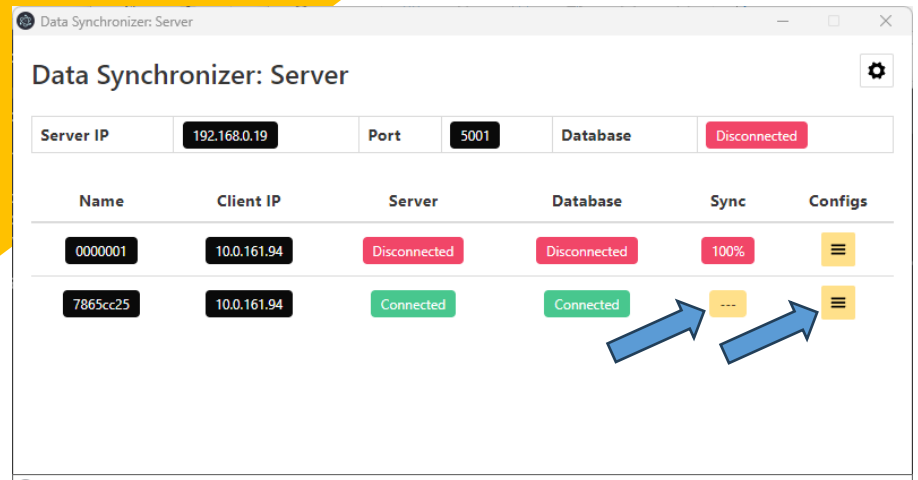


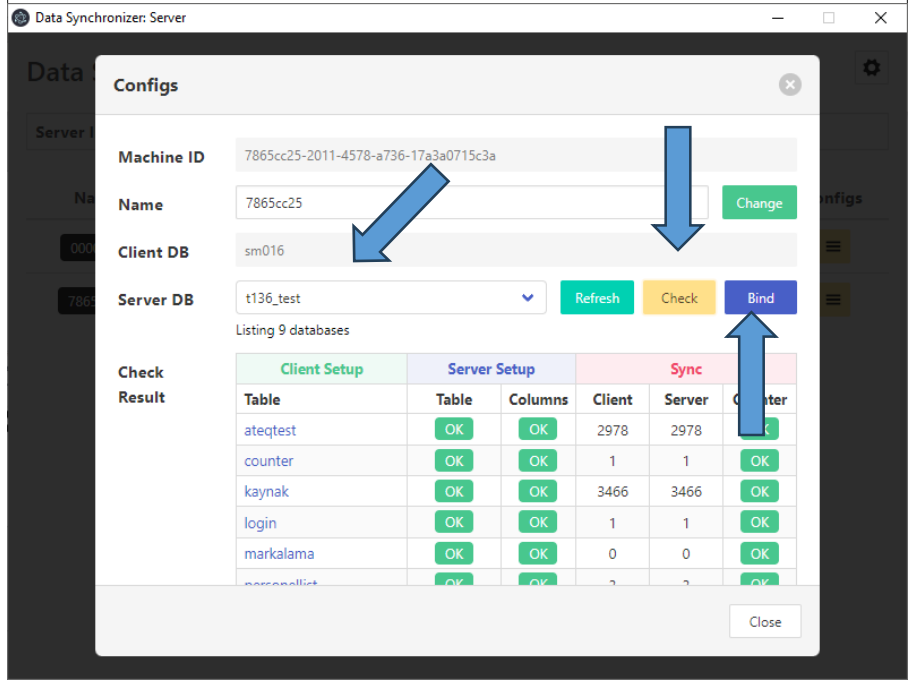


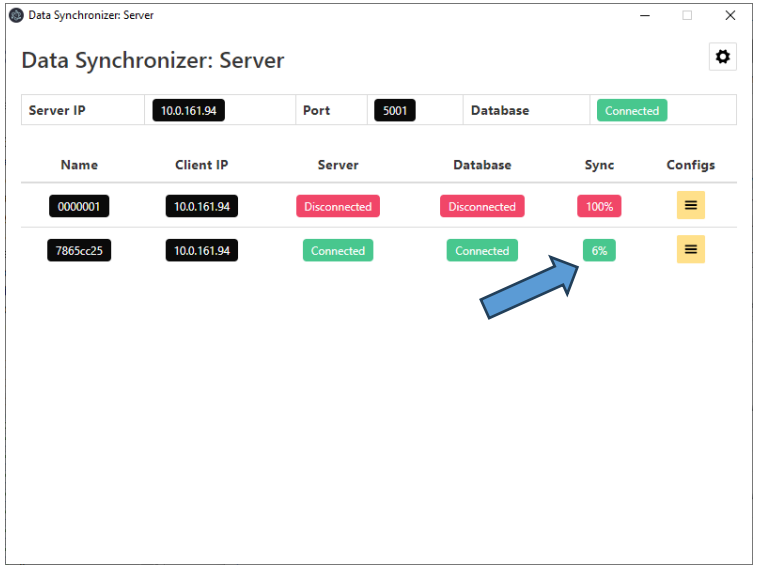


- Click **Cog** button to open settings.
- Enter Port credential. Enter Database credentials.
  (Database name is case-sensitive)
- Click save button and restart application.
- Settings will be valid after restart.

Registration

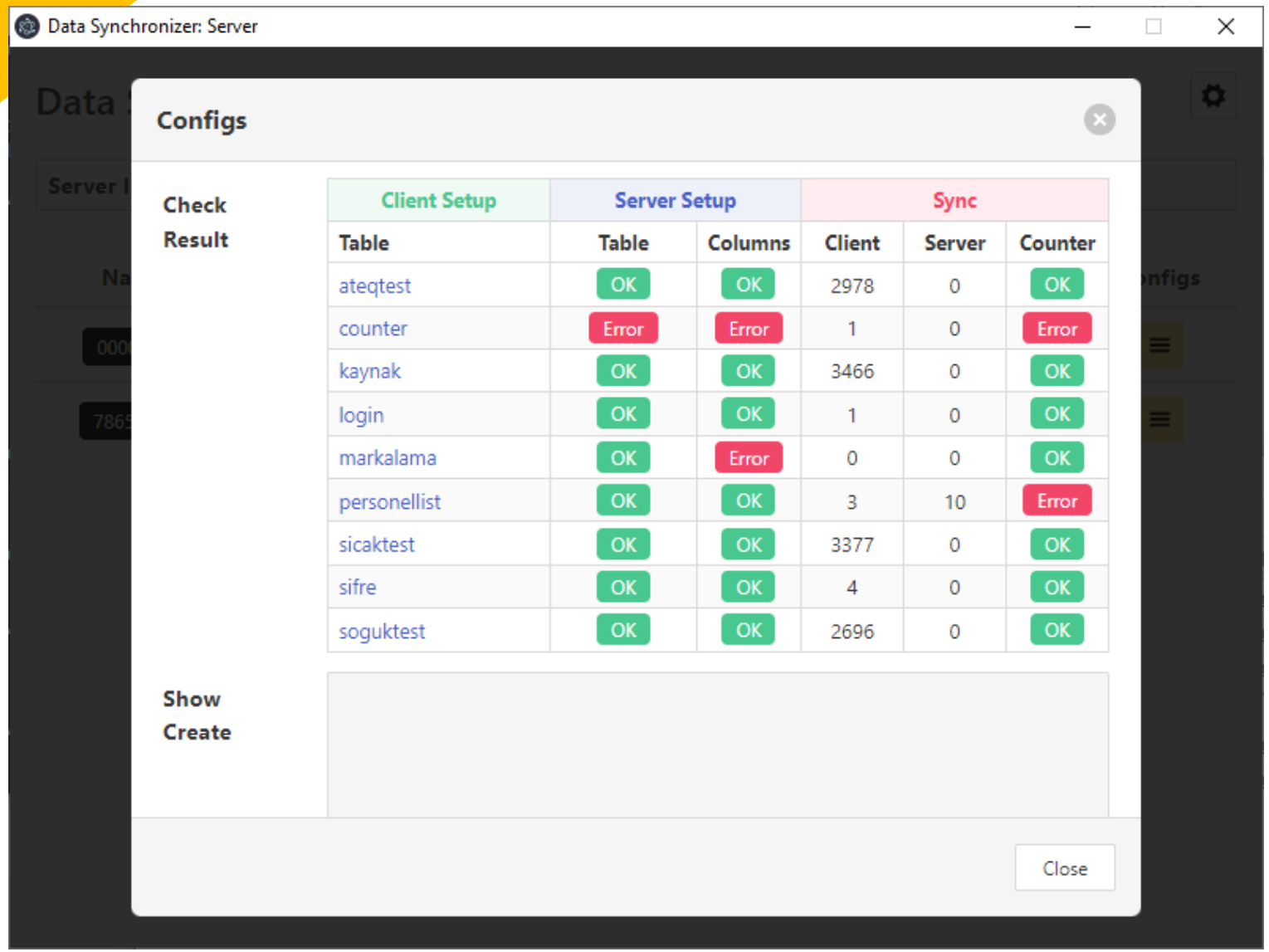






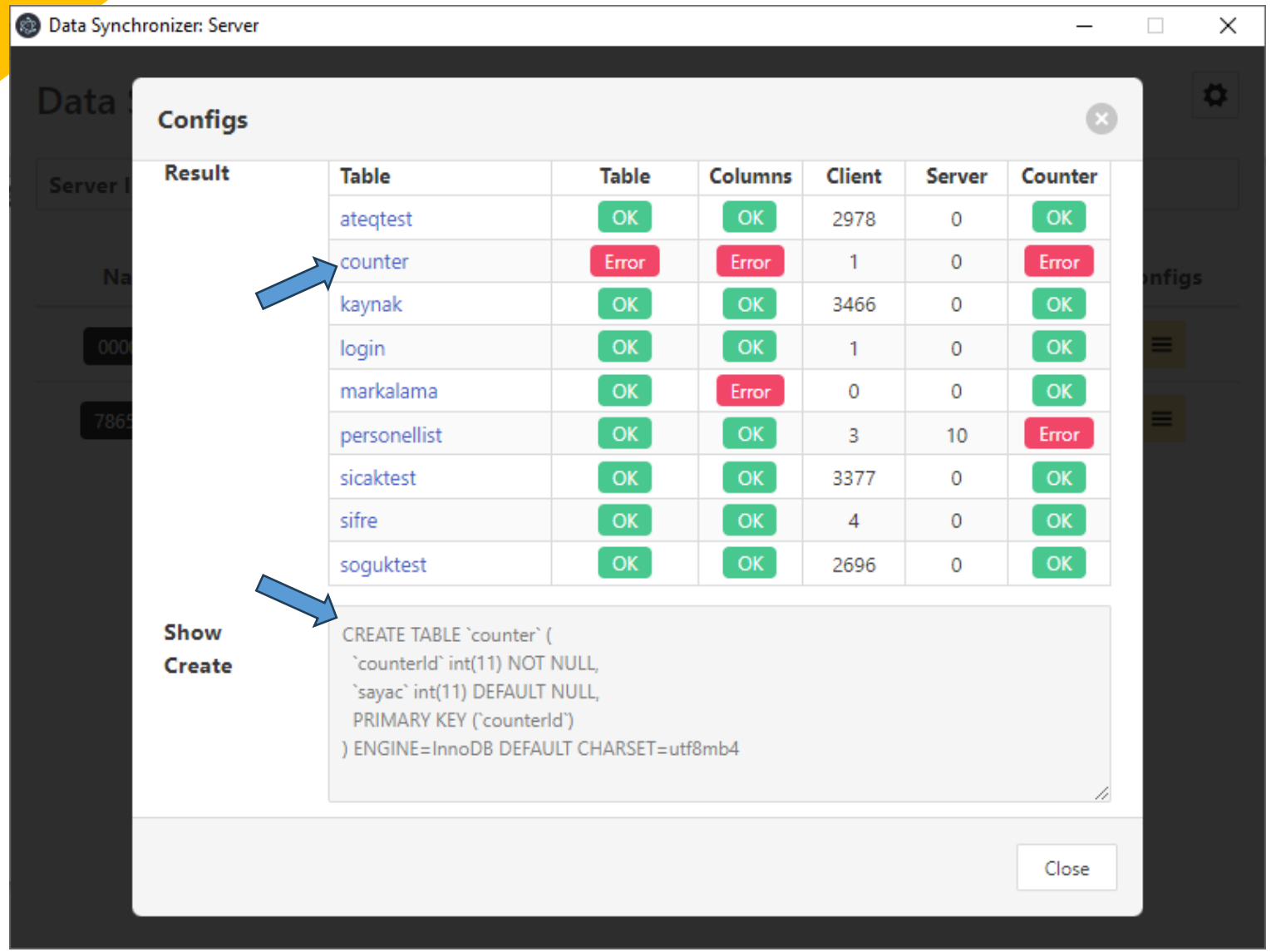
- Clients will appear on screen main screen when connected and will have '---' on Sync column if not registered.
- Click Client's **Config** button to open client configuration window and choose target database.
- Click **Check** button to compare Client & Server database match. *(If check is successful **Bind** button will be activated)*
- Click **Bind** button to bind Client database to selected server database.
- After binding data transfer will start immediately.

Database Binding Error Types



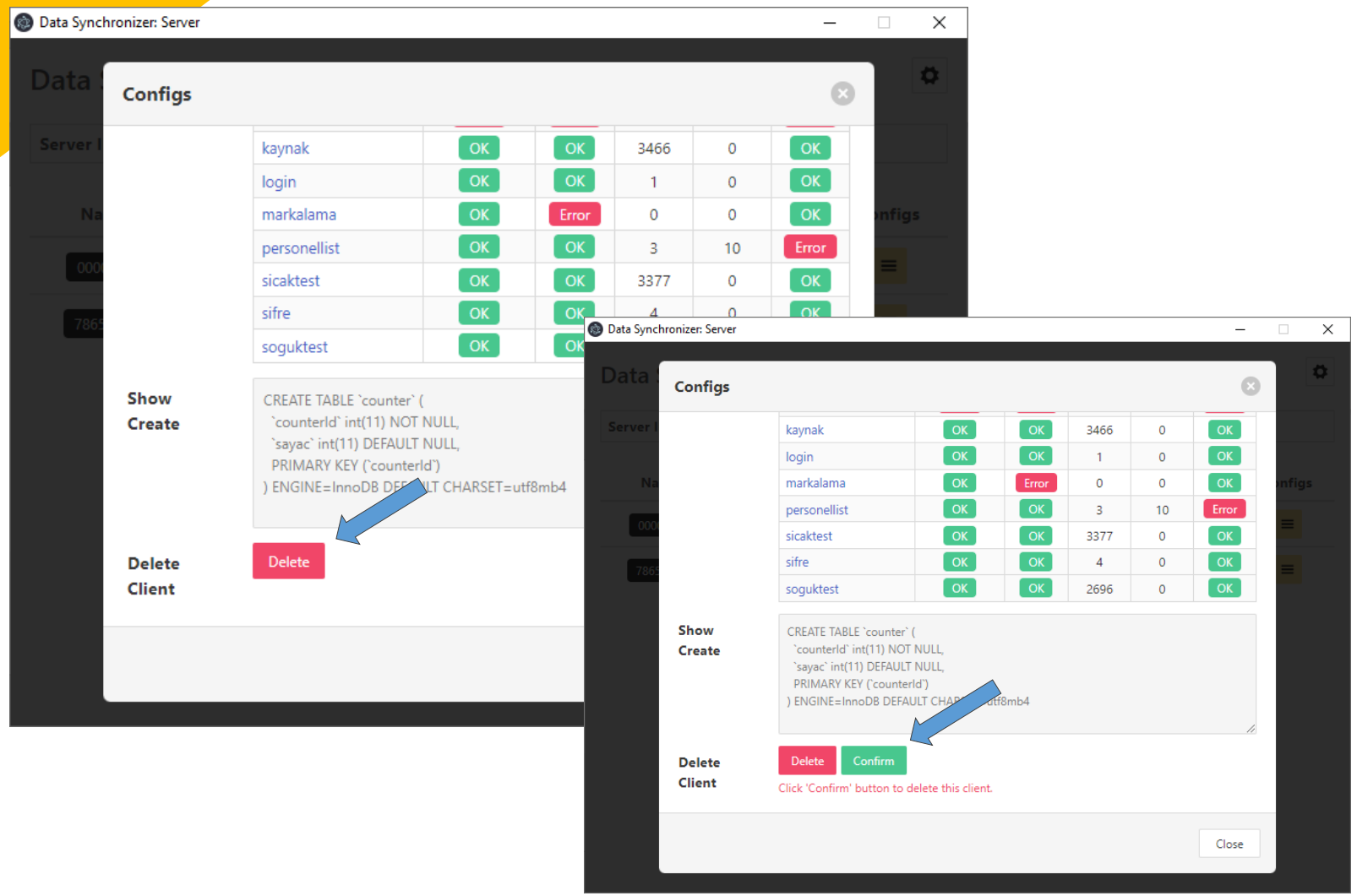
- Check result will be shown in **Check Result** table.
- If **Table** has an error, that means table missing.
- If **Columns** have an error, that means table column name, data type or key does not match.
- If **Counter** has an error this means Server has more records than client. (Counter error does not prevent binding)

Fix Binding Errors



- **Table** and **Column** errors can easily be fixed by using **Show Create** function.
- Click table name to get client's table creation command and fix server database table using this information. *This function can also be used for fast preparing server databases. Create an empty database at server side. Choose that database in server application. Click **Check** button to get **Check Result** table. Click each table name and gather table creation queries*
- In order to fix **Counter** errors, best way to truncate related table from server side and let applications to re-synchronize themselves.

Delete Client



- You can delete a client by clicking Delete and Confirm buttons.
- User will be deleted if Confirm button clicked and will be removed from the main screen.
- This operation will delete user information but will not close socket connection.
- In order to get same user again, restart client or server application. Client will appear. However, you have to register the client again.

Thanks