

Valid as of 3/12/2025

Bitcoin Blockchain Data Extractor - Guideline

This Python script extracts block and transaction data from Bitcoin Core and saves it into a postgresql database. Please note that this script may have a long run time, potentially days. I have spent a lot of time optimizing things, please be cautious if making edits.

Step-by-Step Guide

Installing Bitcoin Core

1. **Download Bitcoin Core:** Visit the Bitcoin Core download page and download the appropriate version for your operating system.
2. **Install Bitcoin Core:** Follow the installation instructions specific to your operating system:
 - o **Windows:** Run the installer and follow the on-screen instructions.
 - o **MacOS:** Open the downloaded .dmg file and drag the Bitcoin Core icon to your Applications folder.
 - o **Linux:** Follow the instructions provided on the download page for your specific Linux distribution.
3. **Configure Bitcoin Core:**
 - o Press windows key + r and navigate to %APPDATA%\Bitcoin
 - o The data directory is typically located at:
 - **Windows:** C:\Users\<YourUsername>\AppData\Roaming\Bitcoin\bitcoin.conf
 - **MacOS:** ~/Library/Application Support/Bitcoin/bitcoin.conf
 - **Linux:** ~/.bitcoin/bitcoin.conf
 - o Add the following lines to the bitcoin.conf file and save it:

```
server=1
rpcuser=<your_rpc_username>
rpcpassword=<your_rpc_password>
rpcport=8332
txindex = 1
rpcthreads=8 # This should be 2 times the number of cores on your machine, 8 is an
example.
rpcworkqueue=32
maxconnections=20 # This helps to prevent overloading the rpc.
datadir = your/storage/path/here #~700 GB for full node as of 2025
```

Replace <your_rpc_username> and <your_rpc_password> with your desired username and password.

- o Note: Bitcoin core will default to the bitcoin.conf file within "C:\Users\USER_NAME\AppData\Roaming\Bitcoin\"
 - If the bitcoin.conf file is not within this path, bitcoin core will ignore it. I could not identify a way to change the default path to search for this file in the documentation.

- o Must restart instance if changes are made to the bitcoin.conf file. **DO NOT UPDATE THE BITCOIN.CONF FILE WHILE BITCOIN DAEMON IS RUNNING.**

Start Bitcoin Core Server

- 1.) Open command prompt
- 2.) Navigate to daemon file
- 3.) Run bitcoind

Soft Stop Bitcoin Core Server

(do NOT press the exit button - this **WILL** result in data corruption.)

- 1.) Open command prompt
- 2.) Navigate to the daemon file
- 3.) Run "bitcoin-cli stop"
- 4.) Wait for prompt that it is safe to press exit button, then exit.

Start postgresql server

- 1.) Open command prompt and navigate to the postgresql bin directory.
- 2.) If database hasn't been built yet, write "initdb -D your_choice_of_drive:\your_postgres_folder\data"
- 3.) Run pg_ctl -D your_choice_of_drive:\your_postgres_folder\data -l logfile start
- 4.) If the database hasn't been built yet, run psql -U postgres, then copy and paste the query in "database_schema.sql"

Note: The lab computer does not have admin privileges and getting the admin user and password is a headache. Download the postgres binaries to your X: drive instead of the main version, it takes seconds!

Installing Python

- **Download Python:** Visit the [Python download page](#) and download the latest version of Python for your operating system.
- **Install Python:**
 - o **Windows/MacOS:** Run the installer and follow the on-screen instructions. Ensure you check the option to add Python to your PATH during installation.
- **Verify Python Installation:** Open Command Prompt or Terminal and run:

```
python --version
```

Run the Script

- Open a terminal or command prompt.
- Navigate to the directory where you saved the extract_bitcoin_data.py script.
- Update Configuration:
 - Open the script in a text editor and update the rpc_user, rpc_password, and csv_folder variables with your settings. Specify the path where the database is stored, the username of the

database (should just be postgres by default) and the databases password. Update any other settings you need to change.

- Run the script with Python:
 - Open Command Prompt or Terminal, navigate to the directory where the script is saved, and run:

```
py extract_bitcoin_data.py Version 3.0 >
```

```
python extract_bitcoin_data.py Version 3.0 <
```

- Input the Starting Block Height. The script will prompt you to enter the starting block height for extraction. If this is your first time downloading this data, it should be 0. Otherwise, it should be the last blockheight you downloaded +1.
 - The script splits the extracted data into multiple CSV files intended to be uploaded to a SQL database. Each file should be somewhat large to speed copying. This method is redundant, however it ensures that the daemon is running for as little time as possible - reducing the risk of corruption (the lab is used by many people).

Example Output CSV Structure (For developers - this will be automatically uploaded to a postgresql database)

The CSV files will have the following columns:

transactions_i_j.csv

- txid: The transaction ID.
- locktime: The locktime for the transaction.
- blocktime: The time the miner reported. median_blocktime < blocktime < network adjusted time + 2 hours.

inputs_i_j.csv

- vin_vout: The index of the output in the previous transaction.
- vin_scriptSig_asm: The asm representation of the scriptSig.
- vin_txinwitness: The witness data for SegWit transactions as a comma separated string ex: "w1, w2, w3"
- vout_value: The value of the output.
- vout_n: The index of the output.

outputs_i_j.csv

- vout_scriptPubKey_asm: The asm representation of the scriptPubKey.
- vout_scriptPubKey_desc: The description of the scriptPubKey.
- vout_scriptPubKey_address: A pubkeyhash address associated with the scriptPubKey if one is "well defined" according to bitcoin core.
- vout_scriptPubKey_type: The type of the scriptPubKey.

Suggestions, Bitcoin Core

- Back up the full node to google drive (one should already exist, though). SBU students have unlimited drive space and it saves lots of time if something goes wrong.
- Developers should carefully read the bitcoin core developer guides and RPC documentation prior to modifying or adding to the code.

Troubleshooting, Bitcoin Core

- **Connection Errors:** Ensure Bitcoin Core is running and the RPC server is configured correctly.
- **Authentication Errors:** Double-check the RPC username and password you entered.
- **Permission Errors:** Ensure you have write permissions to the specified CSV output folder.