

Bitcoin SV Node system requirements.

Below are the recommended system requirements based on our internal testing and scaling progress made with bitcoind node software. Bitcoin SV will continue to scale on the road to and beyond genesis. This will also mean these requirements should be expected to change as time goes on.

For operating a listener node which only operates to follow the most PoW chain and handle small volumes of other tasks (RPC requests as an example) we suggest as a **minimum**, the following.

- 4 Core/Thread CPU
- 6GB of Ram
- 10+ Mbit internet connection (up and down)

Please be aware the above configuration may run into issues if there is a sustained burst of transaction volume. This may result in your node falling behind the current best chain tip, or simply running out of memory and shutting down.

We do not recommend using the above for any production environments.

For operating a listener node which is expected to handle a medium volume of workload while maintaining real time sync with the current chain tip we recommend a **minimum** of the following.

- 8 Core/Thread CPU
- 16GB Ram
- 100Mbit+ Internet connection (up and down)

This may be suitable for a startup, wishing to launch their first product on to mainnet. As volume on their application and mainnet increases, so will the requirements for this tier of node.

For operating a listener node which expects a high volume of work or has txindex enabled, or a mining operation, we suggest the following as a **minimum**.

- 8-12 core/thread CPU
- 32GB Ram
- 1Gbit+ Internet Connection (up and down)

We have seen the above configuration in both mining and listener environments handle sequential 2GB block sizes and blocks with transaction counts exceeding 1 million transactions on the [STN](#) (using the additional recommendations below). This should position you comfortably on mainnet until the current miner enforced block size hard cap (512MB as of AUG 2019).

This may vary as your individual demand scales up with your specific environment, application or use case.

If you are a miner, it is also advisable you spend time ensuring your nodes have the highest possible connectivity with other miners.

Additional Recommendations – Especially for Miners

With the increasing adoption of Bitcoin SV the transaction volume continues to rise, along with the explosive use of data transactions (op_returns) it is possible your Bitcoin SV node will not be able to handle the volume of traffic reaching your mempool or be inundated with computationally heavy requests. As a result the node will drop transactions to allow higher fee paying ones in, increasing computation at a later point, or worse cease to function.

Whilst this is not how Bitcoin is intended to work, it is what we have to deal with for the short term while the SV Node teams focuses on higher priority tasks which have a greater impact on scaling.

A solution is to increase these following values from their defaults is to allow the node to remain efficient under high load situations. These situations include reorgs, which require the node to go back and reconsider transactions or blocks it has most probably already seen. A reorg can be the reason your node spikes from 1-2GB ram use to 3GB or more, if this is too much for your system the operating system may choose to end the process (stop bitcoind), or your node will crash with an “Out of Memory” error code.

Since reorgs and orphans are a part of the Bitcoin ecosystem and should be expected and not feared (more on this [here](#)) it would be wise to best prepare your environment for such situations. The default impaired settings and concepts inherited in bitcoind are too small for the volumes we see during operation on the [STN](#) or during a stress test on mainnet.

With this in mind, we suggest increasing a few default settings on your bitcoind node.

First of all, your mempool size allowance should be set to 6GB or more. This tells the node how much memory it should assign to storing unconfirmed transactions. This is done by adding the following to your bitcoin.conf file.

```
maxmempool=6000
```

This restrictive memory limitation (300MB by default) is a consequence of the fee priority processing inherited from BTC in order to maintain functioning tiny block sizes. In Bitcoin SV, we don't need this. The current overhead for storing transactions is in the realm of 5 times the real transaction size, for small transactions. This decreases dramatically for larger transactions. The SV Node team are actively working to remove all fee prioritisation code and hasten mempool processing to bring much needed improvements to transaction propagation, acceptance and memory allocation. The net result will be a much faster and less memory expensive mempool.

In addition to increasing the mempool allowance we also suggest increasing the signature and script cache. This tells the node how many accepted transactions in megabytes we can keep in our cache (Ram), improving performance by reducing expensive calls to

recalculate signatures and scripts on the fly. We suggest setting these to 250MB or more to improve performance. This is done by adding the following to your bitcoin.conf file.

```
maxsigcachesize=250
```

```
maxscriptcachesize=250
```

Please be aware that setting all three of the mentioned settings will add an additional memory requirement of 6.5GB on your node just for this aspect of bitcoind's operation.

Lastly we suggest adding maxorphantx to your bitcoin.conf as well. This value specifies how many orphan transactions can be kept in memory. This helps if your node is receiving a child transaction whose parent has not been confirmed in the blockchain. This means that the node will remember the child until it sees the parent or it exceeds its expiration time of 20 minutes. This is done by adding the following to your bitcoin.conf file.

```
maxorphantx=10000
```

The result of this, assuming 400byte average transaction sizes is only a 4MB memory increase. The reason to keep this from being set at say 100,000 is if you start holding on to 100,000 100KB transactions in the most extreme case you will need an additional 10GB of memory available.