```
BIP: 101
Layer: Consensus (hard fork)
Title: Increase maximum block size
Author: Gavin Andresen <gavinandresen@gmail.com>
Comments-Summary: No comments yet.
Comments-URI: https://github.com/bitcoin/bips/wiki/Comments:BIP-0101
Status: Withdrawn
Type: Standards Track
Created: 2015-06-22
```

Abstract

This BIP proposes replacing the fixed one megabyte maximum block size with a maximum size that grows over time at a predictable rate.

Motivation

Transaction volume on the Bitcoin network has been growing, and will soon reach the one-megabyte-every-ten-minutes limit imposed by the one megabyte maximum block size. Increasing the maximum size reduces the impact of that limit on Bitcoin adoption and growth.

Specification

After deployment (see the Deployment section for details), the maximum allowed size of a block on the main network shall be calculated based on the timestamp in the block header.

The maximum size shall be 8,000,000 bytes at a timestamp of 2016-01-11 00:00:00 UTC (timestamp 1452470400), and shall double every 63,072,000 seconds (two years, ignoring leap years), until 2036-01-06 00:00:00 UTC (timestamp 2083190400). The maximum size of blocks in between doublings will increase linearly based on the block's timestamp. The maximum size of blocks after 2036-01-06 00:00:00 UTC shall be 8,192,000,000 bytes.

Expressed in pseudo-code, using integer math, assuming that block_timestamp is after the activation time (as described in the Deployment section below):

```
function max_block_size(block_timestamp):
```

```
    time_start = 1452470400
time_double = 60*60*24*365*2
size_start = 8000000
if block_timestamp >= time_start+time_double*10
return size_start * 2^10
```

```
    // Piecewise-linear-between-doublings growth:
time_delta = block_timestamp - time_start
```

```
doublings = time_delta / time_double
remainder = time_delta % time_double
interpolate = (size_start * 2^doublings * remainder) / time_double
max_size = size_start * 2^doublings + interpolate

       return max_size
```

Deployment

Deployment shall be controlled by hash-power supermajority vote (similar to the technique used in BIP34), but the earliest possible activation time is 2016-01-11 00:00:00 UTC.

Activation is achieved when 750 of 1,000 consecutive blocks in the best chain have a version number with the first, second, third, and thirtieth bits set (0x20000007 in hex). The activation time will be the timestamp of the 750'th block plus a two week (1,209,600 second) grace period to give any remaining miners or services time to upgrade to support larger blocks. If a supermajority is achieved more than two weeks before 2016-01-11 00:00:00 UTC, the activation time will be 2016-01-11 00:00:00 UTC.

Block version numbers are used only for activation; once activation is achieved, the maximum block size shall be as described in the specification section, regardless of the version number of the block.

Test network

Test network parameters are the same as the main network, except starting earlier with easier supermajority conditions and a shorter grace period:

```
   starting time: 1 Aug 2015 (timestamp 1438387200)
activation condition: 75 of 100 blocks
grace period: 24 hours
```

Rationale

The initial size of 8,000,000 bytes was chosen after testing the current reference implementation code with larger block sizes and receiving feedback from miners on bandwidth-constrained networks (in particular, Chinese miners behind the Great Firewall of China).

The doubling interval was chosen based on long-term growth trends for CPU power, storage, and Internet bandwidth. The 20-year limit was chosen because exponential growth cannot continue forever. If long-term trends do not continue, maximum block sizes can be reduced by miner consensus (a soft-fork).

Calculations are based on timestamps and not blockchain height because a timestamp is part of every block's header. This allows implementations to know a block's maximum size after they have downloaded it's header, but before downloading any transactions.

The deployment plan is taken from Jeff Garzik's proposed BIP100 block size increase, and is designed to give miners, merchants, and full-node-running-end-users sufficient time to upgrade to software that supports bigger blocks. A 75% supermajority was chosen so that one large mining pool does not have effective veto power over a blocksize increase. The version number scheme is designed to be compatible with Pieter's Wuille's proposed "Version bits" BIP, and to not interfere with any other consensus rule changes in the process of being rolled out.

Objections to this proposal

Raising the 1MB block size has been discussed and debated for years.

Centralization of full nodes

The number of fully-validating nodes reachable on the network has been steadily declining. Increasing the capacity of the network to handle transactions by increasing the maximum block size may accelerate that decline, meaning a less distributed network that is more vulnerable to disruption. The size of this effect is debatable; the author of this BIP believes that the decline in fully validating nodes on the network is largely due to the availability of convenient, attractive, secure, lightweight wallet software and the general trend away from computing on desktop computers to mobile phones and tablets.

Increasing the capacity of the network to handle transactions should enable increased adoption by users and businesses, especially in areas of the world where existing financial infrastructure is weak. That could lead to a more robust network with nodes running in more political jurisdictions.

Centralization of mining: costs

Miners benefit from low-latency, high-bandwidth connections because they increase their chances of winning a "block race" (two or more blocks found at approximately the same time). With the current peer-to-peer networking protocol, announcing larger blocks requires more bandwidth. If the costs grow high enough, the result will be a very small number of very large miners.

The limits proposed by this BIP are designed so that running a fully validating node has very modest costs, which, if current trends in the cost of technology continue, will become even less expensive over time.

Centralization of mining: big-block attacks

Simulations show that with the current peer-to-peer protocol, miners behind high-latency or low-bandwidth links are at a disadvantage compared to miners connected to a majority of hashpower via low-latency, high-bandwidth links. Larger blocks increase the advantage of miners with high-bandwidth connections, although that advantage can be minimized with changes to the way new blocks are announced (e.g. http://bitcoinrelaynetwork.org/).

If latency and bandwidth to other miners were the only variable that affected the profitability of mining, and miners were driven purely by profit, the end result would be one miner running on one machine, where latency was zero and bandwidth was essentially infinite.

However, many other factors influence miner profitability, including cost of electricity and labor and real estate, ability to use waste heat productively, access to capital to invest in mining equipment, etc. Increasing the influence of bandwidth in the mining profitability equation will not necessarily lead to more centralization.

Unspent Transaction Output Growth

This BIP makes no attempt to restrict the approximately 100% per-year growth in unspent transaction outputs (see http://gavinandresen.ninja/utxo-uhoh for details), because the author believe that problem should be solved by a further restriction on blocks described in a separate BIP (perhaps an additional limit on how much the transactions in any block may increase the UTXO set size).

Long-term fee incentives

http://gavinandresen.ninja/block-size-and-miner-fees-again

Other solutions considered

There have been dozens of proposals for increasing the block size over the years. Some notable ideas:

One-time increase

A small, quick one-time increase to, for example, 2MB blocks, would be the most conservative option.

However, a one-time increase requires just as much care in testing and deployment as a longer-term fix. And the entire debate over how large or small a limit is appropriate would be repeated as soon as the new limit was reached.

Dynamic limit proposals

BIP 100 proposes a dynamic limit determined by miner preferences expressed in coinbase transactions, with limits on the rate of growth. It gives miners more direct control over the maximum block size, which some people see as an advantage over this proposal and some see as a disadvantage. It is more complex to implement, because the maximum allowed size for a block depends on information contained in coinbase transactions from previous blocks (which may not be immediately known if block contents are being fetched out-of-order in a 'headers-first' mode).

Meni Rosenfeld has proposed that miners sacrifice mining reward to "pay for" bigger blocks, so there is an incentive to create bigger blocks only if transaction fees cover the cost of creating a larger block. This proposal is significantly more complex to implement, and it is not clear if a set of parameters for setting the cost of making a block bigger can be found that is not equivalent to a centrally-controlled network-wide minimum transaction fee.

Compatibility

This is a hard-forking change to the Bitcoin protocol; anybody running code that fully validates blocks must upgrade before the activation time or they will risk rejecting a chain containing larger-than-one-megabyte blocks.

Simplified Payment Verification software is not affected, unless it makes assumptions about the maximum depth of a transaction's merkle branch based on the minimum size of a transaction and the maximum block size.

Implementation

https://github.com/bitcoin/bitcoin/pull/6341