BTMT Technical Report

The BITMarkets team

February 22, 2023

Abstract

The BITMarkets token, or BTMT, is an ERC20 token that resides on the Polygon blockchain. It enables the users of bitmarkets.com to perform cryptocurrency exchanges with low fees, to participate in new token projects and initial exchange offerings in the platform and to indicate their preference regarding ecological and social impact investments of the company. This document is a work in progress. To check the latest version you can visit its public GitHub repository.

Keywords: bitmarkets, blockchain, token sale, polygon, solidity

1 Introduction

BITMarkets is a cryptocurrency exchange which has the mission to make crypto simple. This document describes the technical details of the native token of the exchange, namely the BTMT. If you would like to read a marketing-oriented document with less technical details, you can read the corresponding litepaper.

Trading pairs that involve BTMT will have lower platform exchange fees than those that don't. Moreover, users that hold BTMT in their portfolio will be eligible for airdrops, NFT lotteries and other perks that will be announced in the future. Finally, holders of BTMT will be able to participate in platform-exclusive initial token sales and IEOs, which will be a great way for aspiring clients to contribute in vetted, high impact projects that they may find interesting.

BTMT is an ERC20 token on the Polygon blockchain. BITMarkets believes that Polygon is the most suitable home for its smart contracts because it offers significant improvements over other Ethereum-compatible blockchains in terms of speed, scalability and transaction costs, which in turn offers greater user experience for its holders.

2 Token

The BTMT token uses the ERC20 standard, which defines a set of functions that a smart contract implements to allow external clients to interact with the fungible token. Some of these functions provide descriptive data about the token, such as its name, its symbol, the number of decimals that it has and the total supply of tokens in existence, while the rest of the functions in the ERC20 standard have to do with getting the token balance of an address, the transfer of tokens from one address to the other, the so-called "approval", which is an operation that allows one address to spend some amount of the balance of another address. To be compliant with the standard, BTMT inherits from the battle-tested ERC20 smart contract offered by OpenZeppelin on their GitHub repository.

3 Distribution

The total minted supply of BTMT on deployment is 300 million tokens. 100 million of those will be sold in public and private sales, specifically 40 million in a private sale and 60 million in a public sale. The second 100 million are allocated on deployment to many different wallets that are adjacent to the company. Specifically, 30% will go to wallets belonging to the BITMarkets team members, 25% to marketing-related wallets, 40% to wallets that will be distributed to the salespeople and 5% to wallets that will eventually be airdropped. All these wallets have a cliff period of 9 months and will receive linear vesting for 10 months. The rest of the tokens are in the company liquidity wallet, which will provide rewards to early backers, bonuses for team members, liquidity for future exchange listings and it will receive monthly burning of 0.1% of the token total supply to gradually reduce its size.

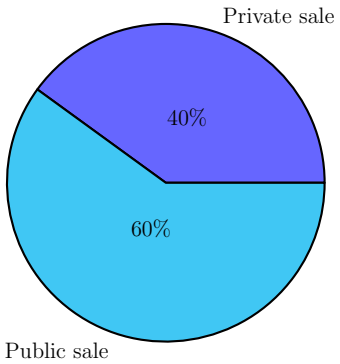

Figure 1: Public/Private Sales.

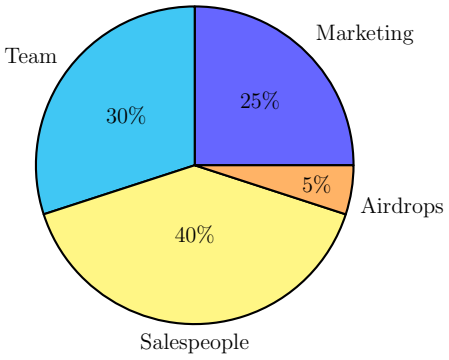

Figure 2: Team allocations.

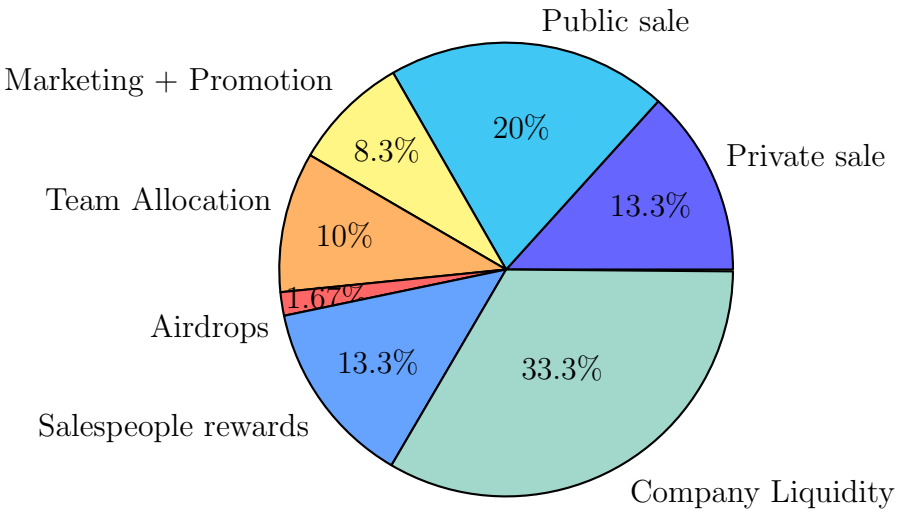

Figure 3: Distribution of total supply after minting.

3.1 Important addresses

The BITMarkets-uploaded smart contracts are the following:

1. The ERC20 contract with address and source code available. This contract inherits from various OpenZeppelin smart contracts that provide functionality such as snapshots, burning and pausing. It also inherits from two BITMarkets-specific contracts that impose restrictions on strategic wallets' transfers and transfer fees of maximum 1% of the total.

2. The team allocations contract with address and source code available. This contract generates and remembers the vesting wallets of the beneficiaries of the allocations in Figure 2. The beneficiaries will access public functions in this contract to give them the vested amount of tokens that they can withdraw.

3. The private sale contract with address and source code available. This contract is a private sale contract, loosely based on an OpenZeppelin contract from version 2.0. It sets up a sale which has a specific exchange rate between MATIC and BTMT, is open in a specific time window, saves individual participants' total contributions and imposes a tariff and a cap on them, generates vesting wallets where the purchased tokens are transferred and provides public functions to access these wallets.

The BITMarkets wallet infrastructure is as follows:

1. Company liquidity wallet: This wallet holds 100 million BTMT, it is classified as a *strategic* wallet and is therefore restricted, in the sense that if the accumulated transactions reach 10 million then it will be locked for 1 month. BITMarkets uses this wallet to deploy the relevant smart contracts. This wallet's private key is split into many parts using Shamir secret sharing so that it cannot be compromised by targeting a single holder.

2. Allocations wallet: This wallet also starts with 100 million tokens, it is also classified as strategic and its restriction is that it is only allowed to transfer tokens to the vesting wallets that are generated by the allocations-related smart contract. After deployment, the allocations will happen according to Figure 2 and this wallet will not have any tokens anymore.

3. Public/Private sales wallet: This wallet also holds 100 million tokens on deployment, it is strategic, and its restriction is that it is only allowed to transfer to vesting wallets generated by the private sale and the eventual public sale smart contracts.

4. Company rewards wallet: This wallet receives 0.33% of each transfer amount as a company reward. The fee structure is handled by this smart contract.

5. ESG wallet: This wallet also receives 0.33% of each transfer amount. The plan is to use this wallet for social impact investments by the company, based on the preferences of the BTMT holders. The smart contract for this functionality has not been created as of yet.

6. Pauser wallet: This wallet has the ability to pause all transfers. This functionality is useful if there is a bug somewhere.

7. Whitelister wallet: This wallet can whitelist client addresses to the private sale so that they can participate.

8. Feeless admin: This wallet can make smart contracts feeless admins. This functionality exists because sales and allocations smart contracts need to be able to make their generated vesting wallets feeless because these wallets do not make signed transfers so as to be able to pay for fees. These feeless admin rights to smart contracts are given on deployment. This key also gives feeless status to the ESG wallet and the company rewards wallet.

9. Company restrictions admin: This wallet can authorize the company liquidity wallet to make one unrestricted transfer to a specific address with specific amount, even if the liquidity wallet is locked. It also gives unrestricted access to the allocations and public/private sales wallets in their respective token wallets.

10. Allocations admin: This wallet can ask the allocations smart contract to create vesting wallets. At some point it will be useless as it will have allocated all the tokens of the allocations wallet to their beneficiaries.

11. Sales client purchaser: This wallet is used by the server infrastructure of BITMarkets in order to participate in the public and private sales on behalf of clients.

3.2 ERC20 Extensions

The ERC20 extensions that BITMarkets has authored for BTMT adhere to our business model and provide safety restrictions for the strategic wallets of the company.

3.2.1 Deflation

BTMT follows a deflationary supply model which reduces overtime the amount of tokens in circulation and therefore increases their scarcity. Deflation is achieved by two avenues. The first one is a 0,33% burn on every transfer and the second one is the monthly burning of 0.1% of the total supply from wallet #1 (company liquidity wallet). The first burning occurs by removing this percentage from the amount to be transferred and then proceeding to burning this amount from the sender's wallet. The second burning is manual and it can be used as a test of the trustworthiness of the company and the Shamir share holders.

3.2.2 Fees

On deployment, this extension is uploaded, which calculates the transfer fees that will be distributed to two company-controlled wallets, namely, wallet #4 (company rewards) and #5 (esg wallet), based on prespecified percentages on the total amount of a transfer that are made available on deployment. These fees are removed from the transfer amount and transferred from the sender's wallet to the two mentioned wallets. The fees will be 0.33% for each wallet and they will apply to all transfers. Another percentage of 0.33% is removed from the transfer amount and burned.

The four wallets that are excluded from fee collection on deployment are wallets #4 (company rewards), #5 (ESG), #3 (public/private sales) and #2 (allocations). The feeless functionality methods can be accessed publically but the ability to add or remove a wallet from the feeless list is reserved for wallet #8 (feeless admin), which provides the priviledge also to the sales and allocations smart contracts for their generated vesting wallets.

```
function addFeelessAdmin(address) public onlyFeelessAdmin;
function addFeeless(address) public onlyFeelessAdmins;
function removeFeeless(address) public onlyFeelessAdmins;
function isFeeless(address) public returns (bool);
```

Listing 1: Solidity feeless functions.

The corresponding events that are emitted on successful completion of the state-mutating functions are:

```
event FeelessAdded(address indexed);
event FeelessRemoved(address indexed);
```

Listing 2: Solidity feeless events.

3.2.3 Restrictions

The three wallets which hold a large amount of tokens in different times of the token's lifecycle (#1, #3, #2) need restrictions on their transfers so that the users can trust them and so that there exists some decentralized security on their tokens. You can browse the source code of the smart contract governing these restrictions. The relevant wallet here is #9 (restrictions admin), which serves as the wallet that offers allowance to the allocations and sales smart contracts to transfer tokens from their respective wallets.

```
function addUnrestrictedReceiver(address, address, uint256)
    public onlyRestrictionsAdmin;
function removeUnrestrictedReceiver(address) public
    onlyRestrictionsAdmin;
function isStrategicWallet(address) public returns (bool);
function getApprovedReceiver(address) public returns (address
    );
function getApprovedReceiverLimit(address) public returns (
    uint256);
function companyLiquidityTransfersLimit() public returns (
    uint256);
function companyLiquidityTransfersSinceLastLimitReached()
    public returns (uint256);
function timeSinceCompanyLiquidityTransferLimitReached()
    public returns (uint256);
```

```
function companyLiquidityTransfersAreRestricted() public
    returns (bool);
```

Listing 3: Solidity strategic wallets restriction functions.

4 Sales

One third of the initial supply of BTMT will be sold in public and private sales. The smart contracts that govern these sales have a combined allowance of 100 000 000 tokens from the sales wallet to distribute to the buyers. The smart contracts expect to trade MATIC, Polygon's native cryptocurrency, with BTMT. The sales wallet will receive the MATIC that the buyer sends to the sales smart contracts and the contract will send in return the amount of BTMT that corresponds to the rate that is derived from the contract's code to a vesting wallet whose beneficiary is the purchaser. The purchased BTMT comes from the sales wallet, provided that it does not exceed the contract's allowance. The most important publicly exposed methods of public and private sale contracts are the following:

```
function buyTokens(address) public payable;
function getContribution(address) public returns (uint256);
function remainingTokens() public view returns (uint256);
function token() public view returns (IERC20);
function tokenWallet() public view returns (address);
function wallet() public view returns (address payable);
function weiRaised() public view returns (uint256);
```

Listing 4: Solidity sale function signatures

Every sale happens in a limited time window that is specified on deployment. The private sale is planned to run from the 1st of March 2023 until 26th of June 2023. In order for external programs to track the timing, the timing smart contract provides the following, non-state-mutating functions:

```
function isOpen() public view returns (bool);
function hasClosed() public view returns (bool);
function paused() public view returns (bool);
function openingTime() public view returns
function closingTime() public view returns (uint256);
```

Listing 5: Solidity timed sale function signatures

The smart contracts have algorithmic safeguards in order to ensure fair access to the sales for as many buyers as possible. The first and most obvious safeguard is that the hard cap of the sale will be 100 000 000 tokens. Two publicly accessible methods that give feedback regarding these safeguards which do not mutate the state are:

```
function cap() public view returns (uint256);
function capReached() public view returns (bool);
```

Listing 6: Solidity sales cap function signatures.

The second safeguard is that an individual address will have both a tariff to participate and an individual cap to contribute:

```
function getInvestorCap() public view returns (uint256);
function getInvestorTariff() public view returns (uint256);
function investorCap() public returns (uint256);
function investorTariff() public returns (uint256);
```

Listing 7: Solidity sales tariff/cap function signatures.

The tariff and the cap is in MATIC and it is added upon deployment. The timing of the sale is also written in code so no buyer can exchange MATIC for BTMT with the conditions that we discussed at a time prior to the specified opening and after the closing time. The contract will emit the following event on successful purchase:

```
event TokensPurchased(
  address indexed purchaser,
  address indexed beneficiary,
  uint256 value,
  uint256 amount
);
```

Listing 8: BTMT private sale events.

Users who are not in possession of a decentralized Ethereum, Polygon wallet will be able to participate in the sales by exchanging USD, EUR, BTC, ETH, USDT, MATIC to BTMT on the BITMarkets platform. They will need to pass a KYC check in order to exchange "fiat" for BTMT and then the BITMarkets backend will trigger a server-side transfer of the corresponding BTMT amount to a vesting wallet that corresponds to the client's platform-managed Polygon wallet.

4.1 Private sale

There will be a private sale with a fixed exchange rate between MATIC and BTMT which will happen before the public sale that will take place later in 2023. The company will provide a way for prospective buyers to make it into the whitelist, either by completing a number of tasks, as a gift for their dedication or for VIP clients on the platform. The publicly accessible functions that are relevant to the whitelist are the following:

```
function addWhitelisted(address) public;
```

```
function removeWhitelisted(address) public;
function isWhitelistAdmin(address) public view returns (bool)
function isWhitelisted(address) public view returns (bool);
```

Listing 9: Solidity whitelisted private sale function signatures.

5 Vesting

In order to ensure fair use of BTMT by the team and to reduce the volatility of its exchange price in the short run, there is a locking and vesting functionality built into the sales and the allocation contracts.

The vesting occurs linearly and starts from a point in time that is called the "cliff". As time goes by, more and more tokens are unlocked from the purpose-generated vesting wallets and are claimable by their original owners. The mapping of the beneficiary and their vesting wallet is stored on the blockchain and it is visible to everyone. The functions that expose this functionality to the public are the following:

```
function vestingWallet(address) public view returns (address)
function vestedAmount(address) public view returns (uint256);
function withdrawTokens(address) public;
```

Listing 10: Solidity vesting function signatures.

In the private sale, the cliff for buyers is 6 months after the initial purchase and then linear vesting for 10 months. In the allocations, the cliff is 9 months and the linear vesting duration is again 10 months.

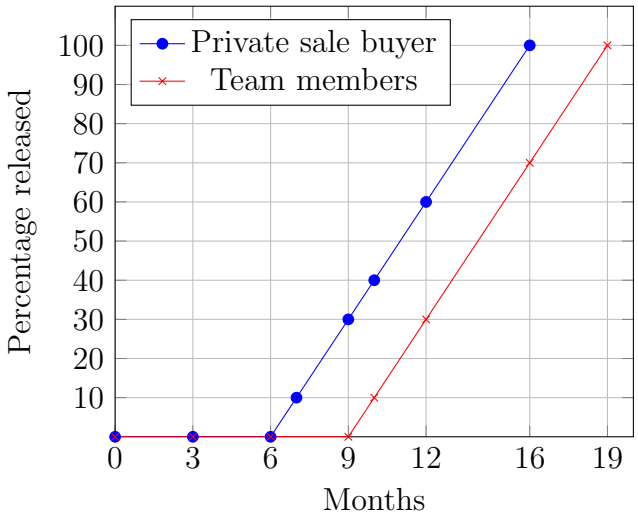

Figure 4: Private sale buyer vesting schedule vs team members.

6 Utility

Holders of BTMT will enjoy reduced exchange fees for trading in spot, future and margin. More specifically, the fee structure is as follows:

VIP Level	Maker	Taker	Token Holding in USD	Or 30-day Spot Volume
General	0.5000	0.5000	< 2500	0
VIP 1	0.4750	0.4875	≥ 2500	1000000
VIP 2	0.4500	0.4750	≥ 5000	3000000
VIP 3	0.4250	0.4625	≥ 10000	5000000
VIP 4	0.4000	0.4500	≥ 50000	7500000
VIP 5	0.3750	0.4375	≥ 100000	10000000
VIP 6	0.3500	0.4250	≥ 300000	13000000
VIP 7	0.3250	0.4125	≥ 500000	16000000
VIP 8	0.3000	0.4000	≥ 1000000	20000000
VIP 9	0.2750	0.3875	≥ 3000000	24000000
VIP 10	0.2500	0.3750	≥ 5000000	30000000

Table 1: Spot trading tiered fees.

Around Q4 2023 there is a plan to introduce a "Token Projects" platform, where promising projects with little-to-no marketing budget will be able to list their upcoming offerings and exchange a predefined amount of their tokens with BTMT. BTMT holders will be able to participate in these sales inside the platform and smart contracts will handle the transfer of their newly purchased tokens to their wallets. The platform will provide this service to the chosen projects in exchange for 15% of their accumulated tokens. Moreover, 2 out of every 5 slots in the list of upcoming token project launches into hourly auctions in order to cover the cost of marketing for all the projects and to monetize the benefit of high placement advertising.

BITMarkets also creates utility from the transaction costs of BTMT. Specifically, the "ESG wallet" that will accumulate transfer fees will serve as an instrument for social contributions by BITMarkets. Our goal is for the BTMT holders to be able to participate in Governance votes that will determine the destination of the accumulated "social contribution" units every 6 months. The BITMarkets team will put together a list of all the potential projects and the community will hold a vote that determines the top 5 projects that will receive the tokens and a corresponding smart contract will execute the transactions.

VIP Level	Maker	Taker	Token Holding in USD	Or 30-day Futures Volume
General	0.1000	0.1000	< 2500	0
VIP 1	0.0950	0.0975	≥ 2500	5000000
VIP 2	0.0900	0.0950	≥ 5000	10000000
VIP 3	0.0850	0.0925	≥ 10000	25000000
VIP 4	0.0800	0.0900	≥ 50000	50000000
VIP 5	0.0750	0.0875	≥ 100000	100000000
VIP 6	0.0700	0.0850	≥ 300000	250000000
VIP 7	0.0650	0.0825	≥ 500000	500000000
VIP 8	0.0600	0.0800	≥ 1000000	1000000000
VIP 9	0.0550	0.0775	≥ 3000000	2000000000
VIP 10	0.0500	0.0750	≥ 5000000	5000000000

Table 2: Futures trading tiered fees.

VIP Level	Maker	Taker	Token Holding in USD	Or 30-day Margin Volume
General	0.0750	0.0750	< 2500	0
VIP 1	0.0600	0.0675	≥ 2500	5000000
VIP 2	0.0450	0.0600	≥ 5000	10000000
VIP 3	0.0300	0.0525	≥ 10000	25000000
VIP 4	0.0150	0.0450	≥ 50000	50000000
VIP 5	0.0000	0.0375	≥ 100000	100000000
VIP 6	-0.0150	0.0300	≥ 300000	250000000
VIP 7	-0.0300	0.0225	≥ 500000	500000000
VIP 8	-0.0450	0.0150	≥ 1000000	1000000000
VIP 9	-0.0600	0.0075	≥ 3000000	2000000000
VIP 10	-0.0750	0.0000	≥ 5000000	5000000000

Table 3: Margin trading tiered fees.

Finally, holders of BTMT will enjoy exclusive perks on the BITMarkets platform such as limited airdrops, NFT lotteries when we start introducing collections in our platform, artificial trading strategies ("ATS"), receive increased referral rates and many more.

7 Conclusion

This was the technical analysis of BTMT, an ERC20 token that will reside on the Polygon blockchain. It's a token whose utility will be in both the digital and the physical world. The BITMarkets platform will be more that an exchange platform and BTMT will be the native currency to its ecosystem. Projects with great potential will benefit from the vibrant community that will be hosted and developer teams that focus on the technical aspect of their innovation will be able to offload the burden of marketing, token sale design and backer search to us. It is our hope that this will make BITMarkets the go-to place for bright innovators and socially-conscious blockchain enthusiasts alike. It is also our goal to be as transparent as possible and this is why we presented in detail all the aspects of the smart contracts that govern our token and its sales.