wagerr

- Peerless Direct Chain Betting Protocols -

Glossary of Terms

Bet Payout: refers to the amount of Wagerr paid out as a result of a Valid Bet(s)

Bet Stake: means the amount of Wagerr placed on a Bet

Bet Transaction: a Transaction on chain holding the details of a verifiable Bet

Betting Options: means the current available betting options to Bettors

Bettor: refers to an individual that uses Wagerr to make a Bet

Blockchain: here referring to the Wagerr Blockchain Network

Burning: the destruction of Wagerr Coins: the process of removing Coins from circulation

Development Fund: refers to 10% of Betting Fees that are allocated to the Wagerr development fund

Event Transaction: a Transaction on chain holding the details of a Betting Event

Betting Fees: 6% of a Bet Stake

Minting: the creation of new Wagerr coins to payout winning Bets

OMN Fees: refers to 40% of betting fees paid to OMN for facilitating the betting & payout process

OMN: or Oracle Masternode, retrieve real world sporting event data and post the Blockchain

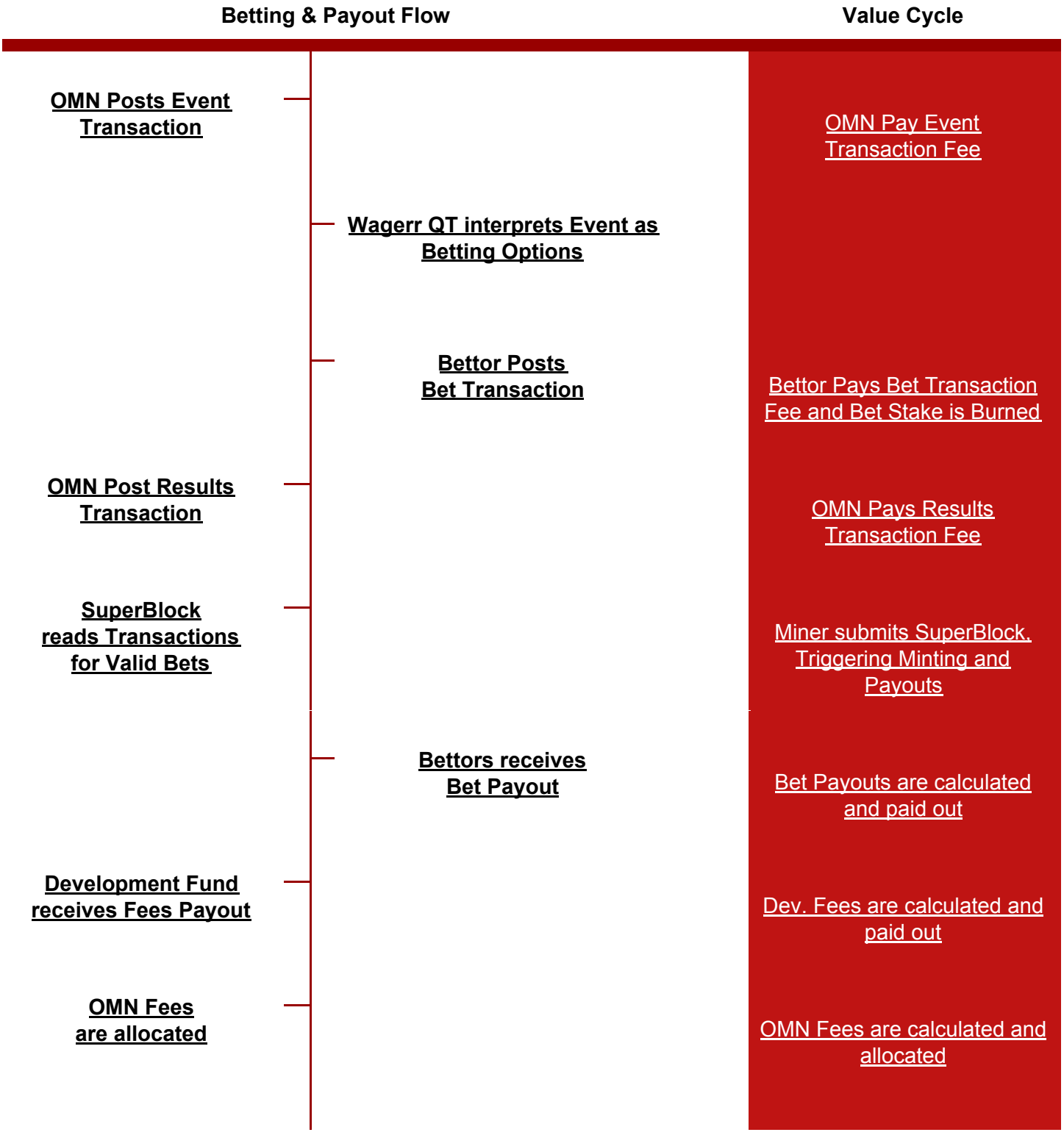
Peerless Direct Chain Betting: or ‘Peerless’ betting is when a Bettor bets directly against the Blockchain

Result Transaction: a Transaction on chain holding the Results of a Betting Event

SuperBlock: a block that processes payments and fees for winning bets and occurs every 1440 blocks (or 24 hours)

Transaction Fee: refers to the amount paid to the network to post a transaction on chain

Betting & Payout Process



Back to Flow Table

Bet Cycle & Payout

OMN Posts Event Transaction

- A trusted OMN posts an Event Transaction to the Blockchain
- The details of this Transaction are then broadcast to the network

EVENT TRANSACTION OP_RETURN FORMAT

TXType|PV|EventId|Timestamp|EventLeague|EventInfo|Home|Away|HomeOdds|AwayOdds|DrawOdds
1|1.0|#453|1528992000|WCUP|R1|RUS|KSA|1.33|11|4.5

HEX Encoded data
317C233435337C31353238393923030307C574355507C52317C5255537C4B53417C312E33337C31317C342E35

EVENT OP RETURN FORMAT DETAILS

TX Type -> the type of wager OP_RETURN transaction.

- EVENT TRANSACTION = 1
- BET TRANSACTION = 2
- RESULT TRANSACTION = 3

PV –> Op Return protocol version number

- Current version is: 1.0

Event Id -> The unique identifier for the event.

Timestamp -> The unix timestamp for the start of the event.

Event League -> The current league or tournament the event is being played under.

Event Info -> Extra event info like the current round of the tournament.

Home -> The team that is playing at home.

Away -> The team that is playing at home.

Home Odds -> The odds for the home team to win the match.

Away Odds -> The odds for the away team to win the match.

Draw Odds -> The odds for the match to be a draw.

Sample raw Event Transaction JSON:


```
{
    ...
    "vin": [{
        "txid": "69dd4f1c690e96c0854196dc37ba8c19fffd8b13b0ca3751a47af1438ad2f7d6",
        "vout": 0,
        "scriptSig": {
            "asm": "REMOVED",
            "hex": "REMOVED"
        },
```


Back to Flow Table

```
        "sequence": 4294967295
    },
    "vout": [{
        "value": 0.01,
        "n": 0,
        "scriptPubKey": {
            "asm": "OP_DUP OP_HASH160 173b3563dd9bc8203fa2c93bfbce83551c8cba6a",
            "hex": "76a914173b3563dd9bc8203fa2c93bfbce83551c8cba6a88ac",
            "reqSigs": 1,
            "type": "pubkeyhash",
            "addresses": ["TVASr4bm6Rz19udhUWmSGtrrDExCjQdATp"]
        }
    },
    {
        "value": 23.45,
        "n": 1,
        "scriptPubKey": {
            "asm": "OP_DUP OP_HASH160 b8cb2971f083c23a26b3a50102f998b1597a6257",
            "hex": "76a914b8cb2971f083c23a26b3a50102f998b1597a625788ac",
            "reqSigs": 1,
            "type": "pubkeyhash",
            "addresses": ["TBZiSM2JMSStYHScwP53D6tsD12mdBhmfM "]
        }
    },
    {
        "value": 0,
        "n": 2,
        "scriptPubKey": {
            "asm": "OP_RETURN 317C233435337C31353238393932303030307C574355507C52317C5255537C4
                    B53417C312E33337C31317C342E35",
            "hex": "6a25317c31353238393932303030307c574355507c52317c5255537c4b53417c312e333",
            "type": "nulldata"
        }
    }],
    ...
}
```


Wagerr QT interprets Event as Betting Options

- The Daemon & QT can interpret and present Betting Options to the user via the Betting tab in the QT or by running the command `wagerr-cli listevents` in the command line interface.

Development Teaser

Bettor Posts Bet Transaction

- A Bettor can place a Bet using command `wagerr-cli placebet "<event-id>" "<team>" <amount>` line or the betting interface, e.g. `wagerr-cli placebet "a1b2" "RUS" 10.5`.
- The details of this Bet transaction are then broadcast to the network

BET TRANSACTION OP_RETURN FORMAT

TXType|PV|EventId|Bet
2|1.0|#453|RUS

Back to Flow Table

Hex Encoded data
327C312E307C233435337C525553

EVENT OP RETURN FORMAT DETAILS

TX Type -> the type of wager OP_RETURN transaction.

- EVENT TRANSACTION = 1
- BET TRANSACTION = 2
- RESULT TRANSACTION = 3

PV –> Op Return protocol version number

- Current version is: 1.0

Event Id -> The unique identifier for the event.

Bet -> The users team to win e.g RUS or KSA or D for a draw

Sample raw Bet Transaction JSON:


```
{
    ….
    "Vin": [{
        "prev_out": {
            "hash": "0767b76406dbaa95cc12d8196196a9e476c81dd328a07b30954d8de256aa1e9f",
            "n": 0
        },
        "scriptSig": "REMOVED"
    }],
   "vout": [{
        "value": "1.00000000",
        "scriptPubKey": "OP_RETURN 327C312E307C233435337C525553"
    },
    {
        "value": 8.899,
        "n": 1,
        "scriptPubKey": {
            "asm": "OP_DUP OP_HASH160 b8cb2971f083c23a26b3a50102f998b1597a6257",
            "hex": "76a914b8cb2971f083c23a26b3a50102f998b1597a625788ac",
            "reqSigs": 1,
            "type": "pubkeyhash",
            "addresses": ["TPbzGC7nGG7yDeRFNtJfopefoXWaAiTQNc"]
        }
    }]
    ….
}
```


OMN Post Results Transaction

- A Trusted Oracle Masternode Address Posts a Result Transaction to the Blockchain
- The details of this Transaction are then broadcast to the network

RESULT TRANSACTION OP_RETURN FORMAT

TXType|PV|EventID|Result|Bet
3|1.0|#453|RUS

Hex Encoded data
337C312E307C233435337C525553

[Back to Flow Table]

RESULT OP RETURN FORMAT DETAILS

TX Type -> the type of wager OP_RETURN transaction.

- EVENT TRANSACTION = 1
- BET TRANSACTION = 2
- RESULT TRANSACTION = 3

PV –> Op Return protocol version number

- Current version is: 1.0

Event Id -> The unique identifier for the event.

Bet -> The users team to win e.g RUS or KSA or D for a draw

Sample raw Result Transaction JSON:


```
{
    ...
    "vin": [{
        "txid": "69dd4f1c690e96c0854196dc37ba8c19fffd8b13b0ca3751a47af1438ad2f7d6",
        "vout": 0,
        "scriptSig": {
            "asm": "REMOVED",
            "hex": "REMOVED"
        },
        "sequence": 4294967295
    },
    "vout": [{
        "value": 0.001,
        "n": 0,
        "scriptPubKey": {
            "asm": "OP_DUP OP_HASH160 173b3563dd9bc8203fa2c93bfbce83551c8cba6a",
            "hex": "76a914173b3563dd9bc8203fa2c93bfbce83551c8cba6a88ac",
            "reqSigs": 1,
            "type": "pubkeyhash",
            "addresses": ["TVASr4bm6Rz19udhUWmSGtrrDExCjQdATp"]
        }
    },
    {
        "value": 9.998,
        "n": 1,
        "scriptPubKey": {
            "asm": "OP_DUP OP_HASH160 b8cb2971f083c23a26b3a50102f998b1597a6257",
            "hex": "76a914b8cb2971f083c23a26b3a50102f998b1597a625788ac",
            "reqSigs": 1,
            "type": "pubkeyhash",
            "addresses": ["TV8n3axPXiYZteRVcCT6TnKM5cffg6FWcy"]
        }
    },
    {
        "value": 0,
        "n": 2,
        "scriptPubKey": {
            "asm": "OP_RETURN 337c312e307c233435337c525553",
            "hex": "6a25317c3135323839393230307c574355507c52317c5255537c4b53417c312e333",
            "type": "nulldata"
        }
```


[Back to Flow Table]

```
    }],
...
}
```

SuperBlock reads Transactions for Valid Bets

- A SuperBlock occurs every 1440 blocks (or 24 hours)
- The SuperBlock reads all past Transaction until/since the previous SuperBlock for Events, Bets and Results
- The SuperBlock compares all Bets to the corresponding Results to determine which bets qualify for Payouts and calculates the Bet Payouts (click here for calculation details)
- The SuperBlock calculates the Betting Fees based off the Bet Payouts (click here for calculation details)

Bettors receives Bet Payout

- Freshly Minted Coins are sent to the address the Bettor used to create a Valid Bet

Development Fund receives Fees Payout

- Click here for calculation details

OMN Fees are allocated

- Click here for calculation details

Value Cycle Explained

OMN Pay Event Transaction Fee

- The cost of posting an Event is covered by the Event Poster

Bettor Pays Bet Transaction Fee and Bet Stake is Burned

- The cost of posting a Bet is covered by the Bettor
- 100% of the Coins Bet are Burned, thus reducing the number of Coins in circulation

OMN Pays Results Transaction Fee

- The cost of posting a Result is covered by the Result Poster

Back to Flow Table

Miner/Staker submits SuperBlock, Triggering Minting and Payouts

- The SuperBlock acts as described above
- Total Coins Minted = Bet Payouts + Dev Fees + OMN Fees

Bet Payouts are calculated and paid out

- (Bet Stake)*(Odds)= Gross Bet Payout
- (Bet Stake)*(0.06)= Betting Fees
- (Gross Bet Payout) - (Betting Fees)= Bet Payout

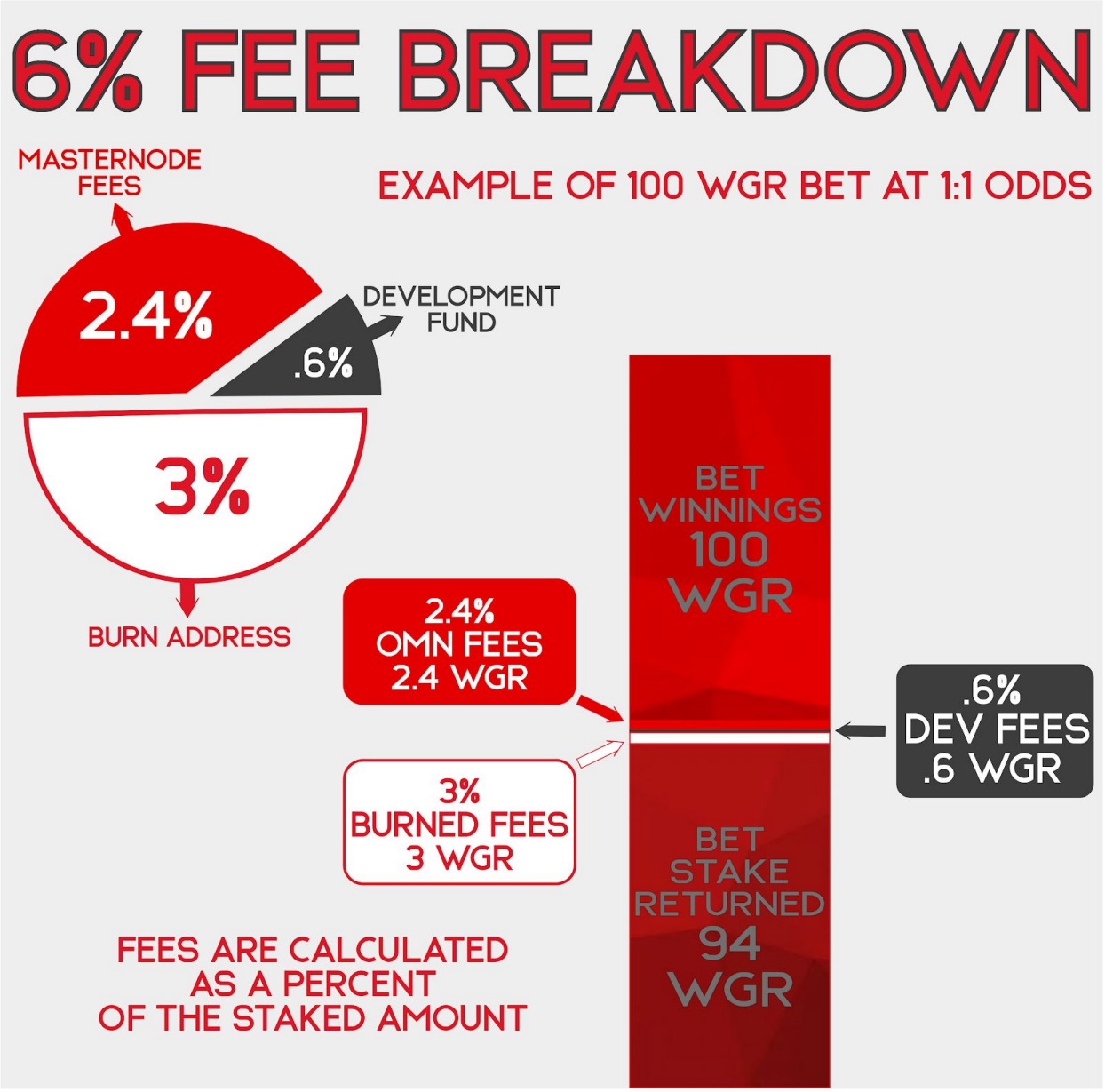
Dev Fees are calculated and paid out

- The Development fund receives 10% of the total Fees deducted from the Bet Payouts
- The current development fund address is Wm5om9hBJTyKqv5FkMSfZ2FDMeGp12fkTe
- These Coins are sent to a specific wallet in the control of the Wagerr team
- (Bet Stake)*(Odds)= Gross Bet Payout
- (Bet Stake)*(0.06)= Betting Fees
- (Betting Fees)*(0.10)= Development Fees

OMN Fees are calculated and allocated

- The OMNs receive 40% of the total Fees deducted from the Bet Payouts
- These Coins are spread evenly among the block rewards of the next 1440 blocks (up to the next SuperBlock)
- This ensures that all OMNs receive their fair share of the reward
- (Bet Stake)*(Odds)= Gross Bet Payout
- (Bet Stake)*(0.06)= Betting Fees
- (Betting Fees)*(0.4)= OMN Fees

Back to Flow Table



Back to Flow Table