A Monero Data Model and API Specification

with implementations in Java, JavaScript, and C++
using monero v0.16.0.0 'Nitrogen Nebula'

Last revised June 04, 2020

Java implementation: https://github.com/monero-ecosystem/monero-java

JavaScript implementation: https://github.com/monero-ecosystem/monero-javascript

C++ implementation: https://github.com/monero-ecosystem/monero-cpp

Document source: https://github.com/monero-ecosystem/monero-java/blob/master/docs/monero-spec.drawio

Document created and revisable using: https://www.draw.io/

Monero Daemon Interface 1/2

isConnected(): bool

getVersion(): MoneroVersion

isTrusted(): bool

getHeight(): ulong

getBlockHash(ulong height): string

getBlockTemplate(string walletAddress, uint reserveSize=null): MoneroBlockTemplate

getLastBlockHeader(): MoneroBlockHeader

getBlockHeaderByHash(string blockHash): MoneroBlockHeader

getBlockHeaderByHeight(ulong height): MoneroBlockHeader

getBlockHeadersByRange(ulong startHeight=0, ulong endHeight=chainHeight): MoneroBlockHeader[]

getBlockByHash(string blockHash): MoneroBlock

getBlocksByHash(string[] blockHashes, ulong startHeight, bool prune=false): MoneroBlock[]

getBlockByHeight(ulong height): MoneroBlock

getBlocksByHeight(ulong[] heights): MoneroBlock[]

getBlocksByRange(ulong startHeight=0, ulong endHeight=chainHeight): MoneroBlock[]

getBlocksByRangeChunked(ulong startHeight=0, ulong endHeight=chainHeight, ulong maxChunkSize=3000000): MoneroBlock[]

getBlockHashes(string[] blockHashes, ulong startHeight): string[]

getTx(string txHash, bool prune=false): MoneroTx

getTxs(string[] txHashes, bool prune=false): MoneroTx[]

getTxHex(string txHash, bool prune=false): string

getTxHexes(string[] txHashes, bool prune=false): string[]

getMinerTxSum(ulong height, ulong numBlocks=chainHeight): MoneroMinerTxSum

getFeeEstimate(ulong graceBlocks=?): ulong

submitTxHex(string txHex, bool doNotRelay=false): MoneroSubmitTxResult

relayTxByHash(string txHash): void

relayTxsByHash(string[] txHashes): void

getTxPool(): MoneroTx[]

getTxPoolHashes(): string[]

getTxPoolBacklog(): MoneroTxBacklogEntry[]

getTxPoolStats(): MoneroTxPoolStats

flushTxPool(string[] txHashes=null): void

getKeyImageSpentStatus(string keyImage): MoneroKeyImageSpentStatus

getKeyImageSpentStatuses(string[] keyImages): MoneroKeyImageSpentStatus[]

getOutputs(MoneroOutput[] outputs): MoneroOutput[]

getInfo(): MoneroDaemonInfo

getSyncInfo(): MoneroDaemonSyncInfo

getHardForkInfo(): MoneroHardForkInfo

Monero Daemon Intereface 2/2

getAltChains(): MoneroAltChain[]

getAltBlockHashes(): string[]

getDownloadLimit(): uint

setDowloadLimit(uint limit): uint

resetDownloadLimit(): uint

getUploadLimit(): uint

setUploadLimit(uint): uint

resetUploadLimit(): uint

getKnownPeers(): MoneroDaemonPeer[]

getConnections(): MoneroDaemonConnection[]

setOutgoingPeerLimit(uint limit): void

setIncomingPeerLimit(uint limit): void

getPeerBans(): MoneroBan[]

setPeerBan(MoneroBan ban): void

setPeerBans(MoneroBan[] bans): void

getOutputDistribution(ulong[] amounts, bool isCumulative, ulong startHeight=0, ulong endHeight=chainHeight):
MoneroOutputDistributionEntry[]

getOutputHistogram(ulong[] amounts, ulong minCount, ulong maxCount, bool isUnlocked, ulong recentCutoff):
MoneroOutputHistogramEntry[]

startMining(string address, ulong numThreads=null, bool isBackground=false, bool ignoreBattery=false): void

stopMining(): void

getMiningStatus(): MoneroMiningStatus

submitBlock(string blockBlob): void

submitBlocks(string[] blockBlobs): void

checkForUpdate(): MoneroDaemonUpdateCheckResult

downloadUpdate(string path): MoneroDaemonUpdateDownloadResult

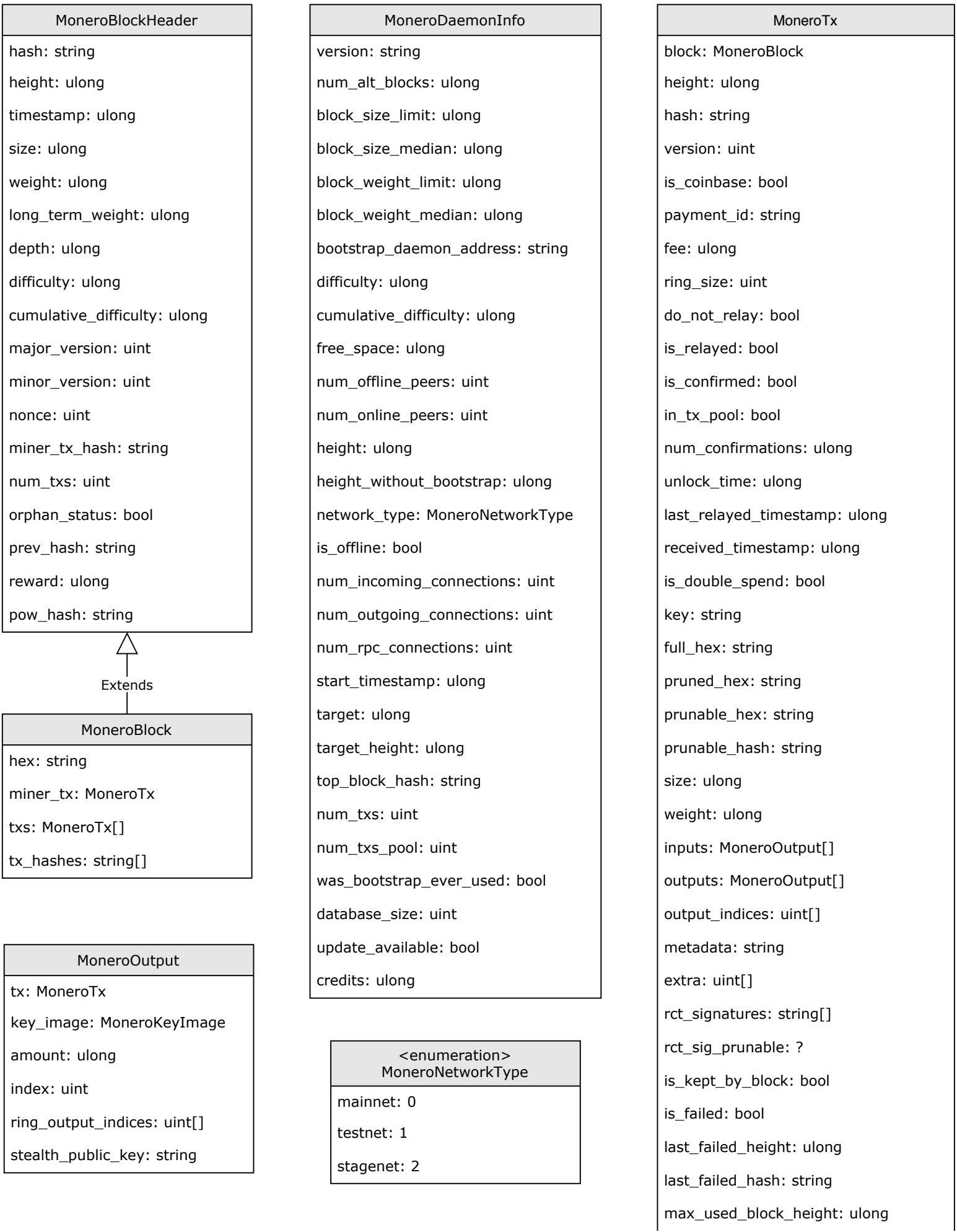
getNextBlockHeader(): MoneroBlockHeader

addListener(MoneroDaemonListener listener): void

removeListener(MoneroDaemonListener listener): void

stop(): void

Monero Daemon Types 1/3



Monero Daemon Types 2/3

MoneroDaemonSyncInfo
height: ulong
connections: MoneroDaemonConnection[]
spans: MoneroDaemonConnectionSpan[]
target_height: ulong
next_needed_pruning_seed: uint
overview: ?

MoneroDaemonConnection
peer: MoneroDaemonPeer
id: string
avg_download: ulong
avg_upload: ulong
current_download: ulong
current_upload: ulong
height: ulong
is_incoming: bool
live_time: ulong
is_local_ip: bool
is_local_host: bool
num_receives: uint
num_sends: uint
receive_idle_time: ulong
send_idle_time: ulong
state: string
num_support_flags: uint
type: ConnectionType

MoneroDaemonConnectionSpan
connection_id: string
num_blocks: ulong
remote_address: string
rate: ulong
speed: ulong
size: ulong
start_block_height: ulong

MoneroDaemonListener
onBlockHeader(MoneroBlockHeader header): void

MoneroKeyImage
hex: string
signature: string

<enumeration> MoneroKeyImageSpentStatus
not_spent: 0
confirmed: 1
tx_pool: 2

MoneroSubmitTxResult
is_good: bool
is_relayed: bool
is_double_spend_seen: bool
is_fee_too_low: bool
is_mixin_too_low: bool
has_invalid_input: bool
has_invalid_output: bool
has_too_few_outputs: bool
is_rct: bool
is_overspend: bool
is_too_big: bool
sanity_check_failed: bool
reason: string
credits: ulong
top_block_hash: string

MoneroDaemonPeer
id: string
address: string
host: string
port: uint
is_online: boolean
last_seen_timestamp: ulong
pruning_seed: uint
rpc_port: uint
rpc_credits_per_hash: ulong

MoneroBan
host: string
ip: string
is_banned: bool
seconds: ulong

MoneroBlockTemplate
block_template_blob: string
block_hashing_blob: string
difficulty: ulong
expected_reward: ulong
height: ulong
prev_hash: string
reserved_offset: ulong
seed_height: ulong
seed_hash: string
next_seed_hash: string
max_used_block_hash: string
signatures: string[]

Monero Daemon Types 3/3

MoneroHardForkInfo
earliest_height: ulong
is_enabled: bool
state: uint
threshold: uint
version: string
num_votes: uint
window: uint
voting: uint
credits: ulong
top_block_hash: string

MoneroDaemonUpdateCheckResult
is_update_available: bool
version: string
hash: string
auto_uri: string
user_uri: string

Extends

MoneroDaemonUpdateDownloadResult
download_path: string

MoneroOutputHistogramEntry
amount: ulong
num_instances: ulong
num_unlocked_instances: ulong
num_recent_instances: ulong

MoneroTxPoolStats
num_txs: uint
num_not_relayed: uint
num_failing: uint
num_double_spends: uint
num_10m: uint
fee_total: ulong
bytes_max: ulong
bytes_med: ulong
bytes_min: ulong
bytes_total: ulong
histo: ?
histo_98pc: unlong
oldest_timestamp: ulong

MoneroVersion
number: uint
is_release: bool

MoneroMinerTxSum
emission_sum: ulong
fee_sum: ulong

<enumeration> ConnectionType
invalid: 0
ipv4: 1
ipv6: 2
tor: 3
i2p: 4

MoneroMiningStatus
is_active: bool
is_background: bool
address: string
speed: ulong
num_threads: uint

MoneroAltChain
block_hashes: string[]
difficulty: ulong
height: ulong
length: ulong
main_chain_parent_block_hash: string

Monero Wallet Interface 1/3

isViewOnly(): bool

setDaemonConnection(MoneroDaemonConnection connection): void

getDaemonConnection(): MoneroDaemonConnection

isConnected(): bool

getVersion(): MoneroVersion

getNetworkType(): MoneroNetworkType

getPath(): string

getMnemonic(): string

getMnemonicLanguage(): string

getMnemonicLanguages(): string[]

getPrivateSpendKey(): string

getPrivateViewKey(): string

getPublicSpendKey(): string

getPublicViewKey(): string

getPrimaryAddress(): string

getAddress(uint accountIdx, uint subaddressIdx): string

getAddressIndex(string address): MoneroSubaddress

getDaemonConnection(): MoneroRpcConnection

getHeight(): ulong

getDaemonHeight(): ulong

getDaemonMaxPeerHeight(): ulong

getApproximateChainHeight(): ulong

sync(ulong startHeight=null, MoneroSyncListener listener=null): MoneroSyncResult

startSyncing(): void

stopSyncing(): void

rescanSpent(): void

rescanBlockchain(): void

getBalance(uint accountIdx=null, uint subaddressIdx=null): ulong

getUnlockedBalance(uint accountIdx=null, uint subaddressIdx=null): ulong

getAccounts(bool includeSubaddresses=false, string tag=null): MoneroAccount[]

getAccount(uint accountIdx, bool includeSubaddresses=false): MoneroAccount

createAccount(string label=null): void

getSubaddresses(uint accountIdx, uint[] subaddressIndices): void

getSubaddress(uint accountIdx, uint subaddressIdx): MoneroSubaddress

createSubaddress(uint accountIdx, string label=null): MoneroSubaddress

getTx(string txId): MoneroTxWallet

getTxs(MoneroTxQuery query=null): MoneroTxWallet[]

getTransfers(uint accountIdx=null, uint subaddressIdx=null): MoneroTransfer[]

Monero Wallet Interface 2/3

getOutputs(MoneroOutputQuery query=null): MoneroOutputWallet[]

getOutgoingTransfers(MoneroTransferQuery query=null): MoneroOutgoingTransfer[]

getIncomingTransfers(MoneroTransferQuery query=null): MoneroIncomingTransfer[]

getTransfers(MoneroTransferQuery query=null): void

getKeyImages(): MoneroKeyImage[]

importOutputsHex(string outputsHex): uint

getOutputsHex(): string

importKeyImages(MoneroKeyImage[] keyImages): MoneroKeyImageImportResult

getNewKeyImagesFromLastImport(): MoneroKeyImage[]

createTx(MoneroTxConfig config): MoneroTxWallet

createTxs(MoneroTxConfig config): MoneroTxWallet[]

sweepOutput(MoneroTxConfig config): MoneroTxWallet

sweepUnlocked(MoneroTxConfig config): MoneroTxWallet[]

sweepDust(bool relay=false): MoneroTxWallet[]

relayTx(string txMetadata): string

relayTx(MoneroTxWallet tx): string

relayTxs(string[] txMetadatas): string[]

relayTxs(MoneroTxWallet[] txs): string[]

parseTxSet(MoneroTxSet txSet): MoneroTxSet

signTxs(string unsignedTxHex): string

submitTxs(string signedTxHex): string[]

signMessage(string message): string

verifyMessage(string message, string address, string signature): bool

getTxKey(string txId): string

checkTxKey(string txId, string txKey, string address): MoneroCheckTx

getTxProof(String txId, string address, string message=null): string

checkTxProof(string txId, string address, string message, string signature): MoneroCheckTx

getSpendProof(string txId, string message=null): string

checkSpendProof(string txId, string message, string signature): bool

Monero Wallet Interface 3/3

getReserveProofWallet(string message): string

getReserveProofAccount(uint accountIdx, ulong amount, string message): string

checkReserveProof(string address, string message, string signature): MoneroCheckReserve

setTxNotes(string[] txHashes, string[] notes): void

setTxNote(string txHash, string note): void

getTxNotes(string[] txHashes): string[]

getTxNote(string txHash): string

getAddressBookEntries(uint[] entryIndices=null): MoneroAddressBookEntry[]

addAddressBookEntry(string address, string description): uint

editAddressBookEntry(uint entryIdx, bool setAddress, string address, bool setDescription, string description): void

deleteAddressBookEntry(uint entryIdx): void

tagAccounts(string tag, uint[] accountIndices): void

untagAccounts(uint[] accountIndices): void

getAccountTags(): MoneroAccountTag[]

setAccountTagLabel(string tag, string label): void

createPaymentUri(MoneroSendRequest request): string

parsePaymentUri(string uri): MoneroSendRequest

getAttribute(string key): string

setAttribute(string key, string val): void

startMining(uint numThreads=null, bool backgroundMining=false, bool ignoreBattery=true): void

stopMining(): void

isMultisigImportNeeded(): bool

isMultisig(): bool

getMultisigInfo(): MoneroMultisigInfo

prepareMultisig(): string

makeMultisig(string[] multisigHexes, uint threshold, string password): MoneroMultisigInitResult

exchangeMultisigKeys(string[] multisigHexes, string password): MoneroMultisigInitResult

getMultisigHex(): string

importMultisigHex(string[] multisigHexes): uint

signMultisigTxHex(string multsiigTxHex): MoneroMultisigSignResult

submitMultisig(string signedMultisigHex): string[]

moveTo(string path, string password): void

save(): void

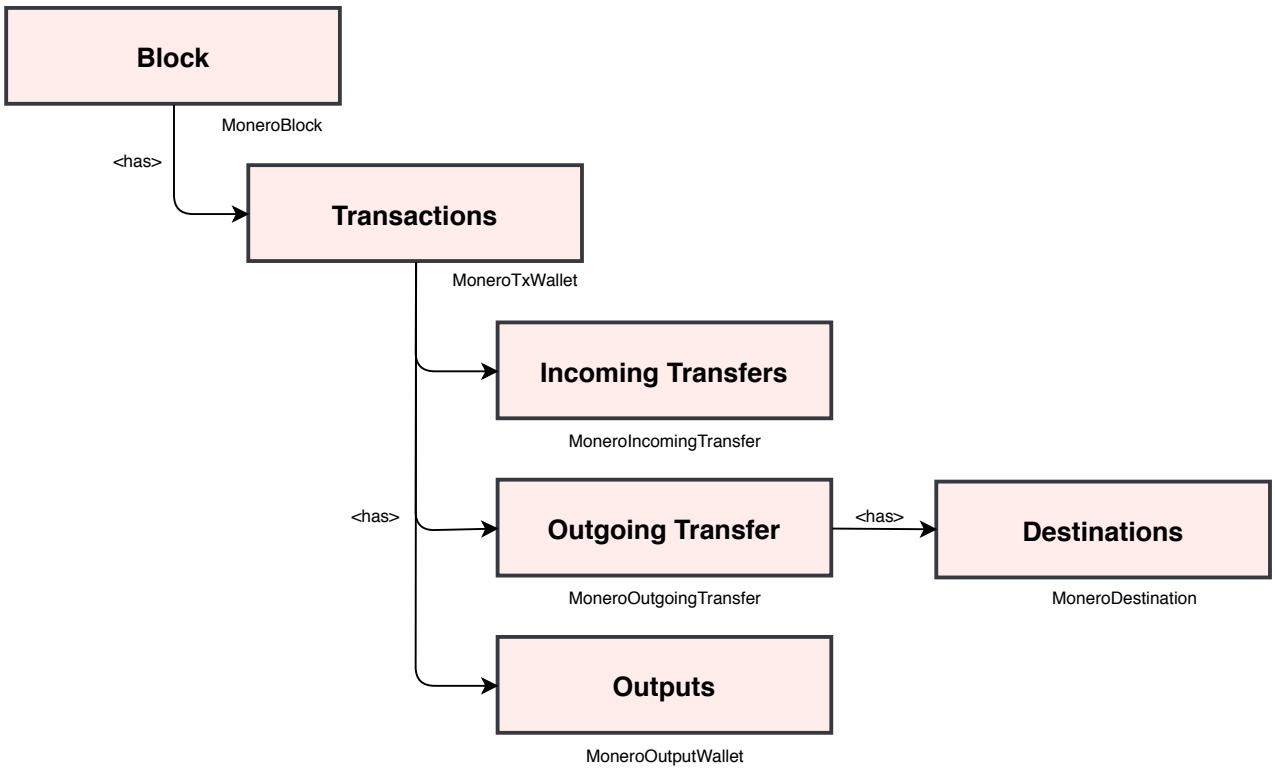
close(bool save=false): void

isClosed(): bool

getIntegratedAddress(string paymentId=null): MoneroIntegratedAddress (deprecated)

decodeIntegratedAddress(string integratedAddress): MoneroIntegratedAddress (deprecated)

Monero Wallet Types 1/4 - Transactions, Transfers, and Outputs



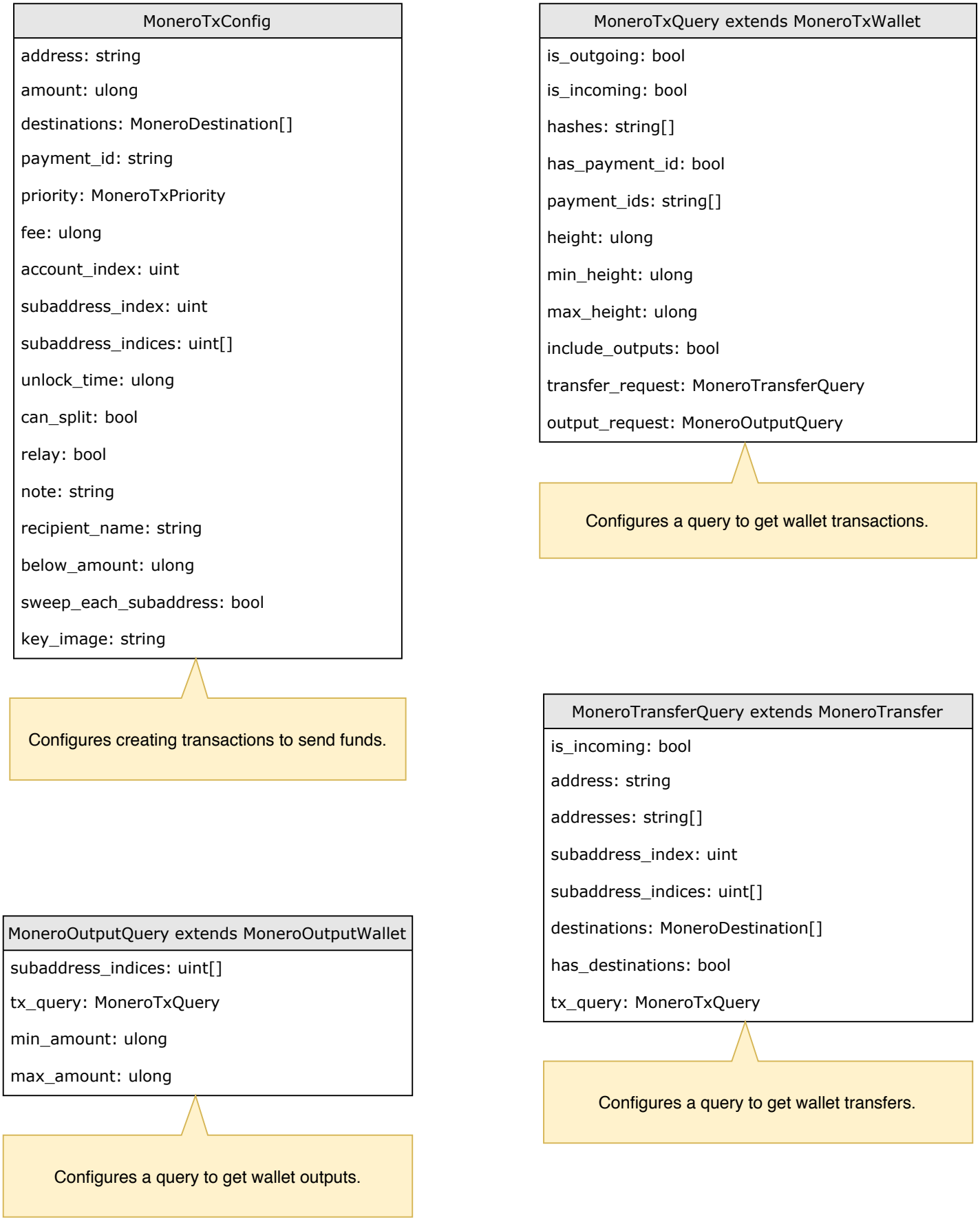
Transactions are part of a block.

In the context of a wallet, transactions have incoming transfers and/or an outgoing transfer.

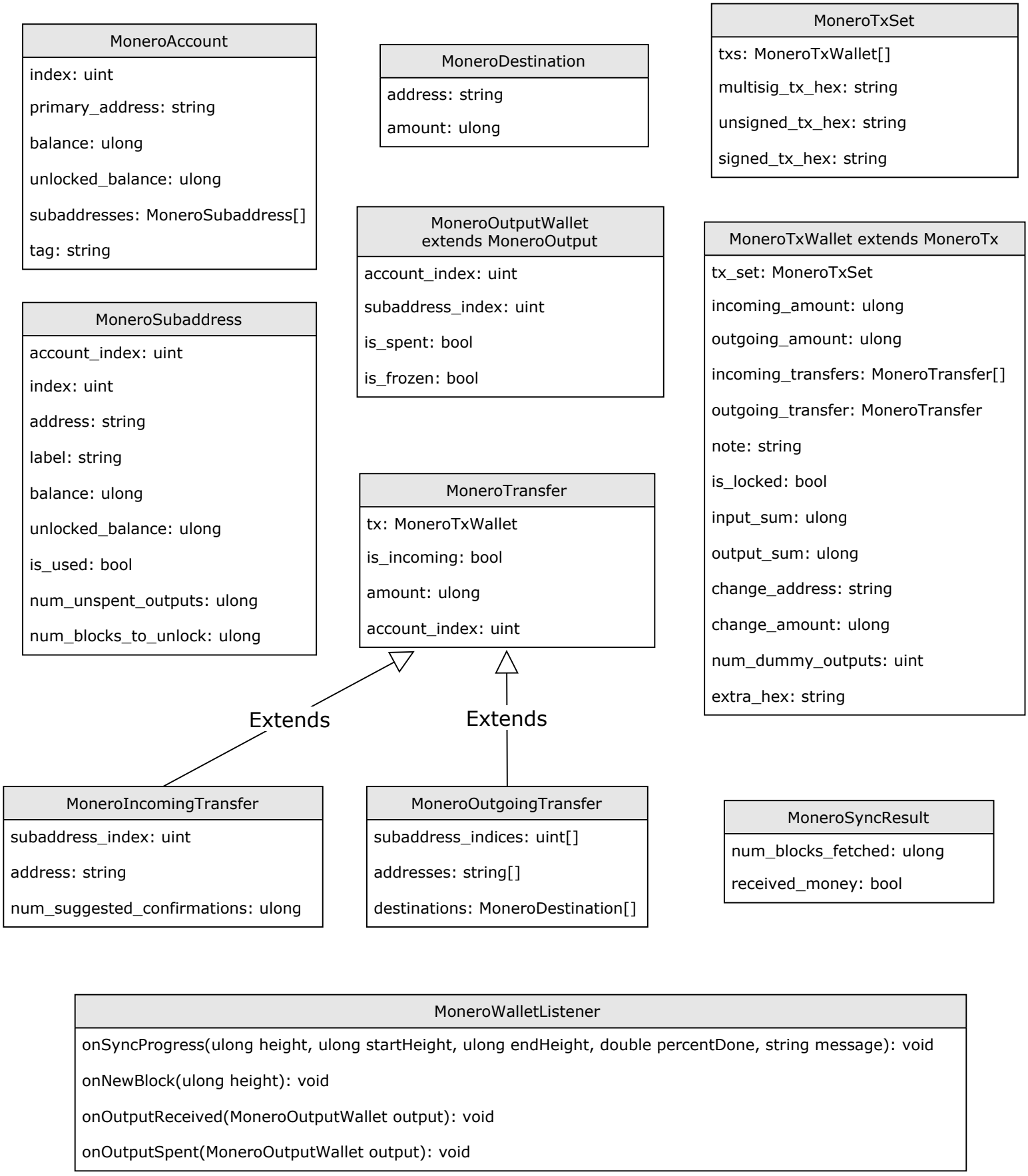
The outgoing transfer may have destination amounts and addresses which are saved locally with the wallet and are not recoverable from the blockchain.

Transactions can also have outputs received by the wallet.

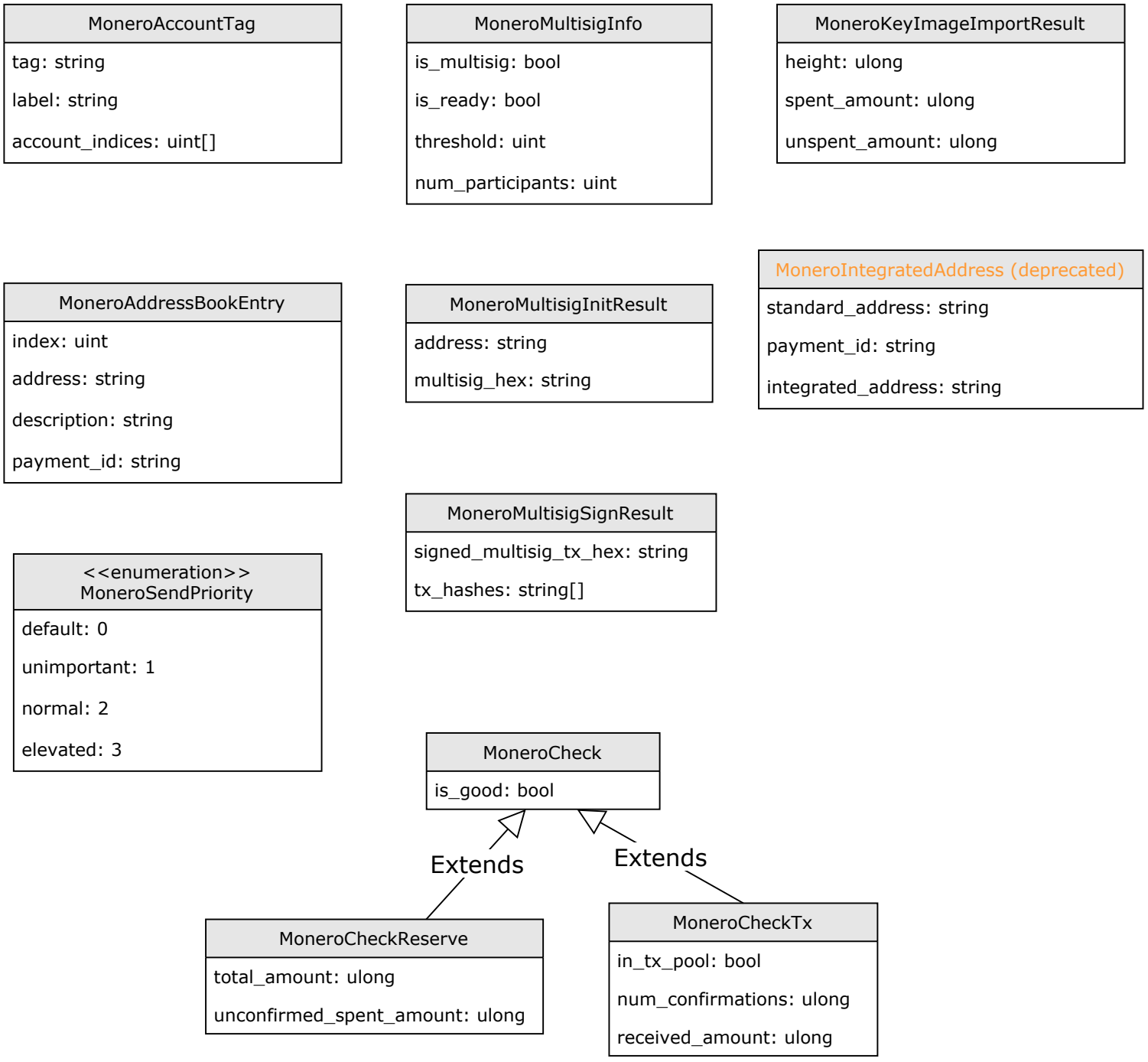
Monero Wallet Types 2/4 - Send Request and Queries for Transactions, Transfers, and Outputs

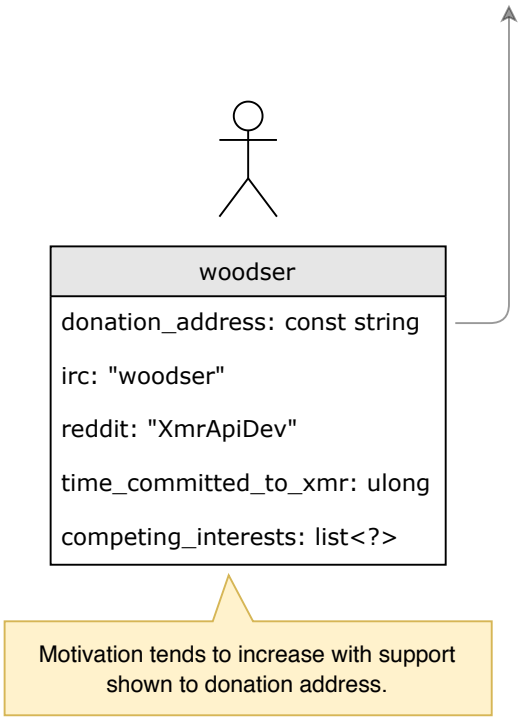


Monero Wallet Types 3/4



Monero Wallet Types 4/4



woodser API 1/1

46FR1GKVqFNQnDiFkH7AuzbUBrGQwz2VdaXTDD4jcjRE8YkkoTYTmZ2Vohsz9gLSqkj5EM6ai9Q7sBoX4FPPYJdGKQQXPVz