



Overview of

Monero Wallet and Daemon Interfaces and Types

based on Monero Core v0.14.0.3

Revision 12 - April 26, 2019

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

Document source: <https://github.com/monero-ecosystem/monero-javascript/blob/master/monero-model.xml>

Document source created and modifiable using: <https://www.draw.io/>

Daemon Interface 1/2

MoneroDaemon
getIsTrusted(): boolean
getHeight(): uint
getBlockId(height): string
getBlockTemplate(walletAddress, reserveSize): MoneroBlockTemplate
getLastBlockHeader(): MoneroBlockHeader
getBlockHeaderById(blockId): MoneroBlockHeader
getBlockHeaderByHeight(height): MoneroBlockHeader
getBlockHeadersByRange(startHeight, endHeight): list<MoneroBlockHeader>
getBlockById(blockId): MoneroBlock
getBlocksById(blockIds, startHeight, prune): list<MoneroBlock>
getBlockByHeight(height): MoneroBlock
getBlocksByHeight(heights): list<MoneroBlock>
getBlocksByRange(startHeight, endHeight): list<MoneroBlock>
getBlockIds(blockIds, startHeight): list<string>
getTx(txId, prune): MoneroTx
getTxs(txIds, prune): list<MoneroTx>
getTxHex(txId, prune): string
getTxHexes(txIds, prune): list<string>
getCoinbaseTxSum(height, numBlocks): MoneroCoinbaseTxSum
getFeeEstimate(graceBlocks): BigInteger
submitTxHex(txHex, doNotRelay): MoneroSubmitTxResult
relayTxById(txId): string
relayTxById(txIds): list<string>
getTxPool(): list<MoneroTx>
getTxPoolIds(): list<string>
getTxPoolBacklog(): list<MoneroTxBacklogEntry>
getTxPoolStats(): MoneroTxPoolStats
flushTxPool(ids): list<string>
getSpentStatus(keyImage): MoneroKeyImageSpentStatus
getSpentStatuses(keyImages): list<MoneroKeyImageSpentStatus>
getOutputs(outputs): list<MoneroOutput>
getOutputHistogram(amounts, minCount, maxCount, isUnlocked, recentCutoff): list<MoneroOutputHistogramEntry>
getOutputDistribution(amounts, cumulative, startHeight, endHeight): list<MoneroOutputDistributionEntry>
getInfo(): MoneroDaemonInfo
getSyncInfo(): MoneroDaemonSyncInfo
getHardForkInfo(): MoneroDaemonHardForkInfo
getAltChains(): list<MoneroAltChain>
getAltBlockIds(): list<string>

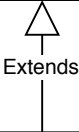
```

getDownloadLimit(): uint
setDownloadLimit(limit): uint
resetDownloadLimit(): uint
getUploadLimit(): uint
setUploadLimit(limit): uint
resetUploadLimit(): uint
getKnownPeers(): list<MoneroDaemonPeer>
getConnections(): list<MoneroDaemonConnection>
setOutgoingPeerLimit(limit): void
setIncomingPeerLimit(limit): void
getPeerBans(): list<MoneroBan>
setPeerBan(ban): void
setPeerBans(bans): void
startMining(address, numThreads, isBackground, ignoreBattery): void
stopMining(): void
getMiningStatus(): MoneroMiningStatus
submitBlock(blockBlob): void
submitBlocks(blockBlobs): void
checkForUpdate(): MoneroDaemonUpdateCheckResult
downloadUpdate(path): MoneroDaemonDownloadUpdateResult
stop(): void

```

Daemon Types 1/3

MoneroBlockHeader
id: string
height: uint
timestamp: long
size: long
weight: long
long_term_weight: uint
depth: long
difficulty: BigInteger
cumulative_difficulty: BigInteger
major_version: uint
minor_version: uint
nonce: long
coinbase_tx_id: string
num_txs: uint
orphan_status: boolean
prev_id: string
reward: BigInteger
pow_hash: string



num
hex: string
coinbase_tx: MoneroTx
txs: list<MoneroTx>
tx_ids: list<string>

MoneroOutput
tx: MoneroTx
key_image: MoneroKeyImage
amount: BigInteger
index: uint
ring_output_indices: list<uint>
stealth_public_key: string

MoneroDaemonInfo
version: string
num_alt_blocks: uint
block_size_limit: uint
block_size_median: uint
block_weight_limit: uint
block_weight_median: uint
bootstrap_daemon_address: string
difficulty: BigInteger
cumulative_difficulty: uint
free_space: uint
num_offline_peers: uint
num_online_peers: uint
height: uint
height_without_bootstrap: uint
network_type: MoneroNetworkType
is_offline: boolean
num_incoming_connections: uint
num_outgoing_connections: uint
num_rpc_connections: uint
start_timestamp: uint
target: uint
target_height: uint
top_block_id: string
num_txs: uint
num_txs_pool: uint
was_bootstrap_ever_used: boolean
database_size: uint
update_available: boolean

MoneroTx
block: MoneroBlock
height: uint
id: string
version: uint
is_coinbase: boolean
payment_id: string
fee: BigInteger
mixin: uint
do_not_relay: boolean
is_relayed: boolean
is_confirmed: boolean
in_tx_pool: boolean
num_confirmations: uint
unlock_time: uint
last_relayed_timestamp: uint
received_timestamp: uint
is_double_spend: boolean
key: string
full_hex: string
pruned_hex: string
prunable_hex: string
prunable_hash: string
size: uint
weight: uint
vins: list<MoneroOutput>
vouts: list<MoneroOutput>
output_indices: list<uint>
metadata: string
common_tx_sets: ?
extra: list<uint>
rct_signatures: list<string>
rct_sig_prunable: TODO
is_kept_by_block: boolean
is_failed: boolean
last_failed_height: uint
last_failed_id: string

Daemon Types 2/3

MoneroDaemonSyncInfo
height: uint
connections: list<MoneroDaemonConnection>
spans: list<MoneroDaemonConnectionSpan>
target_height: uint
next_needed_pruning_seed: uint
overview: ?

MoneroKeyImage
hex: string
signature: string

max_used_block_height: uint
max_used_block_id: string
signatures: list<string>

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

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

MoneroHardForkInfo
earliest_height: uint
is_enabled: boolean
state: uint
threshold: uint
version: string
num_votes: uint
window: uint
voting: uint

MoneroDaemonConnection
peer: MoneroDaemonPeer
id: string
avg_download: uint
avg_upload: uint
current_download: uint
current_upload: uint
height: uint
is_incoming: boolean
live_time: long
is_local_ip: boolean
is_local_host: boolean
num_receives: uint
num_sends: uint
receive_idle_time: long
send_idle_time: long
state: string
num_support_flags: uint

MoneroBan
host: string
ip: string
is_banned: boolean
seconds: long

MoneroSubmitTxResult
is_good: boolean
is_relayed: boolean
is_double_spend: boolean
is_fee_too_low: boolean
is_mixin_too_low: boolean
has_invalid_input: boolean
has_invalid_output: boolean
is_rct: boolean
is_overspend: boolean
is_too_big: boolean
sanity_check_failed
reason: string

<<enumeration>> MoneroNetworkType
mainnet: 0
testnet: 1
stagenet: 2

MoneroBlockTemplate
block_template_blob: string
block_hashing_blob: string
difficulty: BigInteger
expected_reward: BigInteger
height: uint
prev_id: string
reserved_offset: uint

MoneroMiningStatus
is_active: boolean
is_background: boolean
address: string
speed: uint
num_threads: uint

Daemon Types 3/3

MoneroAltChain
block_ids: list<string>
difficulty: BigInteger
height: uint
length: uint
main_chain_parent_block_id: string

MoneroCoinbaseTxSum
emission_sum: BigInteger
fee_sum: BigInteger

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



MoneroDaemonUpdateDownloadResult
download_path: string

MoneroOutputHistogramEntry
amount: BigInteger
num_instances: uint
num_unlocked_instances: uint
num_recent_instances: uint

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

Wallet Interface 1/2

MoneroWallet
getSeed(): string
getMnemonic(): string
getPublicViewKey(): string
getPrivateViewKey(): string
getLanguages(): list<string>
getHeight(): uint
getChainHeight(): uint
getPrimaryAddress(): string
getIntegratedAddress(paymentId): MoneroIntegratedAddress
decodeIntegratedAddress(integratedAddress): MoneroIntegratedAddress
sync(startHeight, endHeight, onProgress): MoneroSyncResult
isMultisigImportNeeded(): boolean
getAccounts(includeSubaddresses): list<MoneroAccount>
getAccount(accountIdx, includeSubaddresses): MoneroAccount
createAccount(label): MoneroAccount
getSubaddresses(accountIdx, subaddressIndices): list<MoneroSubaddress>
getSubaddress(accountIdx, subaddressIdx): MoneroSubaddress
createSubaddress(accountIdx, label): MoneroSubaddress
getAddress(accountIdx, subaddressIdx): string
getAddressIndex(address): uint
getBalance(accountIdx, subaddressIdx): BigInteger
getUnlockedBalance(accountIdx, subaddressIdx): BigInteger
getTxs(config): list<MoneroWalletTx>
getTransfers(config): list<MoneroTransfer>
getVouts(config): list<MoneroWalletOutput>
getKeyImages(): list<MoneroKeyImage>
importKeyImages(keyImages): MoneroKeyImageImportResult
getNewKeyImagesFromLastImport(): list<MoneroKeyImage>
send(configOrAddress, sendAmount, priority, mixin): MoneroWalletTx
sendSplit(configOrAddress, sendAmount, priority, mixin): List<MoneroWalletTx>
sweepWallet(address): list<MoneroWalletTx>
sweepAccount(accountIdx, address): list<MoneroWalletTx>
sweepSubaddress(accountIdx, subaddressIdx, address): list<MoneroWalletTx>
sweepUnlocked(config): list<MoneroWalletTx>
sweepDust(doNotRelay): list<MoneroWalletTx>
sweepOutput(configOrAddress, keyImage, priority, mixin): list<MoneroWalletTx>
relayTx(txMetadata): string
relayTxs(txMetadatas): list<string>

MoneroSendConfig
destinations: list<MoneroDestination>
payment_id: string
priority: MoneroSendPriority
mixin: uint
ring_size: uint
fee: BigInteger
account_index: uint
subaddress_indices: list<uint>
unlock_time: uint
can_split: boolean
do_not_relay: boolean
note: string
recipient_name: string
below_amount: BigInteger
sweep_each_subaddress: boolean
key_image: string

MoneroSendConfig configures outgoing transfers, sweeps, and creation of payment URIs

```

getTxNote(txId): string
setTxNote(txId, note): void
getTxNotes(txIds): list<string>
setTxNotes(txIds, notes): void
sign(msg): string
verify(msg, address, signature): boolean
getTxKey(txId): string
checkTxKey(txId, txKey, address): MoneroCheckTx
getTxProof(txId, address, message): string
checkTxProof(txId, address, message, signature): MoneroCheckTx
getSpendProof(txId, message): string
checkSpendProof(txId, message, signature): boolean
getReserveProofWallet(message): string
getReserveProofAccount(accountIdx, amount, message): string
checkReserveProof(address, message, signature): MoneroCheckReserve
getAddressBookEntries(entryIndices): list<MoneroAddressBookEntry>
addAddressBookEntry(address, description, paymentId): uint
deleteAddressBookEntry(entryIdx): void
getTxProof(txId, address, message): string
tagAccounts(tag, accountIndices): void
untagAccounts(accountIndices): void
getAccountTags(): list<MoneroAccountTag>
setAccountTagLabel(tag, label): void
createPaymentUri(sendConfig): string
parsePaymentUri(uri): MoneroSendConfig
getOutputsHex(): string
importOutputsHex(outputsHex): uint
setAttribute(key, val): void
getAttribute(key): string
startMining(numThreads, backgroundMining, ignoreBattery): void
stopMining(): void

```


Wallet Types 1/1

MoneroAccount
index: uint
primary_address: string
label: string
balance: BigInteger
unlocked_balance: BigInteger
subaddresses: list<MoneroSubaddress>
tag: string

MoneroAccountTag
tag: string
label: string
account_indices: list<int>

MoneroAddressBookEntry
index: uint
address: string
payment_id: string
description: string

MoneroIntegratedAddress
standard_address: string
payment_id: string
integrated_address: string

MoneroSubaddress
account_index: uint
index: uint
address: string
label: string
balance: BigInteger
unlocked_balance: BigInteger
is_used: boolean
num_unspent_outputs: uint
num_blocks_to_unlock: uint

MoneroDestination
address: string
amount: BigInteger

MoneroOutputWallet extends MoneroOutput
account_index: uint
subaddress_index: uint
is_spent: boolean
is_unlocked: boolean
is_frozen: boolean

<<enumeration>> MoneroSendPriority
default: 0
unimportant: 1
normal: 2
elevated: 3

MoneroWalletTx extends MoneroTx
incoming_amount: BigInteger
outgoing_amount: BigInteger
incoming_transfers: list<MoneroTransfer>
outgoing_transfer: MoneroTransfer
num_suggested_confirmations: uint
note: string

MoneroTransfer
tx: MoneroWalletTx
is_outgoing: boolean
is_incoming: boolean
address: string
account_index: uint
subaddress_index: uint
amount: BigInteger
destinations: list<MoneroDestination>

MoneroSyncResult
num_blocks_fetched: uint
received_money: boolean

MoneroKeyImageImportResult
height: uint
spent_amount: BigInteger
unspent_amount: BigInteger

