Sui JSON-RPC Reference - Version:

Sui JSON-RPC API for interaction with Sui Full node. Make RPC calls using https://fullnode.NETWORK.sui.io:443, where NETWORK is the network you want to use (testnet, devnet, mainnet). By default, local networks use port 9000.

Return the total coin balance for all coin type, owned by the address owner.

Parameters

Result

Vec <[Balance

```
]>
```

Example

Gets all balances for the address in the request.

Request

Response

Return all Coin objects owned by an address.

Parameters

Result

CoinPage <

```
Page_for_Coin_and_String
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets all coins for the address in the request body. Begin listing the coins that are after the provided cursor value and return only the limit amount of results per page.

Request

Response

Return the total coin balance for one coin type, owned by the address owner.

Parameters

Result

Balance <

```
Balance
```

```
>
```

Example

Gets the balance of the specified type of coin for the address in the request.

Request

Response

Return metadata (e.g., symbol, decimals) for a coin. Note that if the coin's metadata was wrapped in the transaction that published its marker type, or the latest version of the metadata object is wrapped or deleted, it will not be found.

Parameters

Result

SuiCoinMetadata <

```
SuiCoinMetadata
```

```
>
```

Example

Gets the metadata for the coin type in the request.

Request

Response

Return all Coin< coin_type > objects owned by an address.

Parameters

Result

CoinPage <

```
Page_for_Coin_and_String
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets all SUI coins owned by the address provided. Return a paginated list of limit results per page. Similar to suix_getAllCoins , but provides a way to filter by coin type.

Request

Response

Return total supply for a coin

Parameters

Result

Supply <

```
Supply
```

```
>
```

Example

Gets total supply for the type of coin provided.

Request

Response

Return the dynamic field object information for a specified object

Parameters

Result

SuiObjectResponse <

```
SuiObjectResponse
```

>

Example

Gets the information for the dynamic field the request provides.

Request

Response

Return the list of dynamic field objects owned by an object.

Parameters

Result

DynamicFieldPage <

```
Page_for_DynamicFieldInfo_and_ObjectID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets dynamic fields for the object the request provides in a paginated list of limit dynamic field results per page. The default limit is 50.

Request

Response

Return the list of objects owned by an address. Note that if the address owns more than QUERY_MAX_RESULT_LIMIT objects, the pagination is not accurate, because previous page may have been updated when the next page is fetched. Please use suix_queryObjects if this is a concern.

Parameters

Result

ObjectsPage <

```
Page_for_SuiObjectResponse_and_ObjectID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns all the objects the address provided in the request owns and that match the filter. By default, only the digest value is returned, but the request returns additional information by setting the relevant keys to true. A cursor value is also provided, so the list of results begin after that value.

Request

Response

Return list of events for a specified query criteria.

Parameters

Result

EventPage <

```
Page_for_Event_and_EventID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the events for a specified query criteria.

Request

Response

Return list of transactions for a specified query criteria.

Parameters

Result

TransactionBlocksPage <

```
Page_for_TransactionBlockResponse_and_TransactionDigest
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the transaction digest for specified query criteria.

Request

Response

Return the resolved address given resolver and name

Parameters

Result

SuiAddress <

```
SuiAddress
```

>

Example

Returns the resolved address for the name the request provides.

Request

Response

Return the resolved names given address, if multiple names are resolved, the first one is the primary name.

Parameters

Result

Page <

```
Page_for_String_and_ObjectID
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the SuiNS name for the address the request provides. Currently, the API returns only the first name in cases where there are multiple. Future support will use the cursor ID and limit values in the request to control pagination of the response for addresses with multiple names.

Request

Response

Subscribe to a stream of Sui event

Parameters

Result

SuiEvent <

```
Event
```

```
>
```

Subscribe to a stream of Sui transaction effects

Parameters

Result

SuiTransactionBlockEffects <

```
TransactionBlockEffects
```

```
>
```

Return the committee information for the asked epoch .

Parameters

Result

SuiCommittee <

```
CommitteeInfo
```

>

RPC representation of the [Committee] type.

Example

Gets committee information for epoch 5000.

Request

Response

Return the latest SUI system state object on-chain.

Parameters

None

Result

SuiSystemStateSummary <

```
SuiSystemStateSummary
```

>

This is the JSON-RPC type for the SUI system state object. It flattens all fields to make them top-level fields such that it as minimum dependencies to the internal data structures of the SUI system state type.

Example

Gets objects owned by the address in the request.

Request

Response

Return the reference gas price for the network

Parameters

None

Result

BigInt <

```
BigInt_for_uint64
```

>

Example

Gets reference gas price information for the network.

Request

Response

Return all [DelegatedStake].

Parameters

Result

Vec <[DelegatedStake

```
]>
```

Example

Returns the staking information for the address the request provides.

Request

Response

Return one or more [DelegatedStake]. If a Stake was withdrawn its status will be Unstaked.

Parameters

Result

Vec <[DelegatedStake

```
]>
```

Example

Returns the staking information for the address the request provides.

Request

Response

Return the validator APY

Parameters

None

Result

ValidatorApys <

```
ValidatorApys

>
```

Example

Gets the APY for all validators.

Request

Response

Return the argument types of a Move function, based on normalized Type.

Parameters

Result

Vec <[MoveFunctionArgType

```
]>
```

Example

Returns the argument types for the package and function the request provides.

Request

Response

Return a structured representation of Move function

Parameters

Result

SuiMoveNormalizedFunction <

```
SuiMoveNormalizedFunction
```

```
>
```

Example

Returns the structured representation of the function the request provides.

Request

Response

Return a structured representation of Move module

Parameters

Result

SuiMoveNormalizedModule <

```
SuiMoveNormalizedModule
```

```
>
```

Example

Gets a structured representation of the Move module for the package in the request.

Request

Response

Return structured representations of all modules in the given package

Parameters

Result

BTreeMap <

```
SuiMoveNormalizedModule
```

```
>
```

Example

Gets structured representations of all the modules for the package in the request.

Request

Response

Return a structured representation of Move struct

Parameters

Result

SuiMoveNormalizedStruct <

```
SuiMoveNormalizedStruct
```

```
>
```

Example

Gets a structured representation of the struct in the request.

Request

Response

Return the first four bytes of the chain's genesis checkpoint digest.

Parameters

None

Result

String <

```
string
```

```
>
```

Example

Gets the identifier for the chain receiving the POST.

Request

Response

Return a checkpoint

Parameters

Result

Checkpoint <

```
Checkpoint
```

```
>
```

Example

Gets checkpoint information for the checkpoint ID in the request.

Request

Response

Return paginated list of checkpoints

Parameters

Result

CheckpointPage <

```
Page_for_Checkpoint_and_BigInt_for_uint64
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets a paginated list in descending order of all checkpoints starting at the provided cursor. Each page of results has a maximum number of checkpoints set by the provided limit.

Request

Response

Return transaction events.

Parameters

Result

Vec <[Event

```
]>
```

Example

Returns the events the transaction in the request emits.

Request

Response

Return the sequence number of the latest checkpoint that has been executed

Parameters

None

Result

BigInt <

```
BigInt_for_uint64
```

```
>
```

Example

Gets the sequence number for the latest checkpoint.

Request

Response

Return the object information for a specified object

Parameters

Result

SuiObjectResponse <

```
SuiObjectResponse
```

```
>
```

Example

Gets Object data for the ID in the request.

Request

Response

Return the protocol config table for the given version number. If the version number is not specified, If none is specified, the node uses the version of the latest epoch it has processed.

Parameters

Result

ProtocolConfigResponse <

```
ProtocolConfig

>
```

Example

Returns the protocol config for the given protocol version. If none is specified, the node uses the version of the latest epoch it has processed

Request

Response

Return the total number of transaction blocks known to the server.

Parameters

None

Result

BigInt <

```
BigInt_for_uint64

>
```

Example

Gets total number of transactions on the network.

Request

Response

Return the transaction response object.

Parameters

Result

SuiTransactionBlockResponse <

```
TransactionBlockResponse

>
```

Example

Returns the transaction response object for specified transaction digest.

Request

Response

Return the object data for a list of objects

Parameters

Result

Vec <[SuiObjectResponse

```
]>
```

Example

Gets objects by IDs.

Request

Response

Returns an ordered list of transaction responses The method will throw an error if the input contains any duplicate or the input size exceeds QUERY_MAX_RESULT_LIMIT

Parameters

Result

Vec <[TransactionBlockResponse

```
]>
```

Example

Returns the transaction data for specified digest.

Request

Response

Note there is no software-level guarantee/SLA that objects with past versions can be retrieved by this API, even if the object and version exists/existed. The result may vary across nodes depending on their pruning policies. Return the object information for a specified version

Parameters

Result

SuiPastObjectResponse <

```
ObjectRead
```

```
>
```

Example

Gets Past Object data.

Request

Response

Note there is no software-level guarantee/SLA that objects with past versions can be retrieved by this API, even if the object and version exists/existed. The result may vary across nodes depending on their pruning policies. Return the object information for a specified version

Parameters

Result

Vec <[ObjectRead

```
]>
```

Example

Gets Past Object data for a vector of objects.

Request

Response

Verify a zklogin signature for the given bytes, intent scope and author.

Parameters

Result

ZkLoginVerifyResult <

```
ZkLoginVerifyResult

>
```

Create an unsigned batched transaction.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes

>
```

Create an unsigned transaction to merge multiple coins into one coin.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes

>
```

Create an unsigned transaction to execute a Move call on the network, by calling the specified function in the module of a given package.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Send Coin<T> to a list of addresses, where T can be any coin type, following a list of amounts, The object specified in the gas field will be used to pay the gas fee for the transaction. The gas object can not appear in input_coins . If the gas object is not specified, the RPC server will auto-select one.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Send all SUI coins to one recipient. This is for SUI coin only and does not require a separate gas coin object. Specifically, what pay_all_sui does are: 1. accumulate all SUI from input coins and deposit all SUI to the first input coin 2. transfer the updated first coin to the recipient and also use this first coin as gas coin object. 3. the balance of the first input coin after tx is sum(input_coins) - actual_gas_cost. 4. all other input coins other than the first are deleted.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Send SUI coins to a list of addresses, following a list of amounts. This is for SUI coin only and does not require a separate gas coin object. Specifically, what pay_sui does are: 1. debit each input_coin to create new coin following the order of amounts and assign it to the corresponding recipient. 2. accumulate all residual SUI from input coins left and deposit all SUI to the first input coin, then use the first input coin as the gas coin object. 3. the balance of the first input coin after tx is sum(input_coins) - sum(amounts) - actual_gas_cost 4. all other input coints other than the first one are deleted.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to publish a Move package.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Add stake to a validator's staking pool using multiple coins and amount.

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Withdraw stake from a validator's staking pool.

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Create an unsigned transaction to split a coin object into multiple coins.

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Create an unsigned transaction to split a coin object into multiple equal-size coins.

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Create an unsigned transaction to transfer an object from one address to another. The object's type must allow public transfers

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Create an unsigned transaction to send SUI coin object to a Sui address. The SUI object is also used as the gas object.

Parameters

Result

TransactionBlockBytes <

`TransactionBlockBytes`

>

Runs the transaction in dev-inspect mode. Which allows for nearly any transaction (or Move call) with any arguments. Detailed results are provided, including both the transaction effects and any return values.

Parameters

Result

DevInspectResults <

```
DevInspectResults
```

>

The response from processing a dev inspect transaction

Example

Runs the transaction in dev-inspect mode. Which allows for nearly any transaction (or Move call) with any arguments. Detailed results are provided, including both the transaction effects and any return values.

Request

Response

Return transaction execution effects including the gas cost summary, while the effects are not committed to the chain.

Parameters

Result

DryRunTransactionBlockResponse <

```
DryRunTransactionBlockResponse
```

>

Example

Dry runs a transaction block to get back estimated gas fees and other potential effects.

Request

Response

Execute the transaction and wait for results if desired. Request types: 1. WaitForEffectsCert: waits for TransactionEffectsCert and then return to client. This mode is a proxy for transaction finality. 2. WaitForLocalExecution: waits for TransactionEffectsCert and make sure the node executed the transaction locally before returning the client. The local execution makes sure this node is aware of this transaction when client fires subsequent queries. However if the node fails to execute the transaction locally in a timely manner, a bool type in the response is set to false to indicated the case. request_type is default to be WaitForEffectsCert unless options.show_events or options.show_effects is true

Parameters

Result

SuiTransactionBlockResponse <

```
TransactionBlockResponse
```

>

Example

Executes a transaction with serialized signatures.

Request

Response

One of

Object

The contained SuiAddress exclusively has all permissions: read, write, delete, transfer

Defines the compressed version of the public key that we pass around in Sui

Base64

Object

Object

Owner

String

Base64 encoding

String

String

String

String

String

A struct that stores a Bn254 Fq field element as 32 bytes.

String

A struct that stores a Bn254 Fr field element as 32 bytes.

String

Object

CheckpointDigest

BigInt_for_uint64

GasCostSummary

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

Base64

One of

Object

Representation of a Checkpoint's digest

Digest

Any of

BigInt_for_uint64

CheckpointDigest

A claim consists of value and index_mod_4.

O bject

O bject

RPC representation of the [Committee] type.

O bject

Unlike [enum Signature], [enum CompressedSignature] does not contain public key.

One of

Object

Object

Object

Object

Object

Uses an enum to allow for future expansion of the ConsensusDeterminedVersionAssignments.

One of

Object

Object

One of

Object

Object

O bject

ObjectID

SuiAddress

Additional rguments supplied to dev inspect beyond what is allowed in today's API.

O bject

The response from processing a dev inspect transaction

O bject

TransactionBlockEffects

A representation of a 32 byte digest

Base58

O bject

O bject

O bject

One of

Object

Object

O bject

S tring enum ["DynamicField" | "DynamicObject"]

The Sha256 digest of an EllipticCurveMultisetHash committing to the live object set.

O bject

O bject

ProtocolVersion

O bject

EventID

ObjectID

SuiAddress

One of

Object

Object

One of

Object

Return all events.

Object

Return events that match any of the given filters. Only supported on event subscriptions.

Object

Query by sender address.

Object

Return events emitted by the given transaction.

Object

Return events emitted in a specified Move module. If the event is defined in Module A but emitted in a tx with Module B, query MoveModule by module B returns the event. Query MoveEventModule by module A returns the event too.

ObjectID

Object

Return events with the given Move event struct name (struct tag). For example, if the event is defined in 0xabcd::MyModule , and named Foo , then the struct tag is 0xabcd::MyModule::Foo .

Object

Return events with the given Move module name where the event struct is defined. If the event is defined in Module A but emitted in a tx with Module B, query MoveEventModule by module A returns the event. Query MoveModule by module B returns the event too.

ObjectID

Object

Return events emitted in [start_time, end_time] interval

BigInt_for_uint64

BigInt_for_uint64

Unique ID of a Sui Event, the ID is a combination of transaction digest and event seq number.

O bject

S tring enum ["WaitForEffectsCert" | "WaitForLocalExecution"]

One of

Object

Object

Summary of the charges in a transaction. Storage is charged independently of computation. There are 3 parts to the storage charges: storage_cost : it is the charge of storage at the time the transaction is executed. The cost of storage is the number of bytes of the objects being mutated multiplied by a variable storage cost per byte storage_rebate : this is the amount a user gets back when manipulating an object. The storage_rebate is the storage_cost for an object minus fees. non_refundable_storage_fee : not all the value of the object storage cost is given back to user and there is a small fraction that is kept by the system. This value tracks that charge.

When looking at a gas cost summary the amount charged to the user is computation_cost + storage_cost - storage_rebate and that is the amount that is deducted from the gas coins. non_refundable_storage_fee is collected from the objects being mutated/deleted and it is tracked by the system in storage funds.

Objects deleted, including the older versions of objects mutated, have the storage field on the objects added up to a pool of "potential rebate". This rebate then is reduced by the "nonrefundable rate" such that: potential_rebate(storage cost of deleted/mutated objects) = storage_rebate + non_refundable_storage_fee

O bject

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

O bject

Due to the incompatibility of [enum Signature] (which dispatches a trait that assumes signature and pubkey bytes for verification), here we add a wrapper enum where member can just implement a lightweight [trait AuthenticatorTrait]. This way MultiSig (and future Authenticators) can implement its own verify .

One of

Object

Object

Object

Object

Object

O bject

ObjectID

SequenceNumber

Hex string encoding.

S tring

One of

Object

Object

Object

O bject

One of

String enum: ["Pure"]

Object

Any of

[MoveValue]

Object

Object

Any of

I nteger < uint32 > Minimum: 0

Boolean

SuiAddress

[MoveValue]

String

Object

MoveStruct

MoveVariant

O bject

The struct that contains signatures and public keys necessary for authenticating a MultiSig.

O bject

MultiSigPublicKey

Deprecated, use [struct MultiSig] instead. The struct that contains signatures and public keys necessary for authenticating a MultiSigLegacy.

O bject

Base64

MultiSigPublicKeyLegacy

The struct that contains the public key used for authenticating a MultiSig.

O bject

Deprecated, use [struct MultiSigPublicKey] instead. The struct that contains the public key used for authenticating a MultiSig.

O bject

ObjectChange are derived from the object mutations in the TransactionEffect to provide richer object information.

One of

Object

Module published

Object

Transfer objects to new address / wrap in another object

Object

Object mutated.

Object

Delete object

Object

Wrapped object

Object

New object creation

O bject

ObjectDigest

SequenceNumber

O bject

One of

Object

The object exists and is found with this version

Object

The object does not exist

Object

The object is found to be deleted with this version

Object

The object exists but not found with this version

Object

The asked object version is higher than the latest

O bject

ObjectDigest

ObjectID

SequenceNumber

One of

Object

Object

Object

ObjectDigest

SequenceNumber2

Object

Object

O bject

S tring enum ["ByImmutableReference" | "ByMutableReference" | "ByValue"]

O bject

One of

Object

Object is exclusively owned by a single address, and is mutable.

Object

Object is exclusively owned by a single object, and is mutable. The object ID is converted to SuiAddress as SuiAddress is universal.

Object

Object is shared, can be used by any address, and is mutable.

SequenceNumber2

String enum: ["Immutable"]

Object is immutable, and hence ownership doesn't matter.

Object

Object is sequenced via consensus. Ownership is managed by the configured authenticator.

Note: wondering what happened to V1 ? Shared above was the V1 of consensus objects.

Authenticator

SequenceNumber2

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

An passkey authenticator with parsed fields. See field defition below. Can be initialized from [struct RawPasskeyAuthenticator].

O bject

O bject

One of

Object

Object

Object

Object

Object

One of

Object

Object

Object

Object

Object

One of

Object

Object

One of

Object

Object

I nteger

One of

Object

Object

Object

O bject

ObjectID

One of

Object

Object

Object

O bject

An argument to a transaction in a programmable transaction block

One of

String enum: ["GasCoin"]

The gas coin. The gas coin can only be used by-ref, except for with TransferObjects , which can use it by-value.

Object

One of the input objects or primitive values (from ProgrammableTransactionBlock inputs)

Object

The result of another transaction (from ProgrammableTransactionBlock transactions)

Object

Like a Result but it accesses a nested result. Currently, the only usage of this is to access a value from a Move call with multiple return values.

O bject

One of

Object

Object

O bject

O bject

One of

String enum: ["AuthenticatorStateCreate" | "RandomnessStateCreate" | "CoinDenyListStateCreate" | "StoreExecutionTimeObservations"]

Object

Object

Object

Object

O bject

O bject

O bject

S tring enum ["Copy" | "Drop" | "Store" | "Key"]

O bject

O bject

O bject

O bject

O bject

O bject

O bject

One of

String enum: ["Bool" | "U8" | "U16" | "U32" | "U64" | "U128" | "U256" | "Address" | "Signer"]

Object

Object

Object

Object

Object

O bject

S tring enum ["Private" | "Public" | "Friend"]

One of

Object

Object

Object

Object

Query by type a specified Package.

Object

Query by type a specified Move module.

ObjectID

Object

Query by type

Object

Object

Object

Object

Object

O bject

The transaction for calling a Move function, either an entry function or a public function (which cannot return references).

O bject

ObjectID

This is the JSON-RPC type for the SUI system state object. It flattens all fields to make them top-level fields such that it as minimum dependencies to the internal data structures of the SUI system state type.

O bject

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

A single transaction in a programmable transaction block.

One of

Object

A call to either an entry or a public Move function

Object

(Vec, address) It sends n-objects to the specified address. These objects must have store (public transfer) and either the previous owner must be an address or the object must be newly created.

Object

(&mut Coin, Vec) -> Vec> It splits off some amounts into a new coins with those amounts

Object

(&mut Coin, Vec>) It merges n-coins into the first coin

Object

Publishes a Move package. It takes the package bytes and a list of the package's transitive dependencies to link against on-chain.

Object

Upgrades a Move package

Object

forall T: Vec -> vector Given n-values of the same type, it constructs a vector. For non objects or an empty vector, the type tag must be specified.

One of

String enum: ["Commit"]

Regular Sui Transactions that are committed on chain

String enum: ["DevInspect"]

Simulated transaction that allows calling any Move function with arbitrary values.

This is the JSON-RPC type for the SUI validator. It flattens all inner structures to top-level fields so that they are decoupled from the internal definitions.

O bject

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

O bject

O bject

O bject

Base64

One of

Object

One of

Object

The response from processing a transaction or a certified transaction

BigInt_for_uint64

OwnedObjectRef

ExecutionStatus

TransactionDigest

O bject

One of

Object

A system transaction that will update epoch information on-chain.

Object

A system transaction used for initializing the initial state of the chain.

Object

A system transaction marking the start of a series of transactions scheduled as part of a checkpoint

Object

A series of transactions where the results of one transaction can be used in future transactions

Object

A transaction which updates global authenticator state

Object

A transaction which updates global randomness state

Object

The transaction which occurs only at the end of the epoch

Object

Object

Object

O bject

Base64

O bject

O bject

A transaction will have a (unique) digest.

Digest

One of

Object

CURRENTLY NOT SUPPORTED. Query by checkpoint.

Object

Query by move function.

Object

Query by input object.

Object

Query by changed object, including created, mutated and unwrapped objects.

Object

Query for transactions that touch this object.

Object

Query by sender address.

Object

Query by recipient address.

Object

Query by sender and recipient address.

Object

CURRENTLY NOT SUPPORTED. Query txs that have a given address as sender or recipient.

Object

Query by transaction kind

Object

Query transactions of any given kind in the input.

O bject

Identifies a struct and the module it was defined in

O bject

S tring

Upgraded package info for the linkage table

O bject

ObjectID

SequenceNumber2

O bject

O bject

An zk login authenticator with all the necessary fields.

O bject

All inputs required for the zk login proof verification and other public inputs.

O bject

S tring enum ["TransactionData" | "PersonalMessage"]

The struct for zk login proof.

O bject

A wrapper struct to retrofit in [enum PublicKey] for zkLogin. Useful to construct [struct MultiSigPublicKey].

Base64

O bject

Coin Query API

Return the total coin balance for all coin type, owned by the address owner.

Parameters

Result

Vec <[Balance

] >

Example

Gets all balances for the address in the request.

Request

Response

Return all Coin objects owned by an address.

Parameters

Result

CoinPage <

```
Page_for_Coin_and_String
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets all coins for the address in the request body. Begin listing the coins that are after the provided cursor value and return only the limit amount of results per page.

Request

Response

Return the total coin balance for one coin type, owned by the address owner.

Parameters

Result

Balance <

```
Balance
```

>

Example

Gets the balance of the specified type of coin for the address in the request.

Request

Response

Return metadata (e.g., symbol, decimals) for a coin. Note that if the coin's metadata was wrapped in the transaction that published its marker type, or the latest version of the metadata object is wrapped or deleted, it will not be found.

Parameters

Result

SuiCoinMetadata <

```
SuiCoinMetadata
```

>

Example

Gets the metadata for the coin type in the request.

Request

Response

Return all Coin< coin_type > objects owned by an address.

Parameters

Result

CoinPage <

```
Page_for_Coin_and_String
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets all SUI coins owned by the address provided. Return a paginated list of limit results per page. Similar to suix_getAllCoins , but provides a way to filter by coin type.

Request

Response

Return total supply for a coin

Parameters

Result

Supply <

```
Supply
```

```
>
```

Example

Gets total supply for the type of coin provided.

Request

Response

```
bash ]
```

Extended API

Return the dynamic field object information for a specified object

Parameters

Result

SuiObjectResponse <

```
SuiObjectResponse
```

```
>
```

Example

Gets the information for the dynamic field the request provides.

Request

Response

Return the list of dynamic field objects owned by an object.

Parameters

Result

DynamicFieldPage <

```
Page_for_DynamicFieldInfo_and_ObjectID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets dynamic fields for the object the request provides in a paginated list of limit dynamic field results per page. The default limit is 50.

Request

Response

Return the list of objects owned by an address. Note that if the address owns more than QUERY_MAX_RESULT_LIMIT objects, the pagination is not accurate, because previous page may have been updated when the next page is fetched. Please use suix_queryObjects if this is a concern.

Parameters

Result

ObjectsPage <

```
Page_for_SuiObjectResponse_and_ObjectID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns all the objects the address provided in the request owns and that match the filter. By default, only the digest value is returned, but the request returns additional information by setting the relevant keys to true. A cursor value is also provided, so the list of results begin after that value.

Request

Response

Return list of events for a specified query criteria.

Parameters

Result

EventPage <

```
Page_for_Event_and_EventID
```

>

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the events for a specified query criteria.

Request

Response

Return list of transactions for a specified query criteria.

Parameters

Result

TransactionBlocksPage <

```
Page_for_TransactionBlockResponse_and_TransactionDigest
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the transaction digest for specified query criteria.

Request

Response

Return the resolved address given resolver and name

Parameters

Result

SuiAddress <

```
SuiAddress
```

```
>
```

Example

Returns the resolved address for the name the request provides.

Request

Response

Return the resolved names given address, if multiple names are resolved, the first one is the primary name.

Parameters

Result

Page <

```
Page_for_String_and_ObjectID
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Returns the SuiNS name for the address the request provides. Currently, the API returns only the first name in cases where there are multiple. Future support will use the cursor ID and limit values in the request to control pagination of the response for addresses with multiple names.

Request

Response

Subscribe to a stream of Sui event

Parameters

Result

SuiEvent <

```
Event
```

```
>
```

Subscribe to a stream of Sui transaction effects

Parameters

Result

SuiTransactionBlockEffects <

```
TransactionBlockEffects
```

```
>
```

```
bash {
```

Governance Read API

Return the committee information for the asked epoch .

Parameters

Result

SuiCommittee <

```
CommitteeInfo
```

```
>
```

RPC representation of the [Committee] type.

Example

Gets committee information for epoch 5000.

Request

Response

Return the latest SUI system state object on-chain.

Parameters

None

Result

SuiSystemStateSummary <

```
SuiSystemStateSummary

>
```

This is the JSON-RPC type for the SUI system state object. It flattens all fields to make them top-level fields such that it as minimum dependencies to the internal data structures of the SUI system state type.

Example

Gets objects owned by the address in the request.

Request

Response

Return the reference gas price for the network

Parameters

None

Result

BigInt <

```
BigInt_for_uint64

>
```

Example

Gets reference gas price information for the network.

Request

Response

Return all [DelegatedStake].

Parameters

Result

Vec <[DelegatedStake

```
]>
```

Example

Returns the staking information for the address the request provides.

Request

Response

Return one or more [DelegatedStake]. If a Stake was withdrawn its status will be Unstaked.

Parameters

Result

Vec <[DelegatedStake

```
]>
```

Example

Returns the staking information for the address the request provides.

Request

Response

Return the validator APY

Parameters

None

Result

ValidatorApys <

```
ValidatorApys
```

```
>
```

Example

Gets the APY for all validators.

Request

Response

```
bash ]
```

Move Utils

Return the argument types of a Move function, based on normalized Type.

Parameters

Result

Vec <[MoveFunctionArgType

```
]>
```

Example

Returns the argument types for the package and function the request provides.

Request

Response

Return a structured representation of Move function

Parameters

Result

SuiMoveNormalizedFunction <

```
SuiMoveNormalizedFunction
```

```
>
```

Example

Returns the structured representation of the function the request provides.

Request

Response

Return a structured representation of Move module

Parameters

Result

SuiMoveNormalizedModule <

```
        SuiMoveNormalizedModule

        >
```

Example

Gets a structured representation of the Move module for the package in the request.

Request

Response

Return structured representations of all modules in the given package

Parameters

Result

BTreeMap <

```
        SuiMoveNormalizedModule


        >
```

Example

Gets structured representations of all the modules for the package in the request.

Request

Response

Return a structured representation of Move struct

Parameters

Result

SuiMoveNormalizedStruct <

```
        SuiMoveNormalizedStruct

        >
```

Example

Gets a structured representation of the struct in the request.

Request

Response

```
bash ]
```

Read API

Return the first four bytes of the chain's genesis checkpoint digest.

Parameters

None

Result

String <

```
string
```

```
>
```

Example

Gets the identifier for the chain receiving the POST.

Request

Response

Return a checkpoint

Parameters

Result

Checkpoint <

```
Checkpoint
```

```
>
```

Example

Gets checkpoint information for the checkpoint ID in the request.

Request

Response

Return paginated list of checkpoints

Parameters

Result

CheckpointPage <

```
Page_for_Checkpoint_and_BigInt_for_uint64
```

```
>
```

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Example

Gets a paginated list in descending order of all checkpoints starting at the provided cursor. Each page of results has a maximum number of checkpoints set by the provided limit.

Request

Response

Return transaction events.

Parameters

Result

Vec <[Event

```
]>
```

Example

Returns the events the transaction in the request emits.

Request

Response

Return the sequence number of the latest checkpoint that has been executed

Parameters

None

Result

BigInt <

```
BigInt_for_uint64

>
```

Example

Gets the sequence number for the latest checkpoint.

Request

Response

Return the object information for a specified object

Parameters

Result

SuiObjectResponse <

```
SuiObjectResponse

>
```

Example

Gets Object data for the ID in the request.

Request

Response

Return the protocol config table for the given version number. If the version number is not specified, If none is specified, the node uses the version of the latest epoch it has processed.

Parameters

Result

ProtocolConfigResponse <

```
ProtocolConfig
```

```
>
```

Example

Returns the protocol config for the given protocol version. If none is specified, the node uses the version of the latest epoch it has processed

Request

Response

Return the total number of transaction blocks known to the server.

Parameters

None

Result

BigInt <

```
BigInt_for_uint64
```

```
>
```

Example

Gets total number of transactions on the network.

Request

Response

Return the transaction response object.

Parameters

Result

SuiTransactionBlockResponse <

```
TransactionBlockResponse
```

```
>
```

Example

Returns the transaction response object for specified transaction digest.

Request

Response

Return the object data for a list of objects

Parameters

Result

Vec <[SuiObjectResponse

```
]>
```

Example

Gets objects by IDs.

Request

Response

Returns an ordered list of transaction responses The method will throw an error if the input contains any duplicate or the input size exceeds QUERY_MAX_RESULT_LIMIT

Parameters

Result

Vec <[TransactionBlockResponse

```
]>
```

Example

Returns the transaction data for specified digest.

Request

Response

Note there is no software-level guarantee/SLA that objects with past versions can be retrieved by this API, even if the object and version exists/existed. The result may vary across nodes depending on their pruning policies. Return the object information for a specified version

Parameters

Result

SuiPastObjectResponse <

```
ObjectRead


>
```

Example

Gets Past Object data.

Request

Response

Note there is no software-level guarantee/SLA that objects with past versions can be retrieved by this API, even if the object and version exists/existed. The result may vary across nodes depending on their pruning policies. Return the object information for a specified version

Parameters

Result

Vec <[ObjectRead

```
]>
```

Example

Gets Past Object data for a vector of objects.

Request

Response

Verify a zklogin signature for the given bytes, intent scope and author.

Parameters

Result

ZkLoginVerifyResult <

```
        ZkLoginVerifyResult


        >
```

```
bash }
```

Transaction Builder API

Create an unsigned batched transaction.

Parameters

Result

TransactionBlockBytes <

```
        TransactionBlockBytes


        >
```

Create an unsigned transaction to merge multiple coins into one coin.

Parameters

Result

TransactionBlockBytes <

```
        TransactionBlockBytes


        >
```

Create an unsigned transaction to execute a Move call on the network, by calling the specified function in the module of a given package.

Parameters

Result

TransactionBlockBytes <

```
        TransactionBlockBytes


        >
```

Send Coin<T> to a list of addresses, where T can be any coin type, following a list of amounts, The object specified in the gas field will be used to pay the gas fee for the transaction. The gas object can not appear in input_coins . If the gas object is not specified, the RPC server will auto-select one.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Send all SUI coins to one recipient. This is for SUI coin only and does not require a separate gas coin object. Specifically, what pay_all_sui does are: 1. accumulate all SUI from input coins and deposit all SUI to the first input coin 2. transfer the updated first coin to the recipient and also use this first coin as gas coin object. 3. the balance of the first input coin after tx is sum(input_coins) - actual_gas_cost. 4. all other input coins other than the first are deleted.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Send SUI coins to a list of addresses, following a list of amounts. This is for SUI coin only and does not require a separate gas coin object. Specifically, what pay_sui does are: 1. debit each input_coin to create new coin following the order of amounts and assign it to the corresponding recipient. 2. accumulate all residual SUI from input coins left and deposit all SUI to the first input coin, then use the first input coin as the gas coin object. 3. the balance of the first input coin after tx is sum(input_coins) - sum(amounts) - actual_gas_cost 4. all other input coints other than the first one are deleted.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to publish a Move package.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Add stake to a validator's staking pool using multiple coins and amount.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Withdraw stake from a validator's staking pool.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to split a coin object into multiple coins.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to split a coin object into multiple equal-size coins.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to transfer an object from one address to another. The object's type must allow public transfers

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Create an unsigned transaction to send SUI coin object to a Sui address. The SUI object is also used as the gas object.

Parameters

Result

TransactionBlockBytes <

```
TransactionBlockBytes
```

>

Write API

Runs the transaction in dev-inspect mode. Which allows for nearly any transaction (or Move call) with any arguments. Detailed results are provided, including both the transaction effects and any return values.

Parameters

Result

DevInspectResults <

```
        DevInspectResults


        >
```

The response from processing a dev inspect transaction

Example

Runs the transaction in dev-inspect mode. Which allows for nearly any transaction (or Move call) with any arguments. Detailed results are provided, including both the transaction effects and any return values.

Request

Response

Return transaction execution effects including the gas cost summary, while the effects are not committed to the chain.

Parameters

Result

DryRunTransactionBlockResponse <

```
        DryRunTransactionBlockResponse


        >
```

Example

Dry runs a transaction block to get back estimated gas fees and other potential effects.

Request

Response

Execute the transaction and wait for results if desired. Request types: 1. WaitForEffectsCert: waits for TransactionEffectsCert and then return to client. This mode is a proxy for transaction finality. 2. WaitForLocalExecution: waits for TransactionEffectsCert and make sure the node executed the transaction locally before returning the client. The local execution makes sure this node is aware of this transaction when client fires subsequent queries. However if the node fails to execute the transaction locally in a timely manner, a bool type in the response is set to false to indicated the case. request_type is default to be WaitForEffectsCert unless options.show_events or options.show_effects is true

Parameters

Result

SuiTransactionBlockResponse <

```
        TransactionBlockResponse


        >
```

Example

Executes a transaction with serialized signatures.

Request

Response

```
bash ]
```

AdditionalConsensusStateDigest

```
bash "AdditionalConsensusStateDigest": { "$ref": "#/components/schemas/Digest" }
```

Authenticator

One of

Object

The contained SuiAddress exclusively has all permissions: read, write, delete, transfer

```
bash "Authenticator": { "oneOf": [ { "description": "The contained SuiAddress exclusively has all
permissions: read, write, delete, transfer", "type": "object", "required": [ "SingleOwner" ],
"properties": { "SingleOwner": { "$ref": "#/components/schemas/SuiAddress" } },
"additionalProperties": false } ] }
```

AuthorityPublicKeyBytes

Defines the compressed version of the public key that we pass around in Sui

Base64

```
bash "AuthorityPublicKeyBytes": { "description": "Defines the compressed version of the public key
that we pass around in Sui", "allOf": [ { "$ref": "#/components/schemas/Base64" } ] }
```

Balance

Object

```
bash "Balance": { "type": "object", "required": [ "coinObjectCount", "coinType", "lockedBalance",
"totalBalance" ], "properties": { "coinObjectCount": { "type": "integer", "format": "uint",
"minimum": 0 }, "coinType": { "type": "string" }, "lockedBalance": { "type": "object",
"additionalProperties": { "$ref": "#/components/schemas/BigInt_for_uint128" } }, "totalBalance": {
"$ref": "#/components/schemas/BigInt_for_uint128" } } }
```

BalanceChange

Object

Owner

```
bash "BalanceChange": { "type": "object", "required": [ "amount", "coinType", "owner" ],
"properties": { "amount": { "description": "The amount indicate the balance value changes, negative
amount means spending coin value and positive means receiving coin value.", "type": "string" },
"coinType": { "type": "string" }, "owner": { "description": "Owner of the balance change", "allOf":
[ { "$ref": "#/components/schemas/Owner" } ] } } }
```

Base58

String

```
bash "Base58": { "type": "string" }
```

Base64

Base64 encoding

String

```
bash "Base64": { "description": "Base64 encoding", "type": "string" }
```

BigInt_for_uint128

String

```
bash "BigInt_for_uint128": { "type": "string" }
```

BigInt_for_uint16

String

```
bash "BigInt_for_uint16": { "type": "string" }
```

BigInt_for_uint32

String

```
bash "BigInt_for_uint32": { "type": "string" }
```

BigInt_for_uint64

String

```
bash "BigInt_for_uint64": { "type": "string" }
```

Bn254FqElement

A struct that stores a Bn254 Fq field element as 32 bytes.

String

```
bash "Bn254FqElement": { "description": "A struct that stores a Bn254 Fq field element as 32
bytes.", "type": "string" }
```

Bn254FrElement

A struct that stores a Bn254 Fr field element as 32 bytes.

String

```
bash "Bn254FrElement": { "description": "A struct that stores a Bn254 Fr field element as 32
bytes.", "type": "string" }
```

Checkpoint

Object

CheckpointDigest

BigInt_for_uint64

GasCostSummary

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

Base64

```
bash "Checkpoint": { "type": "object", "required": [ "checkpointCommitments", "digest", "epoch",
"epochRollingGasCostSummary", "networkTotalTransactions", "sequenceNumber", "timestampMs",
"transactions", "validatorSignature" ], "properties": { "checkpointCommitments": { "description":
"Commitments to checkpoint state", "type": "array", "items": { "$ref":
"#/components/schemas/CheckpointCommitment" } }, "digest": { "description": "Checkpoint digest",
"allOf": [ { "$ref": "#/components/schemas/CheckpointDigest" } ] }, "endOfEpochData": {
"description": "Present only on the final checkpoint of the epoch.", "anyOf": [ { "$ref":
"#/components/schemas/EndOfEpochData" }, { "type": "null" } ] }, "epoch": { "description":
"Checkpoint's epoch ID", "allOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" } ] },
"epochRollingGasCostSummary": { "description": "The running total gas costs of all transactions
included in the current epoch so far until this checkpoint.", "allOf": [ { "$ref":
"#/components/schemas/GasCostSummary" } ] }, "networkTotalTransactions": { "description": "Total
number of transactions committed since genesis, including those in this checkpoint.", "allOf": [ {
"$ref": "#/components/schemas/BigInt_for_uint64" } ] }, "previousDigest": { "description": "Digest
of the previous checkpoint", "anyOf": [ { "$ref": "#/components/schemas/CheckpointDigest" }, {
"type": "null" } ] }, "sequenceNumber": { "description": "Checkpoint sequence number", "allOf": [ {
"$ref": "#/components/schemas/BigInt_for_uint64" } ] }, "timestampMs": { "description": "Timestamp
of the checkpoint - number of milliseconds from the Unix epoch Checkpoint timestamps are monotonic,
but not strongly monotonic - subsequent checkpoints can have same timestamp if they originate from
the same underlining consensus commit", "allOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" } ] }, "transactions": { "description": "Transaction
digests", "type": "array", "items": { "$ref": "#/components/schemas/TransactionDigest" } },
"validatorSignature": { "description": "Validator Signature", "allOf": [ { "$ref":
"#/components/schemas/Base64" } ] } } } }
```

CheckpointCommitment

One of

Object

```
bash "CheckpointCommitment": { "oneOf": [ { "type": "object", "required": [
"ECMHLiveObjectSetDigest" ], "properties": { "ECMHLiveObjectSetDigest": { "$ref":
"#/components/schemas/ECMHLiveObjectSetDigest" } }, "additionalProperties": false } ] }
```

CheckpointDigest

Representation of a Checkpoint's digest

Digest

```
bash "CheckpointDigest": { "description": "Representation of a Checkpoint's digest", "allOf": [ {
"$ref": "#/components/schemas/Digest" } ] }
```

CheckpointId

Any of

BigInt_for_uint64

CheckpointDigest

```
bash "CheckpointId": { "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, { "$ref":
"#/components/schemas/CheckpointDigest" } ] }
```

Claim

A claim consists of value and index_mod_4.

O bject

```
bash "Claim": { "description": "A claim consists of value and index_mod_4.", "type": "object",
"required": [ "indexMod4", "value" ], "properties": { "indexMod4": { "type": "integer", "format":
"uint8", "minimum": 0 }, "value": { "type": "string" } } }
```

Coin

O bject

```
bash "Coin": { "type": "object", "required": [ "balance", "coinObjectId", "coinType", "digest",
"previousTransaction", "version" ], "properties": { "balance": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "coinObjectId": { "$ref":
"#/components/schemas/ObjectID" }, "coinType": { "type": "string" }, "digest": { "$ref":
"#/components/schemas/ObjectDigest" }, "previousTransaction": { "$ref":
"#/components/schemas/TransactionDigest" }, "version": { "$ref":
"#/components/schemas/SequenceNumber" } } }
```

CommitteeInfo

RPC representation of the [Committee] type.

O bject

```
bash "CommitteeInfo": { "description": "RPC representation of the [Committee] type.", "type":
"object", "required": [ "epoch", "validators" ], "properties": { "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "validators": { "type": "array", "items": { "type":
"array", "items": [ { "$ref": "#/components/schemas/AuthorityPublicKeyBytes" }, { "$ref":
"#/components/schemas/BigInt_for_uint64" } ], "maxItems": 2, "minItems": 2 } } } }
```

CompressedSignature

Unlike [enum Signature], [enum CompressedSignature] does not contain public key.

One of

Object

Object

Object

Object

Object

```
bash "CompressedSignature": { "description": "Unlike [enum Signature], [enum CompressedSignature]
does not contain public key.", "oneOf": [ { "type": "object", "required": [ "Ed25519" ],
"properties": { "Ed25519": { "$ref": "#/components/schemas/Base64" } }, "additionalProperties":
false }, { "type": "object", "required": [ "Secp256k1" ], "properties": { "Secp256k1": { "$ref":
"#/components/schemas/Base64" } }, "additionalProperties": false }, { "type": "object", "required":
[ "Secp256r1" ], "properties": { "Secp256r1": { "$ref": "#/components/schemas/Base64" } },
"additionalProperties": false }, { "type": "object", "required": [ "ZkLogin" ], "properties": {
"ZkLogin": { "$ref": "#/components/schemas/ZkLoginAuthenticatorAsBytes" } },
"additionalProperties": false }, { "type": "object", "required": [ "Passkey" ], "properties": {
"Passkey": { "$ref": "#/components/schemas/PasskeyAuthenticatorAsBytes" } },
"additionalProperties": false } ] }
```

ConsensusCommitDigest

```
bash "ConsensusCommitDigest": { "$ref": "#/components/schemas/Digest" }
```

ConsensusDeterminedVersionAssignments

Uses an enum to allow for future expansion of the ConsensusDeterminedVersionAssignments.

One of

Object

Object

```
bash "ConsensusDeterminedVersionAssignments": { "description": "Uses an enum to allow for future
expansion of the ConsensusDeterminedVersionAssignments.", "oneOf": [ { "type": "object",
"required": [ "CancelledTransactions" ], "properties": { "CancelledTransactions": { "type":
"array", "items": { "type": "array", "items": [ { "$ref": "#/components/schemas/TransactionDigest"
}, { "type": "array", "items": { "type": "array", "items": [ { "$ref":
"#/components/schemas/ObjectID" }, { "$ref": "#/components/schemas/SequenceNumber2" } ],
"maxItems": 2, "minItems": 2 } } ], "maxItems": 2, "minItems": 2 } } }, "additionalProperties":
false }, { "type": "object", "required": [ "CancelledTransactionsV2" ], "properties": {
"CancelledTransactionsV2": { "type": "array", "items": { "type": "array", "items": [ { "$ref":
"#/components/schemas/TransactionDigest" }, { "type": "array", "items": { "type": "array", "items":
[ { "type": "array", "items": [ { "$ref": "#/components/schemas/ObjectID" }, { "$ref":
"#/components/schemas/SequenceNumber2" } ], "maxItems": 2, "minItems": 2 }, { "$ref":
"#/components/schemas/SequenceNumber2" } ], "maxItems": 2, "minItems": 2 } } ], "maxItems": 2,
"minItems": 2 } } }, "additionalProperties": false } ] }
```

Data

One of

Object

Object

```
bash "Data": { "oneOf": [ { "type": "object", "required": [ "dataType", "fields",
"hasPublicTransfer", "type" ], "properties": { "dataType": { "type": "string", "enum": [
"moveObject" ] }, "fields": { "$ref": "#/components/schemas/MoveStruct" }, "hasPublicTransfer": {
"type": "boolean" }, "type": { "type": "string" } } }, { "type": "object", "required": [
"dataType", "disassembled" ], "properties": { "dataType": { "type": "string", "enum": [ "package" ]
}, "disassembled": { "type": "object", "additionalProperties": true } } } ] }
```

DelegatedStake

O bject

ObjectID

SuiAddress

```
bash "DelegatedStake": { "type": "object", "required": [ "stakes", "stakingPool",
"validatorAddress" ], "properties": { "stakes": { "type": "array", "items": { "$ref":
"#/components/schemas/Stake" } }, "stakingPool": { "description": "Staking pool object id.",
"allOf": [ { "$ref": "#/components/schemas/ObjectID" } ] }, "validatorAddress": { "description":
"Validator's Address.", "allOf": [ { "$ref": "#/components/schemas/SuiAddress" } ] } } }
```

DevInspectArgs

Additional rguments supplied to dev inspect beyond what is allowed in today's API.

O bject

```
bash "DevInspectArgs": { "description": "Additional rguments supplied to dev inspect beyond what is
allowed in today's API.", "type": "object", "properties": { "gasBudget": { "description": "The gas
budget for the transaction.", "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, {
"type": "null" } ] }, "gasObjects": { "description": "The gas objects used to pay for the
transaction.", "type": [ "array", "null" ], "items": { "type": "array", "items": [ { "$ref":
"#/components/schemas/ObjectID" }, { "$ref": "#/components/schemas/SequenceNumber2" }, { "$ref":
"#/components/schemas/ObjectDigest" } ], "maxItems": 3, "minItems": 3 } }, "gasSponsor": {
"description": "The sponsor of the gas for the transaction, might be different from the sender.",
"anyOf": [ { "$ref": "#/components/schemas/SuiAddress" }, { "type": "null" } ] },
"showRawTxnDataAndEffects": { "description": "Whether to return the raw transaction data and
effects.", "type": [ "boolean", "null" ] }, "skipChecks": { "description": "Whether to skip
transaction checks for the transaction.", "type": [ "boolean", "null" ] } } }
```

DevInspectResults

The response from processing a dev inspect transaction

O bject

TransactionBlockEffects

```
bash "DevInspectResults": { "description": "The response from processing a dev inspect
transaction", "type": "object", "required": [ "effects", "events" ], "properties": { "effects": {
"description": "Summary of effects that likely would be generated if the transaction is actually
run. Note however, that not all dev-inspect transactions are actually usable as transactions so it
might not be possible actually generate these effects from a normal transaction.", "allOf": [ {
"$ref": "#/components/schemas/TransactionBlockEffects" } ] }, "error": { "description": "Execution
error from executing the transactions", "type": [ "string", "null" ] }, "events": { "description":
"Events that likely would be generated if the transaction is actually run.", "type": "array",
"items": { "$ref": "#/components/schemas/Event" } }, "rawEffects": { "description": "The raw
effects of the transaction that was dev inspected.", "type": "array", "items": { "type": "integer",
"format": "uint8", "minimum": 0 } }, "rawTxnData": { "description": "The raw transaction data that
was dev inspected.", "type": "array", "items": { "type": "integer", "format": "uint8", "minimum": 0
} }, "results": { "description": "Execution results (including return values) from executing the
transactions", "type": [ "array", "null" ], "items": { "$ref":
"#/components/schemas/SuiExecutionResult" } } } }
```

Digest

A representation of a 32 byte digest

Base58

```
bash "Digest": { "description": "A representation of a 32 byte digest", "allOf": [ { "$ref":
"#/components/schemas/Base58" } ] }
```

DisplayFieldsResponse

O bject

```
bash "DisplayFieldsResponse": { "type": "object", "properties": { "data": { "type": [ "object",
"null" ], "additionalProperties": { "type": "string" } }, "error": { "anyOf": [ { "$ref":
"#/components/schemas/ObjectResponseError" }, { "type": "null" } ] } } }
```

DryRunTransactionBlockResponse

O bject

```
bash "DryRunTransactionBlockResponse": { "type": "object", "required": [ "balanceChanges",
"effects", "events", "input", "objectChanges" ], "properties": { "balanceChanges": { "type":
"array", "items": { "$ref": "#/components/schemas/BalanceChange" } }, "effects": { "$ref":
"#/components/schemas/TransactionBlockEffects" }, "events": { "type": "array", "items": { "$ref":
"#/components/schemas/Event" } }, "executionErrorSource": { "type": [ "string", "null" ] },
"input": { "$ref": "#/components/schemas/TransactionBlockData" }, "objectChanges": { "type":
"array", "items": { "$ref": "#/components/schemas/ObjectChange" } }, "suggestedGasPrice": {
"anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, { "type": "null" } ] } } }
```

DynamicFieldInfo

O bject

One of

Object

Object

```
bash "DynamicFieldInfo": { "type": "object", "oneOf": [ { "type": "object", "required": [
"bcsEncoding", "bcsName" ], "properties": { "bcsEncoding": { "type": "string", "enum": [ "base64" ]
}, "bcsName": { "$ref": "#/components/schemas/Base64" } } }, { "type": "object", "required": [
"bcsEncoding", "bcsName" ], "properties": { "bcsEncoding": { "type": "string", "enum": [ "base58" ]
}, "bcsName": { "$ref": "#/components/schemas/Base58" } } } ], "required": [ "digest", "name",
"objectId", "objectType", "type", "version" ], "properties": { "digest": { "$ref":
"#/components/schemas/ObjectDigest" }, "name": { "$ref": "#/components/schemas/DynamicFieldName" },
"objectId": { "$ref": "#/components/schemas/ObjectID" }, "objectType": { "type": "string" },
"type": { "$ref": "#/components/schemas/DynamicFieldType" }, "version": { "$ref":
"#/components/schemas/SequenceNumber2" } } }
```

DynamicFieldName

O bject

```
bash "DynamicFieldName": { "type": "object", "required": [ "type", "value" ], "properties": {
"type": { "type": "string" }, "value": true } }
```

DynamicFieldType

S tring enum ["DynamicField" | "DynamicObject"]

```
bash "DynamicFieldType": { "type": "string", "enum": [ "DynamicField", "DynamicObject" ] }
```

ECMHLiveObjectSetDigest

The Sha256 digest of an EllipticCurveMultisetHash committing to the live object set.

O bject

```
bash "ECMHLiveObjectSetDigest": { "description": "The Sha256 digest of an EllipticCurveMultisetHash
committing to the live object set.", "type": "object", "required": [ "digest" ], "properties": {
"digest": { "type": "array", "items": { "type": "integer", "format": "uint8", "minimum": 0 },
"maxItems": 32, "minItems": 32 } } }
```

Ed25519SuiSignature

```
bash "Ed25519SuiSignature": { "$ref": "#/components/schemas/Base64" }
```

EndOfEpochData

O bject

ProtocolVersion

```
bash "EndOfEpochData": { "type": "object", "required": [ "epochCommitments", "nextEpochCommittee",
"nextEpochProtocolVersion" ], "properties": { "epochCommitments": { "description": "Commitments to
epoch specific state (e.g. live object set)", "type": "array", "items": { "$ref":
"#/components/schemas/CheckpointCommitment" } }, "nextEpochCommittee": { "description":
"next_epoch_committee is `Some` if and only if the current checkpoint is the last checkpoint of an
epoch. Therefore next_epoch_committee can be used to pick the last checkpoint of an epoch, which is
often useful to get epoch level summary stats like total gas cost of an epoch, or the total number
of transactions from genesis to the end of an epoch. The committee is stored as a vector of
validator pub key and stake pairs. The vector should be sorted based on the Committee data
structure.", "type": "array", "items": { "type": "array", "items": [ { "$ref":
"#/components/schemas/AuthorityPublicKeyBytes" }, { "$ref":
"#/components/schemas/BigInt_for_uint64" } ], "maxItems": 2, "minItems": 2 } },
"nextEpochProtocolVersion": { "description": "The protocol version that is in effect during the
epoch that starts immediately after this checkpoint.", "allOf": [ { "$ref":
"#/components/schemas/ProtocolVersion" } ] } } }
```

Event

O bject

EventID

ObjectID

SuiAddress

One of

Object

Object

```
bash "Event": { "type": "object", "oneOf": [ { "type": "object", "required": [ "bcs", "bcsEncoding"
], "properties": { "bcs": { "$ref": "#/components/schemas/Base64" }, "bcsEncoding": { "type":
"string", "enum": [ "base64" ] } } }, { "type": "object", "required": [ "bcs", "bcsEncoding" ],
"properties": { "bcs": { "$ref": "#/components/schemas/Base58" }, "bcsEncoding": { "type":
"string", "enum": [ "base58" ] } } } ], "required": [ "id", "packageId", "parsedJson", "sender",
"transactionModule", "type" ], "properties": { "id": { "description": "Sequential event ID, ie
(transaction seq number, event seq number). 1) Serves as a unique event ID for each fullnode 2)
Also serves to sequence events for the purposes of pagination and querying. A higher id is an event
seen later by that fullnode. This ID is the \"cursor\" for event querying.", "allOf": [ { "$ref":
"#/components/schemas/EventID" } ] }, "packageId": { "description": "Move package where this event
was emitted.", "allOf": [ { "$ref": "#/components/schemas/ObjectID" } ] }, "parsedJson": {
"description": "Parsed json value of the event" }, "sender": { "description": "Sender's Sui
address.", "allOf": [ { "$ref": "#/components/schemas/SuiAddress" } ] }, "timestampMs": {
"description": "UTC timestamp in milliseconds since epoch (1/1/1970)", "anyOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" }, { "type": "null" } ] }, "transactionModule": {
"description": "Move module where this event was emitted.", "type": "string" }, "type": {
```

"description": "Move event type.", "type": "string" } } }

EventFilter

One of

Object

Return all events.

Object

Return events that match any of the given filters. Only supported on event subscriptions.

Object

Query by sender address.

Object

Return events emitted by the given transaction.

Object

Return events emitted in a specified Move module. If the event is defined in Module A but emitted in a tx with Module B, query MoveModule by module B returns the event. Query MoveEventModule by module A returns the event too.

ObjectID

Object

Return events with the given Move event struct name (struct tag). For example, if the event is defined in 0xabcd::MyModule , and named Foo , then the struct tag is 0xabcd::MyModule::Foo .

Object

Return events with the given Move module name where the event struct is defined. If the event is defined in Module A but emitted in a tx with Module B, query MoveEventModule by module A returns the event. Query MoveModule by module B returns the event too.

ObjectID

Object

Return events emitted in [start_time, end_time] interval

BigInt_for_uint64

BigInt_for_uint64

bash "EventFilter": { "oneOf": [{ "description": "Return all events.", "type": "object", "required": ["All"], "properties": { "All": { "type": "array", "maxItems": 0 } }, "additionalProperties": false }, { "description": "Return events that match any of the given filters. Only supported on event subscriptions.", "type": "object", "required": ["Any"], "properties": { "Any": { "type": "array", "items": { "$ref": "#/components/schemas/EventFilter" } } }, "additionalProperties": false }, { "description": "Query by sender address.", "type": "object", "required": ["Sender"], "properties": { "Sender": { "$ref": "#/components/schemas/SuiAddress" } }, "additionalProperties": false }, { "description": "Return events emitted by the given transaction.", "type": "object", "required": ["Transaction"], "properties": { "Transaction": { "$ref": "#/components/schemas/TransactionDigest" } }, "additionalProperties": false }, { "description": "Return events emitted in a specified Move module. If the event is defined in Module A but emitted in a tx with Module B, query `MoveModule` by module B returns the event. Query `MoveEventModule` by module A returns the event too.", "type": "object", "required": ["MoveModule"], "properties": { "MoveModule": { "type": "object", "required": ["module", "package"], "properties": { "module": { "description": "the module name", "type": "string" }, "package": { "description": "the Move package ID", "allOf": [{ "$ref": "#/components/schemas/ObjectID" }] } } } }, "additionalProperties": false }, { "description": "Return events with the given Move event struct name (struct tag). For example, if the event is defined in `0xabcd::MyModule`, and named `Foo`, then the struct tag is `0xabcd::MyModule::Foo`.", "type": "object", "required": [

```
"MoveEventType" ], "properties": { "MoveEventType": { "type": "string" } }, "additionalProperties":
false }, { "description": "Return events with the given Move module name where the event struct is
defined. If the event is defined in Module A but emitted in a tx with Module B, query
`MoveEventModule` by module A returns the event. Query `MoveModule` by module B returns the event
too.", "type": "object", "required": [ "MoveEventModule" ], "properties": { "MoveEventModule": {
"type": "object", "required": [ "module", "package" ], "properties": { "module": { "description":
"the module name", "type": "string" }, "package": { "description": "the Move package ID", "allOf":
[ { "$ref": "#/components/schemas/ObjectID" } ] } } } }, "additionalProperties": false }, {
"description": "Return events emitted in [start_time, end_time] interval", "type": "object",
"required": [ "TimeRange" ], "properties": { "TimeRange": { "type": "object", "required": [
"endTime", "startTime" ], "properties": { "endTime": { "description": "right endpoint of time
interval, milliseconds since epoch, exclusive", "allOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" } ] }, "startTime": { "description": "left endpoint of
time interval, milliseconds since epoch, inclusive", "allOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" } ] } } } }, "additionalProperties": false } ] }
```

EventID

Unique ID of a Sui Event, the ID is a combination of transaction digest and event seq number.

O bject

```
bash "EventID": { "description": "Unique ID of a Sui Event, the ID is a combination of transaction
digest and event seq number.", "type": "object", "required": [ "eventSeq", "txDigest" ],
"properties": { "eventSeq": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "txDigest": {
"$ref": "#/components/schemas/TransactionDigest" } } }
```

ExecuteTransactionRequestType

S tring enum ["WaitForEffectsCert" | "WaitForLocalExecution"]

```
bash "ExecuteTransactionRequestType": { "type": "string", "enum": [ "WaitForEffectsCert",
"WaitForLocalExecution" ] }
```

ExecutionStatus

One of

Object

Object

```
bash "ExecutionStatus": { "oneOf": [ { "type": "object", "required": [ "status" ], "properties": {
"status": { "type": "string", "enum": [ "success" ] } } }, { "type": "object", "required": [
"error", "status" ], "properties": { "error": { "type": "string" }, "status": { "type": "string",
"enum": [ "failure" ] } } } ] }
```

GasCostSummary

Summary of the charges in a transaction. Storage is charged independently of computation. There are 3 parts to the storage charges: storage_cost : it is the charge of storage at the time the transaction is executed. The cost of storage is the number of bytes of the objects being mutated multiplied by a variable storage cost per byte storage_rebate : this is the amount a user gets back when manipulating an object. The storage_rebate is the storage_cost for an object minus fees. non_refundable_storage_fee : not all the value of the object storage cost is given back to user and there is a small fraction that is kept by the system. This value tracks that charge.

When looking at a gas cost summary the amount charged to the user is computation_cost + storage_cost - storage_rebate and that is the amount that is deducted from the gas coins. non_refundable_storage_fee is collected from the objects being mutated/deleted and it is tracked by the system in storage funds.

Objects deleted, including the older versions of objects mutated, have the storage field on the objects added up to a pool of "potential rebate". This rebate then is reduced by the "nonrefundable rate" such that: potential_rebate(storage cost of deleted/mutated objects) = storage_rebate + non_refundable_storage_fee

O bject

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

bash "GasCostSummary": { "description": "Summary of the charges in a transaction. Storage is charged independently of computation. There are 3 parts to the storage charges: `storage_cost`: it is the charge of storage at the time the transaction is executed. The cost of storage is the number of bytes of the objects being mutated multiplied by a variable storage cost per byte `storage_rebate`: this is the amount a user gets back when manipulating an object. The `storage_rebate` is the `storage_cost` for an object minus fees. `non_refundable_storage_fee`: not all the value of the object storage cost is given back to user and there is a small fraction that is kept by the system. This value tracks that charge.\n\nWhen looking at a gas cost summary the amount charged to the user is `computation_cost + storage_cost - storage_rebate` and that is the amount that is deducted from the gas coins. `non_refundable_storage_fee` is collected from the objects being mutated/deleted and it is tracked by the system in storage funds.\n\nObjects deleted, including the older versions of objects mutated, have the storage field on the objects added up to a pool of \"potential rebate\". This rebate then is reduced by the \"nonrefundable rate\" such that: `potential_rebate(storage cost of deleted/mutated objects) = storage_rebate + non_refundable_storage_fee`", "type": "object", "required": ["computationCost", "nonRefundableStorageFee", "storageCost", "storageRebate"], "properties": { "computationCost": { "description": "Cost of computation/execution", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "nonRefundableStorageFee": { "description": "The fee for the rebate. The portion of the storage rebate kept by the system.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "storageCost": { "description": "Storage cost, it's the sum of all storage cost for all objects created or mutated.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "storageRebate": { "description": "The amount of storage cost refunded to the user for all objects deleted or mutated in the transaction.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] } } }

GasData

O bject

bash "GasData": { "type": "object", "required": ["budget", "owner", "payment", "price"], "properties": { "budget": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "owner": { "$ref": "#/components/schemas/SuiAddress" }, "payment": { "type": "array", "items": { "$ref": "#/components/schemas/ObjectRef" } }, "price": { "$ref": "#/components/schemas/BigInt_for_uint64" } } }

GenericSignature

Due to the incompatibility of [enum Signature] (which dispatches a trait that assumes signature and pubkey bytes for verification), here we add a wrapper enum where member can just implement a lightweight [trait AuthenticatorTrait]. This way MultiSig (and future Authenticators) can implement its own verify .

One of

Object

Object

Object

Object

Object

bash "GenericSignature": { "description": "Due to the incompatibility of [enum Signature] (which dispatches a trait that assumes signature and pubkey bytes for verification), here we add a wrapper enum where member can just implement a lightweight [trait AuthenticatorTrait]. This way MultiSig (and future Authenticators) can implement its own `verify`.", "oneOf": [{ "type": "object", "required": ["MultiSig"], "properties": { "MultiSig": { "$ref": "#/components/schemas/MultiSig" } }, "additionalProperties": false }, { "type": "object", "required": ["MultiSigLegacy"], "properties": { "MultiSigLegacy": { "$ref": "#/components/schemas/MultiSigLegacy" } }, "additionalProperties": false }, { "type": "object", "required": ["Signature"], "properties": { "Signature": { "$ref": "#/components/schemas/Signature" } }, "additionalProperties": false }, { "type": "object", "required": ["ZkLoginAuthenticator"], "properties": { "ZkLoginAuthenticator": { "$ref": "#/components/schemas/ZkLoginAuthenticator" } }, "additionalProperties": false }, { "type": "object", "required": ["PasskeyAuthenticator"], "properties": { "PasskeyAuthenticator": { "$ref":

```
"#/components/schemas/PasskeyAuthenticator" } }, "additionalProperties": false } ] }
```

GetPastObjectRequest

O bject

ObjectID

SequenceNumber

```
bash "GetPastObjectRequest": { "type": "object", "required": [ "objectId", "version" ],
"properties": { "objectId": { "description": "the ID of the queried object", "allOf": [ { "$ref":
"#/components/schemas/ObjectID" } ] }, "version": { "description": "the version of the queried
object.", "allOf": [ { "$ref": "#/components/schemas/SequenceNumber" } ] } } }
```

Hex

Hex string encoding.

S tring

```
bash "Hex": { "description": "Hex string encoding.", "type": "string" }
```

InputObjectKind

One of

Object

Object

Object

```
bash "InputObjectKind": { "oneOf": [ { "type": "object", "required": [ "MovePackage" ],
"properties": { "MovePackage": { "$ref": "#/components/schemas/ObjectID" } },
"additionalProperties": false }, { "type": "object", "required": [ "ImmOrOwnedMoveObject" ],
"properties": { "ImmOrOwnedMoveObject": { "$ref": "#/components/schemas/ObjectRef" } },
"additionalProperties": false }, { "type": "object", "required": [ "SharedMoveObject" ],
"properties": { "SharedMoveObject": { "type": "object", "required": [ "id",
"initial_shared_version" ], "properties": { "id": { "$ref": "#/components/schemas/ObjectID" },
"initial_shared_version": { "$ref": "#/components/schemas/SequenceNumber2" }, "mutable": {
"default": true, "type": "boolean" } } } }, "additionalProperties": false } ] }
```

MoveCallParams

O bject

```
bash "MoveCallParams": { "type": "object", "required": [ "arguments", "function", "module",
"packageObjectId" ], "properties": { "arguments": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiJsonValue" } }, "function": { "type": "string" }, "module": { "type":
"string" }, "packageObjectId": { "$ref": "#/components/schemas/ObjectID" }, "typeArguments": {
"default": [], "type": "array", "items": { "$ref": "#/components/schemas/TypeTag" } } } }
```

MoveFunctionArgType

One of

String enum: ["Pure"]

Object

```
bash "MoveFunctionArgType": { "oneOf": [ { "type": "string", "enum": [ "Pure" ] }, { "type":
"object", "required": [ "Object" ], "properties": { "Object": { "$ref":
"#/components/schemas/ObjectValueKind" } }, "additionalProperties": false } ] }
```

MoveStruct

Any of

[MoveValue]

Object

Object

```
bash "MoveStruct": { "anyOf": [ { "type": "array", "items": { "$ref":
"#/components/schemas/MoveValue" } }, { "type": "object", "required": [ "fields", "type" ],
"properties": { "fields": { "type": "object", "additionalProperties": { "$ref":
"#/components/schemas/MoveValue" } }, "type": { "type": "string" } } }, { "type": "object",
"additionalProperties": { "$ref": "#/components/schemas/MoveValue" } } ] }
```

MoveValue

Any of

I nteger < uint32 > Minimum: 0

Boolean

SuiAddress

[MoveValue]

String

Object

MoveStruct

MoveVariant

```
bash "MoveValue": { "anyOf": [ { "type": "integer", "format": "uint32", "minimum": 0 }, { "type":
"boolean" }, { "$ref": "#/components/schemas/SuiAddress" }, { "type": "array", "items": { "$ref":
"#/components/schemas/MoveValue" } }, { "type": "string" }, { "type": "object", "required": [ "id"
], "properties": { "id": { "$ref": "#/components/schemas/ObjectID" } } }, { "$ref":
"#/components/schemas/MoveStruct" }, { "anyOf": [ { "$ref": "#/components/schemas/MoveValue" }, {
"type": "null" } ] }, { "$ref": "#/components/schemas/MoveVariant" } ] }
```

MoveVariant

O bject

```
bash "MoveVariant": { "type": "object", "required": [ "fields", "type", "variant" ], "properties":
{ "fields": { "type": "object", "additionalProperties": { "$ref": "#/components/schemas/MoveValue"
} }, "type": { "type": "string" }, "variant": { "type": "string" } } }
```

MultiSig

The struct that contains signatures and public keys necessary for authenticating a MultiSig.

O bject

MultiSigPublicKey

```
bash "MultiSig": { "description": "The struct that contains signatures and public keys necessary
for authenticating a MultiSig.", "type": "object", "required": [ "bitmap", "multisig_pk", "sigs" ],
"properties": { "bitmap": { "description": "A bitmap that indicates the position of which public
key the signature should be authenticated with.", "type": "integer", "format": "uint16", "minimum":
0 }, "multisig_pk": { "description": "The public key encoded with each public key with its
signature scheme used along with the corresponding weight.", "allOf": [ { "$ref":
"#/components/schemas/MultiSigPublicKey" } ] }, "sigs": { "description": "The plain signature
encoded with signature scheme.", "type": "array", "items": { "$ref":
"#/components/schemas/CompressedSignature" } } } }
```

MultiSigLegacy

Deprecated, use [struct MultiSig] instead. The struct that contains signatures and public keys necessary for authenticating a MultiSigLegacy.

O bject

Base64

MultiSigPublicKeyLegacy

```
bash "MultiSigLegacy": { "description": "Deprecated, use [struct MultiSig] instead. The struct that
contains signatures and public keys necessary for authenticating a MultiSigLegacy.", "type":
"object", "required": [ "bitmap", "multisig_pk", "sigs" ], "properties": { "bitmap": {
"description": "A bitmap that indicates the position of which public key the signature should be
authenticated with.", "allOf": [ { "$ref": "#/components/schemas/Base64" } ] }, "multisig_pk": {
"description": "The public key encoded with each public key with its signature scheme used along
with the corresponding weight.", "allOf": [ { "$ref":
"#/components/schemas/MultiSigPublicKeyLegacy" } ] }, "sigs": { "description": "The plain signature
encoded with signature scheme.", "type": "array", "items": { "$ref":
"#/components/schemas/CompressedSignature" } } } }
```

MultiSigPublicKey

The struct that contains the public key used for authenticating a MultiSig.

O bject

```
bash "MultiSigPublicKey": { "description": "The struct that contains the public key used for
authenticating a MultiSig.", "type": "object", "required": [ "pk_map", "threshold" ], "properties":
{ "pk_map": { "description": "A list of public key and its corresponding weight.", "type": "array",
"items": { "type": "array", "items": [ { "$ref": "#/components/schemas/PublicKey" }, { "type":
"integer", "format": "uint8", "minimum": 0 } ], "maxItems": 2, "minItems": 2 } }, "threshold": {
"description": "If the total weight of the public keys corresponding to verified signatures is
larger than threshold, the MultiSig is verified.", "type": "integer", "format": "uint16",
"minimum": 0 } } }
```

MultiSigPublicKeyLegacy

Deprecated, use [struct MultiSigPublicKey] instead. The struct that contains the public key used for authenticating a MultiSig.

O bject

```
bash "MultiSigPublicKeyLegacy": { "description": "Deprecated, use [struct MultiSigPublicKey]
instead. The struct that contains the public key used for authenticating a MultiSig.", "type":
"object", "required": [ "pk_map", "threshold" ], "properties": { "pk_map": { "description": "A list
of public key and its corresponding weight.", "type": "array", "items": { "type": "array", "items":
[ { "$ref": "#/components/schemas/PublicKey" }, { "type": "integer", "format": "uint8", "minimum":
0 } ], "maxItems": 2, "minItems": 2 } }, "threshold": { "description": "If the total weight of the
public keys corresponding to verified signatures is larger than threshold, the MultiSig is
verified.", "type": "integer", "format": "uint16", "minimum": 0 } } }
```

ObjectChange

ObjectChange are derived from the object mutations in the TransactionEffect to provide richer object information.

One of

Object

Module published

Object

Transfer objects to new address / wrap in another object

Object

Object mutated.

Object

Delete object

Object

Wrapped object

Object

New object creation

```
bash "ObjectChange": { "description": "ObjectChange are derived from the object mutations in the
TransactionEffect to provide richer object information.", "oneOf": [ { "description": "Module
published", "type": "object", "required": [ "digest", "modules", "packageId", "type", "version" ],
"properties": { "digest": { "$ref": "#/components/schemas/ObjectDigest" }, "modules": { "type":
"array", "items": { "type": "string" } }, "packageId": { "$ref": "#/components/schemas/ObjectID" },
"type": { "type": "string", "enum": [ "published" ] }, "version": { "$ref":
"#/components/schemas/SequenceNumber" } } }, { "description": "Transfer objects to new address /
wrap in another object", "type": "object", "required": [ "digest", "objectId", "objectType",
"recipient", "sender", "type", "version" ], "properties": { "digest": { "$ref":
"#/components/schemas/ObjectDigest" }, "objectId": { "$ref": "#/components/schemas/ObjectID" },
"objectType": { "type": "string" }, "recipient": { "$ref": "#/components/schemas/Owner" },
"sender": { "$ref": "#/components/schemas/SuiAddress" }, "type": { "type": "string", "enum": [
"transferred" ] }, "version": { "$ref": "#/components/schemas/SequenceNumber" } } }, {
"description": "Object mutated.", "type": "object", "required": [ "digest", "objectId",
"objectType", "owner", "previousVersion", "sender", "type", "version" ], "properties": { "digest":
{ "$ref": "#/components/schemas/ObjectDigest" }, "objectId": { "$ref":
"#/components/schemas/ObjectID" }, "objectType": { "type": "string" }, "owner": { "$ref":
"#/components/schemas/Owner" }, "previousVersion": { "$ref": "#/components/schemas/SequenceNumber"
}, "sender": { "$ref": "#/components/schemas/SuiAddress" }, "type": { "type": "string", "enum": [
"mutated" ] }, "version": { "$ref": "#/components/schemas/SequenceNumber" } } }, { "description":
"Delete object", "type": "object", "required": [ "objectId", "objectType", "sender", "type",
"version" ], "properties": { "objectId": { "$ref": "#/components/schemas/ObjectID" }, "objectType":
{ "type": "string" }, "sender": { "$ref": "#/components/schemas/SuiAddress" }, "type": { "type":
"string", "enum": [ "deleted" ] }, "version": { "$ref": "#/components/schemas/SequenceNumber" } }
}, { "description": "Wrapped object", "type": "object", "required": [ "objectId", "objectType",
"sender", "type", "version" ], "properties": { "objectId": { "$ref":
"#/components/schemas/ObjectID" }, "objectType": { "type": "string" }, "sender": { "$ref":
"#/components/schemas/SuiAddress" }, "type": { "type": "string", "enum": [ "wrapped" ] },
"version": { "$ref": "#/components/schemas/SequenceNumber" } } }, { "description": "New object
creation", "type": "object", "required": [ "digest", "objectId", "objectType", "owner", "sender",
"type", "version" ], "properties": { "digest": { "$ref": "#/components/schemas/ObjectDigest" },
"objectId": { "$ref": "#/components/schemas/ObjectID" }, "objectType": { "type": "string" },
"owner": { "$ref": "#/components/schemas/Owner" }, "sender": { "$ref":
"#/components/schemas/SuiAddress" }, "type": { "type": "string", "enum": [ "created" ] },
"version": { "$ref": "#/components/schemas/SequenceNumber" } } } ] }
```

ObjectData

O bject

ObjectDigest

SequenceNumber

```
bash "ObjectData": { "type": "object", "required": [ "digest", "objectId", "version" ],
"properties": { "bcs": { "description": "Move object content or package content in BCS, default to
be None unless SuiObjectDataOptions.showBcs is set to true", "anyOf": [ { "$ref":
"#/components/schemas/RawData" }, { "type": "null" } ] }, "content": { "description": "Move object
content or package content, default to be None unless SuiObjectDataOptions.showContent is set to
true", "anyOf": [ { "$ref": "#/components/schemas/Data" }, { "type": "null" } ] }, "digest": {
"description": "Base64 string representing the object digest", "allOf": [ { "$ref":
"#/components/schemas/ObjectDigest" } ] }, "display": { "description": "The Display metadata for
frontend UI rendering, default to be None unless SuiObjectDataOptions.showContent is set to true
This can also be None if the struct type does not have Display defined See more details in
<https://forums.sui.io/t/nft-object-display-proposal/4872>", "anyOf": [ { "$ref":
"#/components/schemas/DisplayFieldsResponse" }, { "type": "null" } ] }, "objectId": { "$ref":
"#/components/schemas/ObjectID" }, "owner": { "description": "The owner of this object. Default to
be None unless SuiObjectDataOptions.showOwner is set to true", "anyOf": [ { "$ref":
"#/components/schemas/Owner" }, { "type": "null" } ] }, "previousTransaction": { "description":
"The digest of the transaction that created or last mutated this object. Default to be None unless
SuiObjectDataOptions.showPreviousTransaction is set to true", "anyOf": [ { "$ref":
```

```
"#/components/schemas/TransactionDigest" }, { "type": "null" } ] }, "storageRebate": {
"description": "The amount of SUI we would rebate if this object gets deleted. This number is re-
calculated each time the object is mutated based on the present storage gas price.", "anyOf": [ {
"$ref": "#/components/schemas/BigInt_for_uint64" }, { "type": "null" } ] }, "type": {
"description": "The type of the object. Default to be None unless SuiObjectDataOptions.showType is
set to true", "type": [ "string", "null" ] }, "version": { "description": "Object version.",
"allOf": [ { "$ref": "#/components/schemas/SequenceNumber" } ] } } }
```

ObjectDataOptions

O bject

```
bash "ObjectDataOptions": { "type": "object", "properties": { "showBcs": { "description": "Whether
to show the content in BCS format. Default to be False", "default": false, "type": "boolean" },
"showContent": { "description": "Whether to show the content(i.e., package content or Move struct
content) of the object. Default to be False", "default": false, "type": "boolean" }, "showDisplay":
{ "description": "Whether to show the Display metadata of the object for frontend rendering.
Default to be False", "default": false, "type": "boolean" }, "showOwner": { "description": "Whether
to show the owner of the object. Default to be False", "default": false, "type": "boolean" },
"showPreviousTransaction": { "description": "Whether to show the previous transaction digest of the
object. Default to be False", "default": false, "type": "boolean" }, "showStorageRebate": {
"description": "Whether to show the storage rebate of the object. Default to be False", "default":
false, "type": "boolean" }, "showType": { "description": "Whether to show the type of the object.
Default to be False", "default": false, "type": "boolean" } } }
```

ObjectDigest

```
bash "ObjectDigest": { "$ref": "#/components/schemas/Digest" }
```

ObjectID

```
bash "ObjectID": { "$ref": "#/components/schemas/Hex" }
```

ObjectRead

One of

Object

The object exists and is found with this version

Object

The object does not exist

Object

The object is found to be deleted with this version

Object

The object exists but not found with this version

Object

The asked object version is higher than the latest

```
bash "ObjectRead": { "oneOf": [ { "description": "The object exists and is found with this
version", "type": "object", "required": [ "details", "status" ], "properties": { "details": {
"$ref": "#/components/schemas/ObjectData" }, "status": { "type": "string", "enum": [ "VersionFound"
] } } }, { "description": "The object does not exist", "type": "object", "required": [ "details",
"status" ], "properties": { "details": { "$ref": "#/components/schemas/ObjectID" }, "status": {
"type": "string", "enum": [ "ObjectNotExists" ] } } }, { "description": "The object is found to be
deleted with this version", "type": "object", "required": [ "details", "status" ], "properties": {
"details": { "$ref": "#/components/schemas/ObjectRef" }, "status": { "type": "string", "enum": [
"ObjectDeleted" ] } } }, { "description": "The object exists but not found with this version",
"type": "object", "required": [ "details", "status" ], "properties": { "details": { "type":
"array", "items": [ { "$ref": "#/components/schemas/ObjectID" }, { "$ref":
"#/components/schemas/SequenceNumber2" } ], "maxItems": 2, "minItems": 2 }, "status": { "type":
```

```
"string", "enum": [ "VersionNotFound" ] } } }, { "description": "The asked object version is higher
than the latest", "type": "object", "required": [ "details", "status" ], "properties": { "details":
{ "type": "object", "required": [ "asked_version", "latest_version", "object_id" ], "properties": {
"asked_version": { "$ref": "#/components/schemas/SequenceNumber2" }, "latest_version": { "$ref":
"#/components/schemas/SequenceNumber2" }, "object_id": { "$ref": "#/components/schemas/ObjectID" }
} }, "status": { "type": "string", "enum": [ "VersionTooHigh" ] } } } ] }
```

ObjectRef

O bject

ObjectDigest

ObjectID

SequenceNumber

```
bash "ObjectRef": { "type": "object", "required": [ "digest", "objectId", "version" ],
"properties": { "digest": { "description": "Base64 string representing the object digest", "allOf":
[ { "$ref": "#/components/schemas/ObjectDigest" } ] }, "objectId": { "description": "Hex code as
string representing the object id", "allOf": [ { "$ref": "#/components/schemas/ObjectID" } ] },
"version": { "description": "Object version.", "allOf": [ { "$ref":
"#/components/schemas/SequenceNumber" } ] } } }
```

ObjectResponseError

One of

Object

Object

Object

ObjectDigest

SequenceNumber2

Object

Object

```
bash "ObjectResponseError": { "oneOf": [ { "type": "object", "required": [ "code", "object_id" ],
"properties": { "code": { "type": "string", "enum": [ "notExists" ] }, "object_id": { "$ref":
"#/components/schemas/ObjectID" } } }, { "type": "object", "required": [ "code", "parent_object_id"
], "properties": { "code": { "type": "string", "enum": [ "dynamicFieldNotFound" ] },
"parent_object_id": { "$ref": "#/components/schemas/ObjectID" } } }, { "type": "object",
"required": [ "code", "digest", "object_id", "version" ], "properties": { "code": { "type":
"string", "enum": [ "deleted" ] }, "digest": { "description": "Base64 string representing the
object digest", "allOf": [ { "$ref": "#/components/schemas/ObjectDigest" } ] }, "object_id": {
"$ref": "#/components/schemas/ObjectID" }, "version": { "description": "Object version.", "allOf":
[ { "$ref": "#/components/schemas/SequenceNumber2" } ] } } }, { "type": "object", "required": [
"code" ], "properties": { "code": { "type": "string", "enum": [ "unknown" ] } } }, { "type":
"object", "required": [ "code", "error" ], "properties": { "code": { "type": "string", "enum": [
"displayError" ] }, "error": { "type": "string" } } } ] }
```

ObjectResponseQuery

O bject

```
bash "ObjectResponseQuery": { "type": "object", "properties": { "filter": { "description": "If
None, no filter will be applied", "default": null, "anyOf": [ { "$ref":
"#/components/schemas/SuiObjectDataFilter" }, { "type": "null" } ] }, "options": { "description":
"config which fields to include in the response, by default only digest is included", "default":
null, "anyOf": [ { "$ref": "#/components/schemas/ObjectDataOptions" }, { "type": "null" } ] } } }
```

ObjectValueKind

S tring enum ["ByImmutableReference" | "ByMutableReference" | "ByValue"]

```
bash "ObjectValueKind": { "type": "string", "enum": [ "ByImmutableReference", "ByMutableReference",
"ByValue" ] }
```

OwnedObjectRef

O bject

```
bash "OwnedObjectRef": { "type": "object", "required": [ "owner", "reference" ], "properties": {
"owner": { "$ref": "#/components/schemas/Owner" }, "reference": { "$ref":
"#/components/schemas/ObjectRef" } } }
```

Owner

One of

Object

Object is exclusively owned by a single address, and is mutable.

Object

Object is exclusively owned by a single object, and is mutable. The object ID is converted to SuiAddress as SuiAddress is universal.

Object

Object is shared, can be used by any address, and is mutable.

SequenceNumber2

String enum: ["Immutable"]

Object is immutable, and hence ownership doesn't matter.

Object

Object is sequenced via consensus. Ownership is managed by the configured authenticator.

Note: wondering what happened to V1 ? Shared above was the V1 of consensus objects.

Authenticator

SequenceNumber2

```
bash "Owner": { "oneOf": [ { "description": "Object is exclusively owned by a single address, and
is mutable.", "type": "object", "required": [ "AddressOwner" ], "properties": { "AddressOwner": {
"$ref": "#/components/schemas/SuiAddress" } }, "additionalProperties": false }, { "description":
"Object is exclusively owned by a single object, and is mutable. The object ID is converted to
SuiAddress as SuiAddress is universal.", "type": "object", "required": [ "ObjectOwner" ],
"properties": { "ObjectOwner": { "$ref": "#/components/schemas/SuiAddress" } },
"additionalProperties": false }, { "description": "Object is shared, can be used by any address,
and is mutable.", "type": "object", "required": [ "Shared" ], "properties": { "Shared": { "type":
"object", "required": [ "initial_shared_version" ], "properties": { "initial_shared_version": {
"description": "The version at which the object became shared", "allOf": [ { "$ref":
"#/components/schemas/SequenceNumber2" } ] } } } }, "additionalProperties": false }, {
"description": "Object is immutable, and hence ownership doesn't matter.", "type": "string",
"enum": [ "Immutable" ] }, { "description": "Object is sequenced via consensus. Ownership is
managed by the configured authenticator.\n\nNote: wondering what happened to `V1`? `Shared` above
was the V1 of consensus objects.", "type": "object", "required": [ "ConsensusV2" ], "properties": {
"ConsensusV2": { "type": "object", "required": [ "authenticator", "start_version" ], "properties":
{ "authenticator": { "description": "The authentication mode of the object", "allOf": [ { "$ref":
"#/components/schemas/Authenticator" } ] }, "start_version": { "description": "The version at which
the object most recently became a consensus object. This serves the same function as
`initial_shared_version`, except it may change if the object's Owner type changes.", "allOf": [ {
"$ref": "#/components/schemas/SequenceNumber2" } ] } } } }, "additionalProperties": false } ] }
```

Page_for_Checkpoint_and_BigInt_for_uint64

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

```
bash "Page_for_Checkpoint_and_BigInt_for_uint64": { "description": "`next_cursor` points to the
last item in the page; Reading with `next_cursor` will start from the next item after `next_cursor`
if `next_cursor` is `Some`, otherwise it will start from the first item.", "type": "object",
"required": [ "data", "hasNextPage" ], "properties": { "data": { "type": "array", "items": {
"$ref": "#/components/schemas/Checkpoint" } }, "hasNextPage": { "type": "boolean" }, "nextCursor":
{ "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, { "type": "null" } ] } } } }
```

Page_for_Coin_and_String

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

```
bash "Page_for_Coin_and_String": { "description": "`next_cursor` points to the last item in the
page; Reading with `next_cursor` will start from the next item after `next_cursor` if `next_cursor`
is `Some`, otherwise it will start from the first item.", "type": "object", "required": [ "data",
"hasNextPage" ], "properties": { "data": { "type": "array", "items": { "$ref":
"#/components/schemas/Coin" } }, "hasNextPage": { "type": "boolean" }, "nextCursor": { "type": [
"string", "null" ] } } } }
```

Page_for_DynamicFieldInfo_and_ObjectID

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

```
bash "Page_for_DynamicFieldInfo_and_ObjectID": { "description": "`next_cursor` points to the last
item in the page; Reading with `next_cursor` will start from the next item after `next_cursor` if
`next_cursor` is `Some`, otherwise it will start from the first item.", "type": "object",
"required": [ "data", "hasNextPage" ], "properties": { "data": { "type": "array", "items": {
"$ref": "#/components/schemas/DynamicFieldInfo" } }, "hasNextPage": { "type": "boolean" },
"nextCursor": { "anyOf": [ { "$ref": "#/components/schemas/ObjectID" }, { "type": "null" } ] } } } }
```

Page_for_Event_and_EventID

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

```
bash "Page_for_Event_and_EventID": { "description": "`next_cursor` points to the last item in the
page; Reading with `next_cursor` will start from the next item after `next_cursor` if `next_cursor`
is `Some`, otherwise it will start from the first item.", "type": "object", "required": [ "data",
"hasNextPage" ], "properties": { "data": { "type": "array", "items": { "$ref":
"#/components/schemas/Event" } }, "hasNextPage": { "type": "boolean" }, "nextCursor": { "anyOf": [
{ "$ref": "#/components/schemas/EventID" }, { "type": "null" } ] } } } }
```

Page_for_String_and_ObjectID

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

O bject

```
bash "Page_for_String_and_ObjectID": { "description": "`next_cursor` points to the last item in the
page; Reading with `next_cursor` will start from the next item after `next_cursor` if `next_cursor`
is `Some`, otherwise it will start from the first item.", "type": "object", "required": [ "data",
"hasNextPage" ], "properties": { "data": { "type": "array", "items": { "type": "string" } },
"hasNextPage": { "type": "boolean" }, "nextCursor": { "anyOf": [ { "$ref":
"#/components/schemas/ObjectID" }, { "type": "null" } ] } } } }
```

Page_for_SuiObjectResponse_and_ObjectID

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Object

```
bash "Page_for_SuiObjectResponse_and_ObjectID": { "description": "`next_cursor` points to the last
item in the page; Reading with `next_cursor` will start from the next item after `next_cursor` if
`next_cursor` is `Some`, otherwise it will start from the first item.", "type": "object",
"required": [ "data", "hasNextPage" ], "properties": { "data": { "type": "array", "items": {
"$ref": "#/components/schemas/SuiObjectResponse" } }, "hasNextPage": { "type": "boolean" },
"nextCursor": { "anyOf": [ { "$ref": "#/components/schemas/ObjectID" }, { "type": "null" } ] } } }
```

Page_for_TransactionBlockResponse_and_TransactionDigest

next_cursor points to the last item in the page; Reading with next_cursor will start from the next item after next_cursor if next_cursor is Some , otherwise it will start from the first item.

Object

```
bash "Page_for_TransactionBlockResponse_and_TransactionDigest": { "description": "`next_cursor`
points to the last item in the page; Reading with `next_cursor` will start from the next item after
`next_cursor` if `next_cursor` is `Some`, otherwise it will start from the first item.", "type":
"object", "required": [ "data", "hasNextPage" ], "properties": { "data": { "type": "array",
"items": { "$ref": "#/components/schemas/TransactionBlockResponse" } }, "hasNextPage": { "type":
"boolean" }, "nextCursor": { "anyOf": [ { "$ref": "#/components/schemas/TransactionDigest" }, {
"type": "null" } ] } } }
```

PasskeyAuthenticator

An passkey authenticator with parsed fields. See field defition below. Can be initialized from [struct RawPasskeyAuthenticator].

Object

```
bash "PasskeyAuthenticator": { "description": "An passkey authenticator with parsed fields. See
field defition below. Can be initialized from [struct RawPasskeyAuthenticator].", "type": "object",
"required": [ "authenticator_data", "client_data_json" ], "properties": { "authenticator_data": {
"description": "`authenticatorData` is a bytearray that encodes [Authenticator Data]
(https://www.w3.org/TR/webauthn-2/#sctn-authenticator-data) structure returned by the authenticator
attestation response as is.", "type": "array", "items": { "type": "integer", "format": "uint8",
"minimum": 0 } }, "client_data_json": { "description": "`clientDataJSON` contains a JSON-compatible
UTF-8 encoded string of the client data which is passed to the authenticator by the client during
the authentication request (see [CollectedClientData](https://www.w3.org/TR/webauthn-2/#dictdef-
collectedclientdata))", "type": "string" } } }
```

PasskeyAuthenticatorAsBytes

```
bash "PasskeyAuthenticatorAsBytes": { "$ref": "#/components/schemas/Base64" }
```

ProtocolConfig

Object

```
bash "ProtocolConfig": { "type": "object", "required": [ "attributes", "featureFlags",
"maxSupportedProtocolVersion", "minSupportedProtocolVersion", "protocolVersion" ], "properties": {
"attributes": { "type": "object", "additionalProperties": { "anyOf": [ { "$ref":
"#/components/schemas/ProtocolConfigValue" }, { "type": "null" } ] } }, "featureFlags": { "type":
"object", "additionalProperties": { "type": "boolean" } }, "maxSupportedProtocolVersion": { "$ref":
"#/components/schemas/ProtocolVersion" }, "minSupportedProtocolVersion": { "$ref":
"#/components/schemas/ProtocolVersion" }, "protocolVersion": { "$ref":
"#/components/schemas/ProtocolVersion" } } }
```

ProtocolConfigValue

One of

Object

Object

Object

Object

Object

```
bash "ProtocolConfigValue": { "oneOf": [ { "type": "object", "required": [ "u16" ], "properties": {
"u16": { "$ref": "#/components/schemas/BigInt_for_uint16" } }, "additionalProperties": false }, {
"type": "object", "required": [ "u32" ], "properties": { "u32": { "$ref":
"#/components/schemas/BigInt_for_uint32" } }, "additionalProperties": false }, { "type": "object",
"required": [ "u64" ], "properties": { "u64": { "$ref": "#/components/schemas/BigInt_for_uint64" }
}, "additionalProperties": false }, { "type": "object", "required": [ "f64" ], "properties": {
"f64": { "type": "string" } }, "additionalProperties": false }, { "type": "object", "required": [
"bool" ], "properties": { "bool": { "type": "string" } }, "additionalProperties": false } ] }
```

ProtocolVersion

```
bash "ProtocolVersion": { "$ref": "#/components/schemas/BigInt_for_uint64" }
```

PublicKey

One of

Object

Object

Object

Object

Object

```
bash "PublicKey": { "oneOf": [ { "type": "object", "required": [ "Ed25519" ], "properties": {
"Ed25519": { "$ref": "#/components/schemas/Base64" } }, "additionalProperties": false }, { "type":
"object", "required": [ "Secp256k1" ], "properties": { "Secp256k1": { "$ref":
"#/components/schemas/Base64" } }, "additionalProperties": false }, { "type": "object", "required":
[ "Secp256r1" ], "properties": { "Secp256r1": { "$ref": "#/components/schemas/Base64" } },
"additionalProperties": false }, { "type": "object", "required": [ "ZkLogin" ], "properties": {
"ZkLogin": { "$ref": "#/components/schemas/ZkLoginPublicIdentifier" } }, "additionalProperties":
false }, { "type": "object", "required": [ "Passkey" ], "properties": { "Passkey": { "$ref":
"#/components/schemas/Base64" } }, "additionalProperties": false } ] }
```

RPCTransactionRequestParams

One of

Object

Object

```
bash "RPCTransactionRequestParams": { "oneOf": [ { "type": "object", "required": [
"transferObjectRequestParams" ], "properties": { "transferObjectRequestParams": { "$ref":
"#/components/schemas/TransferObjectParams" } }, "additionalProperties": false }, { "type":
"object", "required": [ "moveCallRequestParams" ], "properties": { "moveCallRequestParams": {
"$ref": "#/components/schemas/MoveCallParams" } }, "additionalProperties": false } ] }
```

RawData

One of

Object

Object

```
bash "RawData": { "oneOf": [ { "type": "object", "required": [ "bcsBytes", "dataType",
"hasPublicTransfer", "type", "version" ], "properties": { "bcsBytes": { "$ref":
"#/components/schemas/Base64" }, "dataType": { "type": "string", "enum": [ "moveObject" ] },
"hasPublicTransfer": { "type": "boolean" }, "type": { "type": "string" }, "version": { "$ref":
"#/components/schemas/SequenceNumber2" } } }, { "type": "object", "required": [ "dataType", "id",
"linkageTable", "moduleMap", "typeOriginTable", "version" ], "properties": { "dataType": { "type":
"string", "enum": [ "package" ] }, "id": { "$ref": "#/components/schemas/ObjectID" },
"linkageTable": { "type": "object", "additionalProperties": { "$ref":
"#/components/schemas/UpgradeInfo" } }, "moduleMap": { "type": "object", "additionalProperties": {
"$ref": "#/components/schemas/Base64" } }, "typeOriginTable": { "type": "array", "items": { "$ref":
"#/components/schemas/TypeOrigin" } }, "version": { "$ref": "#/components/schemas/SequenceNumber2"
} } } ] }
```

Secp256k1SuiSignature

```
bash "Secp256k1SuiSignature": { "$ref": "#/components/schemas/Base64" }
```

Secp256r1SuiSignature

```
bash "Secp256r1SuiSignature": { "$ref": "#/components/schemas/Base64" }
```

SequenceNumber

Integer

```
bash "SequenceNumber": { "type": "integer", "format": "uint64", "minimum": 0 }
```

SequenceNumber2

```
bash "SequenceNumber2": { "$ref": "#/components/schemas/BigInt_for_uint64" }
```

Signature

One of

Object

Object

Object

```
bash "Signature": { "oneOf": [ { "type": "object", "required": [ "Ed25519SuiSignature" ],
"properties": { "Ed25519SuiSignature": { "$ref": "#/components/schemas/Ed25519SuiSignature" } },
"additionalProperties": false }, { "type": "object", "required": [ "Secp256k1SuiSignature" ],
"properties": { "Secp256k1SuiSignature": { "$ref": "#/components/schemas/Secp256k1SuiSignature" }
}, "additionalProperties": false }, { "type": "object", "required": [ "Secp256r1SuiSignature" ],
"properties": { "Secp256r1SuiSignature": { "$ref": "#/components/schemas/Secp256r1SuiSignature" }
}, "additionalProperties": false } ] }
```

Stake

Object

ObjectID

One of

Object

Object

Object

```
bash "Stake": { "type": "object", "oneOf": [ { "type": "object", "required": [ "status" ],
"properties": { "status": { "type": "string", "enum": [ "Pending" ] } } }, { "type": "object",
"required": [ "estimatedReward", "status" ], "properties": { "estimatedReward": { "$ref":
```

```
"#/components/schemas/BigInt_for_uint64" }, "status": { "type": "string", "enum": [ "Active" ] } }
}, { "type": "object", "required": [ "status" ], "properties": { "status": { "type": "string",
"enum": [ "Unstaked" ] } } } ], "required": [ "principal", "stakeActiveEpoch", "stakeRequestEpoch",
"stakedSuiId" ], "properties": { "principal": { "$ref": "#/components/schemas/BigInt_for_uint64" },
"stakeActiveEpoch": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "stakeRequestEpoch": {
"$ref": "#/components/schemas/BigInt_for_uint64" }, "stakedSuiId": { "description": "ID of the
StakedSui receipt object.", "allOf": [ { "$ref": "#/components/schemas/ObjectID" } ] } } } }
```

SuiActiveJwk

O bject

```
bash "SuiActiveJwk": { "type": "object", "required": [ "epoch", "jwk", "jwk_id" ], "properties": {
"epoch": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "jwk": { "$ref":
"#/components/schemas/SuiJWK" }, "jwk_id": { "$ref": "#/components/schemas/SuiJwkId" } } }
```

SuiAddress

```
bash "SuiAddress": { "$ref": "#/components/schemas/Hex" }
```

SuiArgument

An argument to a transaction in a programmable transaction block

One of

String enum: ["GasCoin"]

The gas coin. The gas coin can only be used by-ref, except for with TransferObjects , which can use it by-value.

Object

One of the input objects or primitive values (from ProgrammableTransactionBlock inputs)

Object

The result of another transaction (from ProgrammableTransactionBlock transactions)

Object

Like a Result but it accesses a nested result. Currently, the only usage of this is to access a value from a Move call with multiple return values.

```
bash "SuiArgument": { "description": "An argument to a transaction in a programmable transaction
block", "oneOf": [ { "description": "The gas coin. The gas coin can only be used by-ref, except for
with `TransferObjects`, which can use it by-value.", "type": "string", "enum": [ "GasCoin" ] }, {
"description": "One of the input objects or primitive values (from `ProgrammableTransactionBlock`
inputs)", "type": "object", "required": [ "Input" ], "properties": { "Input": { "type": "integer",
"format": "uint16", "minimum": 0 } }, "additionalProperties": false }, { "description": "The result
of another transaction (from `ProgrammableTransactionBlock` transactions)", "type": "object",
"required": [ "Result" ], "properties": { "Result": { "type": "integer", "format": "uint16",
"minimum": 0 } }, "additionalProperties": false }, { "description": "Like a `Result` but it
accesses a nested result. Currently, the only usage of this is to access a value from a Move call
with multiple return values.", "type": "object", "required": [ "NestedResult" ], "properties": {
"NestedResult": { "type": "array", "items": [ { "type": "integer", "format": "uint16", "minimum": 0
}, { "type": "integer", "format": "uint16", "minimum": 0 } ], "maxItems": 2, "minItems": 2 } },
"additionalProperties": false } ] }
```

SuiAuthenticatorStateExpire

O bject

```
bash "SuiAuthenticatorStateExpire": { "type": "object", "required": [ "min_epoch" ], "properties":
{ "min_epoch": { "$ref": "#/components/schemas/BigInt_for_uint64" } } }
```

SuiCallArg

One of

Object

Object

```
bash "SuiCallArg": { "oneOf": [ { "type": "object", "oneOf": [ { "type": "object", "required": [
"digest", "objectId", "objectType", "version" ], "properties": { "digest": { "$ref":
"#/components/schemas/ObjectDigest" }, "objectId": { "$ref": "#/components/schemas/ObjectID" },
"objectType": { "type": "string", "enum": [ "immOrOwnedObject" ] }, "version": { "$ref":
"#/components/schemas/SequenceNumber" } } }, { "type": "object", "required": [
"initialSharedVersion", "mutable", "objectId", "objectType" ], "properties": {
"initialSharedVersion": { "$ref": "#/components/schemas/SequenceNumber" }, "mutable": { "type":
"boolean" }, "objectId": { "$ref": "#/components/schemas/ObjectID" }, "objectType": { "type":
"string", "enum": [ "sharedObject" ] } } }, { "type": "object", "required": [ "digest", "objectId",
"objectType", "version" ], "properties": { "digest": { "$ref": "#/components/schemas/ObjectDigest"
}, "objectId": { "$ref": "#/components/schemas/ObjectID" }, "objectType": { "type": "string",
"enum": [ "receiving" ] }, "version": { "$ref": "#/components/schemas/SequenceNumber" } } } ],
"required": [ "type" ], "properties": { "type": { "type": "string", "enum": [ "object" ] } } }, {
"type": "object", "required": [ "type", "value" ], "properties": { "type": { "type": "string",
"enum": [ "pure" ] }, "value": { "$ref": "#/components/schemas/SuiJsonValue" }, "valueType": {
"default": null, "type": [ "string", "null" ] } } } ] }
```

SuiChangeEpoch

Object

```
bash "SuiChangeEpoch": { "type": "object", "required": [ "computation_charge", "epoch",
"epoch_start_timestamp_ms", "storage_charge", "storage_rebate" ], "properties": {
"computation_charge": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "epoch_start_timestamp_ms": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "storage_charge": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "storage_rebate": { "$ref":
"#/components/schemas/BigInt_for_uint64" } } }
```

SuiCoinMetadata

Object

```
bash "SuiCoinMetadata": { "type": "object", "required": [ "decimals", "description", "name",
"symbol" ], "properties": { "decimals": { "description": "Number of decimal places the coin uses.",
"type": "integer", "format": "uint8", "minimum": 0 }, "description": { "description": "Description
of the token", "type": "string" }, "iconUrl": { "description": "URL for the token logo", "type": [
"string", "null" ] }, "id": { "description": "Object id for the CoinMetadata object", "anyOf": [ {
"$ref": "#/components/schemas/ObjectID" }, { "type": "null" } ] }, "name": { "description": "Name
for the token", "type": "string" }, "symbol": { "description": "Symbol for the token", "type":
"string" } } }
```

SuiEndOfEpochTransactionKind

One of

String enum: ["AuthenticatorStateCreate" | "RandomnessStateCreate" | "CoinDenyListStateCreate" | "StoreExecutionTimeObservations"]

Object

Object

Object

Object

```
bash "SuiEndOfEpochTransactionKind": { "oneOf": [ { "type": "string", "enum": [
"AuthenticatorStateCreate", "RandomnessStateCreate", "CoinDenyListStateCreate",
"StoreExecutionTimeObservations" ] }, { "type": "object", "required": [ "ChangeEpoch" ],
"properties": { "ChangeEpoch": { "$ref": "#/components/schemas/SuiChangeEpoch" } },
"additionalProperties": false }, { "type": "object", "required": [ "AuthenticatorStateExpire" ],
"properties": { "AuthenticatorStateExpire": { "$ref":
"#/components/schemas/SuiAuthenticatorStateExpire" } }, "additionalProperties": false }, { "type":
```

```
"object", "required": [ "BridgeStateCreate" ], "properties": { "BridgeStateCreate": { "$ref":
"#/components/schemas/CheckpointDigest" } }, "additionalProperties": false }, { "type": "object",
"required": [ "BridgeCommitteeUpdate" ], "properties": { "BridgeCommitteeUpdate": { "$ref":
"#/components/schemas/SequenceNumber2" } }, "additionalProperties": false } ] }
```

SuiExecutionResult

O bject

```
bash "SuiExecutionResult": { "type": "object", "properties": { "mutableReferenceOutputs": {
"description": "The value of any arguments that were mutably borrowed. Non-mut borrowed values are
not included", "type": "array", "items": { "type": "array", "items": [ { "$ref":
"#/components/schemas/SuiArgument" }, { "type": "array", "items": { "type": "integer", "format":
"uint8", "minimum": 0 } }, { "$ref": "#/components/schemas/TypeTag" } ], "maxItems": 3, "minItems":
3 } }, "returnValues": { "description": "The return values from the transaction", "type": "array",
"items": { "type": "array", "items": [ { "type": "array", "items": { "type": "integer", "format":
"uint8", "minimum": 0 } }, { "$ref": "#/components/schemas/TypeTag" } ], "maxItems": 2, "minItems":
2 } } } }
```

SuiJWK

O bject

```
bash "SuiJWK": { "type": "object", "required": [ "alg", "e", "kty", "n" ], "properties": { "alg": {
"type": "string" }, "e": { "type": "string" }, "kty": { "type": "string" }, "n": { "type": "string"
} } }
```

SuiJsonValue

```
bash "SuiJsonValue": {}
```

SuiJwkId

O bject

```
bash "SuiJwkId": { "type": "object", "required": [ "iss", "kid" ], "properties": { "iss": { "type":
"string" }, "kid": { "type": "string" } } }
```

SuiMoveAbility

S tring enum ["Copy" | "Drop" | "Store" | "Key"]

```
bash "SuiMoveAbility": { "type": "string", "enum": [ "Copy", "Drop", "Store", "Key" ] }
```

SuiMoveAbilitySet

O bject

```
bash "SuiMoveAbilitySet": { "type": "object", "required": [ "abilities" ], "properties": {
"abilities": { "type": "array", "items": { "$ref": "#/components/schemas/SuiMoveAbility" } } } }
```

SuiMoveModuleId

O bject

```
bash "SuiMoveModuleId": { "type": "object", "required": [ "address", "name" ], "properties": {
"address": { "type": "string" }, "name": { "type": "string" } } }
```

SuiMoveNormalizedEnum

O bject

```
bash "SuiMoveNormalizedEnum": { "type": "object", "required": [ "abilities", "typeParameters",
"variants" ], "properties": { "abilities": { "$ref": "#/components/schemas/SuiMoveAbilitySet" },
"typeParameters": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiMoveStructTypeParameter" } }, "variants": { "type": "object",
"additionalProperties": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiMoveNormalizedField" } } } } }
```

SuiMoveNormalizedField

O bject

```
bash "SuiMoveNormalizedField": { "type": "object", "required": [ "name", "type" ], "properties": {
"name": { "type": "string" }, "type": { "$ref": "#/components/schemas/SuiMoveNormalizedType" } } }
```

SuiMoveNormalizedFunction

O bject

```
bash "SuiMoveNormalizedFunction": { "type": "object", "required": [ "isEntry", "parameters",
"return", "typeParameters", "visibility" ], "properties": { "isEntry": { "type": "boolean" },
"parameters": { "type": "array", "items": { "$ref": "#/components/schemas/SuiMoveNormalizedType" }
}, "return": { "type": "array", "items": { "$ref": "#/components/schemas/SuiMoveNormalizedType" }
}, "typeParameters": { "type": "array", "items": { "$ref": "#/components/schemas/SuiMoveAbilitySet"
} }, "visibility": { "$ref": "#/components/schemas/SuiMoveVisibility" } } }
```

SuiMoveNormalizedModule

O bject

```
bash "SuiMoveNormalizedModule": { "type": "object", "required": [ "address", "exposedFunctions",
"fileFormatVersion", "friends", "name", "structs" ], "properties": { "address": { "type": "string"
}, "enums": { "type": "object", "additionalProperties": { "$ref":
"#/components/schemas/SuiMoveNormalizedEnum" } }, "exposedFunctions": { "type": "object",
"additionalProperties": { "$ref": "#/components/schemas/SuiMoveNormalizedFunction" } },
"fileFormatVersion": { "type": "integer", "format": "uint32", "minimum": 0 }, "friends": { "type":
"array", "items": { "$ref": "#/components/schemas/SuiMoveModuleId" } }, "name": { "type": "string"
}, "structs": { "type": "object", "additionalProperties": { "$ref":
"#/components/schemas/SuiMoveNormalizedStruct" } } } }
```

SuiMoveNormalizedStruct

O bject

```
bash "SuiMoveNormalizedStruct": { "type": "object", "required": [ "abilities", "fields",
"typeParameters" ], "properties": { "abilities": { "$ref": "#/components/schemas/SuiMoveAbilitySet"
}, "fields": { "type": "array", "items": { "$ref": "#/components/schemas/SuiMoveNormalizedField" }
}, "typeParameters": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiMoveStructTypeParameter" } } } }
```

SuiMoveNormalizedType

One of

String enum: ["Bool" | "U8" | "U16" | "U32" | "U64" | "U128" | "U256" | "Address" | "Signer"]

Object

Object

Object

Object

Object

```
bash "SuiMoveNormalizedType": { "oneOf": [ { "type": "string", "enum": [ "Bool", "U8", "U16",
"U32", "U64", "U128", "U256", "Address", "Signer" ] }, { "type": "object", "required": [ "Struct"
], "properties": { "Struct": { "type": "object", "required": [ "address", "module", "name",
"typeArguments" ], "properties": { "address": { "type": "string" }, "module": { "type": "string" },
"name": { "type": "string" }, "typeArguments": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiMoveNormalizedType" } } } } }, "additionalProperties": false }, { "type":
"object", "required": [ "Vector" ], "properties": { "Vector": { "$ref":
```

```
"#/components/schemas/SuiMoveNormalizedType" } }, "additionalProperties": false }, { "type":
"object", "required": [ "TypeParameter" ], "properties": { "TypeParameter": { "type": "integer",
"format": "uint16", "minimum": 0 } }, "additionalProperties": false }, { "type": "object",
"required": [ "Reference" ], "properties": { "Reference": { "$ref":
"#/components/schemas/SuiMoveNormalizedType" } }, "additionalProperties": false }, { "type":
"object", "required": [ "MutableReference" ], "properties": { "MutableReference": { "$ref":
"#/components/schemas/SuiMoveNormalizedType" } }, "additionalProperties": false } ] }
```

SuiMoveStructTypeParameter

Object

```
bash "SuiMoveStructTypeParameter": { "type": "object", "required": [ "constraints", "isPhantom" ],
"properties": { "constraints": { "$ref": "#/components/schemas/SuiMoveAbilitySet" }, "isPhantom": {
"type": "boolean" } } }
```

SuiMoveVisibility

String enum ["Private" | "Public" | "Friend"]

```
bash "SuiMoveVisibility": { "type": "string", "enum": [ "Private", "Public", "Friend" ] }
```

SuiObjectDataFilter

One of

Object

Object

Object

Object

Query by type a specified Package.

Object

Query by type a specified Move module.

ObjectID

Object

Query by type

Object

Object

Object

Object

Object

```
bash "SuiObjectDataFilter": { "oneOf": [ { "type": "object", "required": [ "MatchAll" ],
"properties": { "MatchAll": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiObjectDataFilter" } } }, "additionalProperties": false }, { "type":
"object", "required": [ "MatchAny" ], "properties": { "MatchAny": { "type": "array", "items": {
"$ref": "#/components/schemas/SuiObjectDataFilter" } } }, "additionalProperties": false }, {
"type": "object", "required": [ "MatchNone" ], "properties": { "MatchNone": { "type": "array",
"items": { "$ref": "#/components/schemas/SuiObjectDataFilter" } } }, "additionalProperties": false
}, { "description": "Query by type a specified Package.", "type": "object", "required": [ "Package"
], "properties": { "Package": { "$ref": "#/components/schemas/ObjectID" } },
"additionalProperties": false }, { "description": "Query by type a specified Move module.", "type":
"object", "required": [ "MoveModule" ], "properties": { "MoveModule": { "type": "object",
"required": [ "module", "package" ], "properties": { "module": { "description": "the module name",
```

"type": "string" }, "package": { "description": "the Move package ID", "allOf": [{ "$ref":
"#/components/schemas/ObjectID" }] } } } }, "additionalProperties": false }, { "description":
"Query by type", "type": "object", "required": ["StructType"], "properties": { "StructType": {
"type": "string" } }, "additionalProperties": false }, { "type": "object", "required": [
"AddressOwner"], "properties": { "AddressOwner": { "$ref": "#/components/schemas/SuiAddress" } },
"additionalProperties": false }, { "type": "object", "required": ["ObjectOwner"], "properties": {
"ObjectOwner": { "$ref": "#/components/schemas/ObjectID" } }, "additionalProperties": false }, {
"type": "object", "required": ["ObjectId"], "properties": { "ObjectId": { "$ref":
"#/components/schemas/ObjectID" } }, "additionalProperties": false }, { "type": "object",
"required": ["ObjectIds"], "properties": { "ObjectIds": { "type": "array", "items": { "$ref":
"#/components/schemas/ObjectID" } } }, "additionalProperties": false }, { "type": "object",
"required": ["Version"], "properties": { "Version": { "$ref":
"#/components/schemas/BigInt_for_uint64" } }, "additionalProperties": false }] }

SuiObjectResponse

O bject

bash "SuiObjectResponse": { "type": "object", "properties": { "data": { "anyOf": [{ "$ref":
"#/components/schemas/ObjectData" }, { "type": "null" }] }, "error": { "anyOf": [{ "$ref":
"#/components/schemas/ObjectResponseError" }, { "type": "null" }] } } }

SuiProgrammableMoveCall

The transaction for calling a Move function, either an entry function or a public function (which cannot return references).

O bject

ObjectID

bash "SuiProgrammableMoveCall": { "description": "The transaction for calling a Move function,
either an entry function or a public function (which cannot return references).", "type": "object",
"required": ["function", "module", "package"], "properties": { "arguments": { "description": "The
arguments to the function.", "type": "array", "items": { "$ref": "#/components/schemas/SuiArgument"
} }, "function": { "description": "The function to be called.", "type": "string" }, "module": {
"description": "The specific module in the package containing the function.", "type": "string" },
"package": { "description": "The package containing the module and function.", "allOf": [{ "$ref":
"#/components/schemas/ObjectID" }] }, "type_arguments": { "description": "The type arguments to
the function.", "type": "array", "items": { "type": "string" } } } }

SuiSystemStateSummary

This is the JSON-RPC type for the SUI system state object. It flattens all fields to make them top-level fields such that it as minimum dependencies to the internal data structures of the SUI system state type.

O bject

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

bash "SuiSystemStateSummary": { "description": "This is the JSON-RPC type for the SUI system state object. It flattens all fields to make them top-level fields such that it as minimum dependencies to the internal data structures of the SUI system state type.", "type": "object", "required": ["activeValidators", "atRiskValidators", "epoch", "epochDurationMs", "epochStartTimestampMs", "inactivePoolsId", "inactivePoolsSize", "maxValidatorCount", "minValidatorJoiningStake", "pendingActiveValidatorsId", "pendingActiveValidatorsSize", "pendingRemovals", "protocolVersion", "referenceGasPrice", "safeMode", "safeModeComputationRewards", "safeModeNonRefundableStorageFee", "safeModeStorageRebates", "safeModeStorageRewards", "stakeSubsidyBalance", "stakeSubsidyCurrentDistributionAmount", "stakeSubsidyDecreaseRate", "stakeSubsidyDistributionCounter", "stakeSubsidyPeriodLength", "stakeSubsidyStartEpoch", "stakingPoolMappingsId", "stakingPoolMappingsSize", "storageFundNonRefundableBalance", "storageFundTotalObjectStorageRebates", "systemStateVersion", "totalStake", "validatorCandidatesId", "validatorCandidatesSize", "validatorLowStakeGracePeriod", "validatorLowStakeThreshold", "validatorReportRecords", "validatorVeryLowStakeThreshold"], "properties": { "activeValidators": { "description": "The list of active validators in the current epoch.", "type": "array", "items": { "$ref": "#/components/schemas/SuiValidatorSummary" } }, "atRiskValidators": { "description": "Map storing the number of epochs for which each validator has been below the low stake threshold.", "type": "array", "items": { "type": "array", "items": [{ "$ref": "#/components/schemas/SuiAddress" }, { "$ref": "#/components/schemas/BigInt_for_uint64" }], "maxItems": 2, "minItems": 2 } }, "epoch": { "description": "The current epoch ID, starting from 0.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "epochDurationMs": { "description": "The duration of an epoch, in milliseconds.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "epochStartTimestampMs": { "description": "Unix timestamp of the current epoch start", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "inactivePoolsId": { "description": "ID of the object that maps from a staking pool ID to the inactive validator that has that pool as its staking pool.", "allOf": [{ "$ref": "#/components/schemas/ObjectID" }] }, "inactivePoolsSize": { "description": "Number of inactive staking pools.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "maxValidatorCount": { "description": "Maximum number of active validators at any moment. We do not allow the number of validators in any epoch to go above this.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "minValidatorJoiningStake": { "description": "Lower-bound on the amount of stake required to become a validator.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "pendingActiveValidatorsId": { "description": "ID of the object that contains the list of new validators that will join at the end of the epoch.", "allOf": [{ "$ref": "#/components/schemas/ObjectID" }] }, "pendingActiveValidatorsSize": { "description": "Number of new validators that will join at the end of the epoch.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "pendingRemovals": { "description": "Removal requests from the validators. Each element is an index pointing to `active_validators`.", "type": "array", "items": { "$ref": "#/components/schemas/BigInt_for_uint64" } }, "protocolVersion": { "description": "The current protocol version, starting from 1.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "referenceGasPrice": { "description": "The reference gas price for the current epoch.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "safeMode": { "description": "Whether the system is running in a downgraded safe mode due to a non-recoverable bug. This is set whenever we failed to execute advance_epoch, and ended up executing advance_epoch_safe_mode. It can be reset once we are able to successfully execute advance_epoch.", "type": "boolean" }, "safeModeComputationRewards": { "description": "Amount of computation rewards accumulated (and not yet distributed) during safe mode.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "safeModeNonRefundableStorageFee": { "description": "Amount of non-refundable storage fee accumulated during safe mode.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "safeModeStorageRebates": { "description": "Amount of storage rebates accumulated (and not yet burned) during safe mode.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "safeModeStorageRewards": { "description": "Amount of storage rewards accumulated (and not yet distributed) during safe mode.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakeSubsidyBalance": { "description": "Balance of SUI set aside for stake subsidies that will be drawn down over time.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakeSubsidyCurrentDistributionAmount": { "description": "The amount of stake subsidy to be drawn down per epoch. This amount decays and decreases over time.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakeSubsidyDecreaseRate": { "description": "The rate at which the distribution amount decays at the end of each period. Expressed in basis points.", "type": "integer", "format": "uint16", "minimum": 0 }, "stakeSubsidyDistributionCounter": { "description": "This counter may be different from the current epoch number if in some epochs we decide to skip the subsidy.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakeSubsidyPeriodLength": { "description": "Number of distributions to occur before the distribution amount decays.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakeSubsidyStartEpoch": { "description": "The starting epoch in which stake subsidies start being paid out", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "stakingPoolMappingsId": { "description": "ID of the object that maps from staking pool's ID to the sui address of a validator.", "allOf": [{ "$ref": "#/components/schemas/ObjectID" }] }, "stakingPoolMappingsSize": { "description": "Number of staking pool mappings.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "storageFundNonRefundableBalance": { "description": "The non-refundable portion of the storage fund coming from storage reinvestment, non-refundable storage rebates and any leftover staking rewards.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "storageFundTotalObjectStorageRebates": { "description": "The storage rebates of all the objects on-chain stored in the storage fund.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "systemStateVersion": { "description": "The current version of the system state data structure type.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "totalStake": { "description": "Total amount of stake from all active validators at the beginning of the epoch.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "validatorCandidatesId": { "description": "ID of the object that stores preactive validators, mapping their addresses to their `Validator` structs.", "allOf": [{ "$ref": "#/components/schemas/ObjectID" }] }, "validatorCandidatesSize": { "description": "Number of preactive validators.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "validatorLowStakeGracePeriod": { "description": "A validator can have stake below `validator_low_stake_threshold` for this many epochs before being kicked out.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "validatorLowStakeThreshold": { "description": "Validators with stake amount below `validator_low_stake_threshold` are considered to have low stake and will be escorted out of the validator set after being below this threshold for more than `validator_low_stake_grace_period` number of epochs.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] }, "validatorReportRecords": { "description": "A map storing the records of validator reporting each other.", "type": "array", "items": { "type": "array", "items": [{ "$ref": "#/components/schemas/SuiAddress" }, { "type": "array", "items": { "$ref": "#/components/schemas/SuiAddress" } }], "maxItems": 2, "minItems": 2 } }, "validatorVeryLowStakeThreshold": { "description": "Validators with stake below `validator_very_low_stake_threshold` will be removed immediately at epoch change, no grace period.", "allOf": [{ "$ref": "#/components/schemas/BigInt_for_uint64" }] } } }

SuiTransaction

A single transaction in a programmable transaction block.

One of

Object

A call to either an entry or a public Move function

Object

(Vec, address) It sends n-objects to the specified address. These objects must have store (public transfer) and either the previous owner must be an address or the object must be newly created.

Object

(&mut Coin, Vec) -> Vec> It splits off some amounts into a new coins with those amounts

Object

(&mut Coin, Vec>) It merges n-coins into the first coin

Object

Publishes a Move package. It takes the package bytes and a list of the package's transitive dependencies to link against on-chain.

Object

Upgrades a Move package

Object

forall T: Vec -> vector Given n-values of the same type, it constructs a vector. For non objects or an empty vector, the type tag must be specified.

```
bash "SuiTransaction": { "description": "A single transaction in a programmable transaction
block.", "oneOf": [ { "description": "A call to either an entry or a public Move function", "type":
"object", "required": [ "MoveCall" ], "properties": { "MoveCall": { "$ref":
"#/components/schemas/SuiProgrammableMoveCall" } }, "additionalProperties": false }, {
"description": "`(Vec<forall T:key+store. T>, address)` It sends n-objects to the specified
address. These objects must have store (public transfer) and either the previous owner must be an
address or the object must be newly created.", "type": "object", "required": [ "TransferObjects" ],
"properties": { "TransferObjects": { "type": "array", "items": [ { "type": "array", "items": {
"$ref": "#/components/schemas/SuiArgument" } }, { "$ref": "#/components/schemas/SuiArgument" } ],
"maxItems": 2, "minItems": 2 } }, "additionalProperties": false }, { "description": "`(&mut
Coin<T>, Vec<u64>)` -> `Vec<Coin<T>>` It splits off some amounts into a new coins with those
amounts", "type": "object", "required": [ "SplitCoins" ], "properties": { "SplitCoins": { "type":
"array", "items": [ { "$ref": "#/components/schemas/SuiArgument" }, { "type": "array", "items": {
"$ref": "#/components/schemas/SuiArgument" } } ], "maxItems": 2, "minItems": 2 } },
"additionalProperties": false }, { "description": "`(&mut Coin<T>, Vec<Coin<T>>)` It merges n-coins
into the first coin", "type": "object", "required": [ "MergeCoins" ], "properties": { "MergeCoins":
{ "type": "array", "items": [ { "$ref": "#/components/schemas/SuiArgument" }, { "type": "array",
"items": { "$ref": "#/components/schemas/SuiArgument" } } ], "maxItems": 2, "minItems": 2 } },
"additionalProperties": false }, { "description": "Publishes a Move package. It takes the package
bytes and a list of the package's transitive dependencies to link against on-chain.", "type":
"object", "required": [ "Publish" ], "properties": { "Publish": { "type": "array", "items": {
"$ref": "#/components/schemas/ObjectID" } } }, "additionalProperties": false }, { "description":
"Upgrades a Move package", "type": "object", "required": [ "Upgrade" ], "properties": { "Upgrade":
{ "type": "array", "items": [ { "type": "array", "items": { "$ref": "#/components/schemas/ObjectID"
} }, { "$ref": "#/components/schemas/ObjectID" }, { "$ref": "#/components/schemas/SuiArgument" } ],
"maxItems": 3, "minItems": 3 } }, "additionalProperties": false }, { "description": "`forall T:
Vec<T> -> vector<T>` Given n-values of the same type, it constructs a vector. For non objects or an
empty vector, the type tag must be specified.", "type": "object", "required": [ "MakeMoveVec" ],
"properties": { "MakeMoveVec": { "type": "array", "items": [ { "type": [ "string", "null" ] }, {
"type": "array", "items": { "$ref": "#/components/schemas/SuiArgument" } } ], "maxItems": 2,
"minItems": 2 } }, "additionalProperties": false } ] }
```

SuiTransactionBlockBuilderMode

One of

String enum: ["Commit"]

Regular Sui Transactions that are committed on chain

String enum: ["DevInspect"]

Simulated transaction that allows calling any Move function with arbitrary values.

```
bash "SuiTransactionBlockBuilderMode": { "oneOf": [ { "description": "Regular Sui Transactions that
are committed on chain", "type": "string", "enum": [ "Commit" ] }, { "description": "Simulated
transaction that allows calling any Move function with arbitrary values.", "type": "string",
"enum": [ "DevInspect" ] } ] }
```

SuiValidatorSummary

This is the JSON-RPC type for the SUI validator. It flattens all inner structures to top-level fields so that they are decoupled from the internal definitions.

O bject

ObjectID

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

BigInt_for_uint64

ObjectID

BigInt_for_uint64

```
bash "SuiValidatorSummary": { "description": "This is the JSON-RPC type for the SUI validator. It
flattens all inner structures to top-level fields so that they are decoupled from the internal
definitions.", "type": "object", "required": [ "commissionRate", "description", "exchangeRatesId",
"exchangeRatesSize", "gasPrice", "imageUrl", "name", "netAddress", "networkPubkeyBytes",
"nextEpochCommissionRate", "nextEpochGasPrice", "nextEpochStake", "operationCapId", "p2pAddress",
"pendingPoolTokenWithdraw", "pendingStake", "pendingTotalSuiWithdraw", "poolTokenBalance",
"primaryAddress", "projectUrl", "proofOfPossessionBytes", "protocolPubkeyBytes", "rewardsPool",
"stakingPoolId", "stakingPoolSuiBalance", "suiAddress", "votingPower", "workerAddress",
"workerPubkeyBytes" ], "properties": { "commissionRate": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "description": { "type": "string" }, "exchangeRatesId":
{ "description": "ID of the exchange rate table object.", "allOf": [ { "$ref":
"#/components/schemas/ObjectID" } ] }, "exchangeRatesSize": { "description": "Number of exchange
rates in the table.", "allOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" } ] },
"gasPrice": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "imageUrl": { "type": "string" },
"name": { "type": "string" }, "netAddress": { "type": "string" }, "networkPubkeyBytes": { "$ref":
"#/components/schemas/Base64" }, "nextEpochCommissionRate": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "nextEpochGasPrice": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "nextEpochNetAddress": { "type": [ "string", "null" ]
}, "nextEpochNetworkPubkeyBytes": { "default": null, "anyOf": [ { "$ref":
"#/components/schemas/Base64" }, { "type": "null" } ] }, "nextEpochP2pAddress": { "type": [
"string", "null" ] }, "nextEpochPrimaryAddress": { "type": [ "string", "null" ] },
"nextEpochProofOfPossession": { "default": null, "anyOf": [ { "$ref": "#/components/schemas/Base64"
}, { "type": "null" } ] }, "nextEpochProtocolPubkeyBytes": { "default": null, "anyOf": [ { "$ref":
"#/components/schemas/Base64" }, { "type": "null" } ] }, "nextEpochStake": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "nextEpochWorkerAddress": { "type": [ "string", "null"
] }, "nextEpochWorkerPubkeyBytes": { "default": null, "anyOf": [ { "$ref":
"#/components/schemas/Base64" }, { "type": "null" } ] }, "operationCapId": { "$ref":
"#/components/schemas/ObjectID" }, "p2pAddress": { "type": "string" }, "pendingPoolTokenWithdraw":
{ "description": "Pending pool token withdrawn during the current epoch, emptied at epoch
boundaries.", "allOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" } ] }, "pendingStake":
{ "description": "Pending stake amount for this epoch.", "allOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" } ] }, "pendingTotalSuiWithdraw": { "description":
"Pending stake withdrawn during the current epoch, emptied at epoch boundaries.", "allOf": [ {
"$ref": "#/components/schemas/BigInt_for_uint64" } ] }, "poolTokenBalance": { "description": "Total
number of pool tokens issued by the pool.", "allOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" } ] }, "primaryAddress": { "type": "string" },
"projectUrl": { "type": "string" }, "proofOfPossessionBytes": { "$ref":
```

```
"#/components/schemas/Base64" }, "protocolPubkeyBytes": { "$ref": "#/components/schemas/Base64" },
"rewardsPool": { "description": "The epoch stake rewards will be added here at the end of each
epoch.", "allOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" } ] },
"stakingPoolActivationEpoch": { "description": "The epoch at which this pool became active.",
"default": null, "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, { "type": "null"
} ] }, "stakingPoolDeactivationEpoch": { "description": "The epoch at which this staking pool
ceased to be active. `None` = {pre-active, active},", "default": null, "anyOf": [ { "$ref":
"#/components/schemas/BigInt_for_uint64" }, { "type": "null" } ] }, "stakingPoolId": {
"description": "ID of the staking pool object.", "allOf": [ { "$ref":
"#/components/schemas/ObjectID" } ] }, "stakingPoolSuiBalance": { "description": "The total number
of SUI tokens in this pool.", "allOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" } ] },
"suiAddress": { "$ref": "#/components/schemas/SuiAddress" }, "votingPower": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "workerAddress": { "type": "string" },
"workerPubkeyBytes": { "$ref": "#/components/schemas/Base64" } } }
```

Supply

O bject

```
bash "Supply": { "type": "object", "required": [ "value" ], "properties": { "value": { "$ref":
"#/components/schemas/BigInt_for_uint64" } } }
```

TransactionBlock

O bject

```
bash "TransactionBlock": { "type": "object", "required": [ "data", "txSignatures" ], "properties":
{ "data": { "$ref": "#/components/schemas/TransactionBlockData" }, "txSignatures": { "type":
"array", "items": { "$ref": "#/components/schemas/GenericSignature" } } } }
```

TransactionBlockBytes

O bject

Base64

```
bash "TransactionBlockBytes": { "type": "object", "required": [ "gas", "inputObjects", "txBytes" ],
"properties": { "gas": { "description": "the gas objects to be used", "type": "array", "items": {
"$ref": "#/components/schemas/ObjectRef" } }, "inputObjects": { "description": "objects to be used
in this transaction", "type": "array", "items": { "$ref": "#/components/schemas/InputObjectKind" }
}, "txBytes": { "description": "BCS serialized transaction data bytes without its type tag, as
base-64 encoded string.", "allOf": [ { "$ref": "#/components/schemas/Base64" } ] } } }
```

TransactionBlockData

One of

Object

```
bash "TransactionBlockData": { "oneOf": [ { "type": "object", "required": [ "gasData",
"messageVersion", "sender", "transaction" ], "properties": { "gasData": { "$ref":
"#/components/schemas/GasData" }, "messageVersion": { "type": "string", "enum": [ "v1" ] },
"sender": { "$ref": "#/components/schemas/SuiAddress" }, "transaction": { "$ref":
"#/components/schemas/TransactionBlockKind" } } } ] }
```

TransactionBlockEffects

One of

Object

The response from processing a transaction or a certified transaction

BigInt_for_uint64

OwnedObjectRef

ExecutionStatus

TransactionDigest

```
bash "TransactionBlockEffects": { "oneOf": [ { "description": "The response from processing a
transaction or a certified transaction", "type": "object", "required": [ "executedEpoch",
"gasObject", "gasUsed", "messageVersion", "status", "transactionDigest" ], "properties": {
"created": { "description": "ObjectRef and owner of new objects created.", "type": "array",
"items": { "$ref": "#/components/schemas/OwnedObjectRef" } }, "deleted": { "description": "Object
Refs of objects now deleted (the old refs).", "type": "array", "items": { "$ref":
"#/components/schemas/ObjectRef" } }, "dependencies": { "description": "The set of transaction
digests this transaction depends on.", "type": "array", "items": { "$ref":
"#/components/schemas/TransactionDigest" } }, "eventsDigest": { "description": "The digest of the
events emitted during execution, can be None if the transaction does not emit any event.", "anyOf":
[ { "$ref": "#/components/schemas/TransactionEventsDigest" }, { "type": "null" } ] },
"executedEpoch": { "description": "The epoch when this transaction was executed.", "allOf": [ {
"$ref": "#/components/schemas/BigInt_for_uint64" } ] }, "gasObject": { "description": "The updated
gas object reference. Have a dedicated field for convenient access. It's also included in
mutated.", "allOf": [ { "$ref": "#/components/schemas/OwnedObjectRef" } ] }, "gasUsed": { "$ref":
"#/components/schemas/GasCostSummary" }, "messageVersion": { "type": "string", "enum": [ "v1" ] },
"modifiedAtVersions": { "description": "The version that every modified (mutated or deleted) object
had before it was modified by this transaction.", "type": "array", "items": { "$ref":
"#/components/schemas/TransactionBlockEffectsModifiedAtVersions" } }, "mutated": { "description":
"ObjectRef and owner of mutated objects, including gas object.", "type": "array", "items": {
"$ref": "#/components/schemas/OwnedObjectRef" } }, "sharedObjects": { "description": "The object
references of the shared objects used in this transaction. Empty if no shared objects were used.",
"type": "array", "items": { "$ref": "#/components/schemas/ObjectRef" } }, "status": {
"description": "The status of the execution", "allOf": [ { "$ref":
"#/components/schemas/ExecutionStatus" } ] }, "transactionDigest": { "description": "The
transaction digest", "allOf": [ { "$ref": "#/components/schemas/TransactionDigest" } ] },
"unwrapped": { "description": "ObjectRef and owner of objects that are unwrapped in this
transaction. Unwrapped objects are objects that were wrapped into other objects in the past, and
just got extracted out.", "type": "array", "items": { "$ref": "#/components/schemas/OwnedObjectRef"
} }, "unwrappedThenDeleted": { "description": "Object refs of objects previously wrapped in other
objects but now deleted.", "type": "array", "items": { "$ref": "#/components/schemas/ObjectRef" }
}, "wrapped": { "description": "Object refs of objects now wrapped in other objects.", "type":
"array", "items": { "$ref": "#/components/schemas/ObjectRef" } } } } ] }
```

TransactionBlockEffectsModifiedAtVersions

O bject

```
bash "TransactionBlockEffectsModifiedAtVersions": { "type": "object", "required": [ "objectId",
"sequenceNumber" ], "properties": { "objectId": { "$ref": "#/components/schemas/ObjectID" },
"sequenceNumber": { "$ref": "#/components/schemas/SequenceNumber" } } }
```

TransactionBlockKind

One of

Object

A system transaction that will update epoch information on-chain.

Object

A system transaction used for initializing the initial state of the chain.

Object

A system transaction marking the start of a series of transactions scheduled as part of a checkpoint

Object

A series of transactions where the results of one transaction can be used in future transactions

Object

A transaction which updates global authenticator state

Object

A transaction which updates global randomness state

Object

The transaction which occurs only at the end of the epoch

Object

Object

Object

```
bash "TransactionBlockKind": { "oneOf": [ { "description": "A system transaction that will update
epoch information on-chain.", "type": "object", "required": [ "computation_charge", "epoch",
"epoch_start_timestamp_ms", "kind", "storage_charge", "storage_rebate" ], "properties": {
"computation_charge": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "epoch_start_timestamp_ms": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum": [ "ChangeEpoch" ]
}, "storage_charge": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "storage_rebate": {
"$ref": "#/components/schemas/BigInt_for_uint64" } } }, { "description": "A system transaction used
for initializing the initial state of the chain.", "type": "object", "required": [ "kind",
"objects" ], "properties": { "kind": { "type": "string", "enum": [ "Genesis" ] }, "objects": {
"type": "array", "items": { "$ref": "#/components/schemas/ObjectID" } } } }, { "description": "A
system transaction marking the start of a series of transactions scheduled as part of a
checkpoint", "type": "object", "required": [ "commit_timestamp_ms", "epoch", "kind", "round" ],
"properties": { "commit_timestamp_ms": { "$ref": "#/components/schemas/BigInt_for_uint64" },
"epoch": { "$ref": "#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum":
[ "ConsensusCommitPrologue" ] }, "round": { "$ref": "#/components/schemas/BigInt_for_uint64" } } },
{ "description": "A series of transactions where the results of one transaction can be used in
future transactions", "type": "object", "required": [ "inputs", "kind", "transactions" ],
"properties": { "inputs": { "description": "Input objects or primitive values", "type": "array",
"items": { "$ref": "#/components/schemas/SuiCallArg" } }, "kind": { "type": "string", "enum": [
"ProgrammableTransaction" ] }, "transactions": { "description": "The transactions to be executed
sequentially. A failure in any transaction will result in the failure of the entire programmable
transaction block.", "type": "array", "items": { "$ref": "#/components/schemas/SuiTransaction" } }
} }, { "description": "A transaction which updates global authenticator state", "type": "object",
"required": [ "epoch", "kind", "new_active_jwks", "round" ], "properties": { "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum": [
"AuthenticatorStateUpdate" ] }, "new_active_jwks": { "type": "array", "items": { "$ref":
"#/components/schemas/SuiActiveJwk" } }, "round": { "$ref":
"#/components/schemas/BigInt_for_uint64" } } }, { "description": "A transaction which updates
global randomness state", "type": "object", "required": [ "epoch", "kind", "random_bytes",
"randomness_round" ], "properties": { "epoch": { "$ref": "#/components/schemas/BigInt_for_uint64"
}, "kind": { "type": "string", "enum": [ "RandomnessStateUpdate" ] }, "random_bytes": { "type":
"array", "items": { "type": "integer", "format": "uint8", "minimum": 0 } }, "randomness_round": {
"$ref": "#/components/schemas/BigInt_for_uint64" } } }, { "description": "The transaction which
occurs only at the end of the epoch", "type": "object", "required": [ "kind", "transactions" ],
"properties": { "kind": { "type": "string", "enum": [ "EndOfEpochTransaction" ] }, "transactions":
{ "type": "array", "items": { "$ref": "#/components/schemas/SuiEndOfEpochTransactionKind" } } } },
{ "type": "object", "required": [ "commit_timestamp_ms", "consensus_commit_digest", "epoch",
"kind", "round" ], "properties": { "commit_timestamp_ms": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "consensus_commit_digest": { "$ref":
"#/components/schemas/ConsensusCommitDigest" }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum": [
"ConsensusCommitPrologueV2" ] }, "round": { "$ref": "#/components/schemas/BigInt_for_uint64" } } },
{ "type": "object", "required": [ "commit_timestamp_ms", "consensus_commit_digest",
"consensus_determined_version_assignments", "epoch", "kind", "round" ], "properties": {
"commit_timestamp_ms": { "$ref": "#/components/schemas/BigInt_for_uint64" },
"consensus_commit_digest": { "$ref": "#/components/schemas/ConsensusCommitDigest" },
"consensus_determined_version_assignments": { "$ref":
"#/components/schemas/ConsensusDeterminedVersionAssignments" }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum": [
"ConsensusCommitPrologueV3" ] }, "round": { "$ref": "#/components/schemas/BigInt_for_uint64" },
"sub_dag_index": { "default": null, "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64"
}, { "type": "null" } ] } } }, { "type": "object", "required": [ "additional_state_digest",
"commit_timestamp_ms", "consensus_commit_digest", "consensus_determined_version_assignments",
"epoch", "kind", "round" ], "properties": { "additional_state_digest": { "$ref":
"#/components/schemas/AdditionalConsensusStateDigest" }, "commit_timestamp_ms": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "consensus_commit_digest": { "$ref":
"#/components/schemas/ConsensusCommitDigest" }, "consensus_determined_version_assignments": {
"$ref": "#/components/schemas/ConsensusDeterminedVersionAssignments" }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" }, "kind": { "type": "string", "enum": [
"ConsensusCommitPrologueV4" ] }, "round": { "$ref": "#/components/schemas/BigInt_for_uint64" },
"sub_dag_index": { "default": null, "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64"
}, { "type": "null" } ] } } } ] }
```

TransactionBlockResponse

Object

Base64

```
bash "TransactionBlockResponse": { "type": "object", "required": [ "digest" ], "properties": {
"balanceChanges": { "type": [ "array", "null" ], "items": { "$ref":
"#/components/schemas/BalanceChange" } }, "checkpoint": { "description": "The checkpoint number
when this transaction was included and hence finalized. This is only returned in the read api, not
in the transaction execution api.", "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64"
}, { "type": "null" } ] }, "confirmedLocalExecution": { "type": [ "boolean", "null" ] }, "digest":
{ "$ref": "#/components/schemas/TransactionDigest" }, "effects": { "anyOf": [ { "$ref":
"#/components/schemas/TransactionBlockEffects" }, { "type": "null" } ] }, "errors": { "type":
"array", "items": { "type": "string" } }, "events": { "type": [ "array", "null" ], "items": {
"$ref": "#/components/schemas/Event" } }, "objectChanges": { "type": [ "array", "null" ], "items":
{ "$ref": "#/components/schemas/ObjectChange" } }, "rawEffects": { "type": "array", "items": {
"type": "integer", "format": "uint8", "minimum": 0 } }, "rawTransaction": { "description": "BCS
encoded [SenderSignedData] that includes input object references returns empty array if
`show_raw_transaction` is false", "allOf": [ { "$ref": "#/components/schemas/Base64" } ] },
"timestampMs": { "anyOf": [ { "$ref": "#/components/schemas/BigInt_for_uint64" }, { "type": "null"
} ] }, "transaction": { "description": "Transaction input data", "anyOf": [ { "$ref":
"#/components/schemas/TransactionBlock" }, { "type": "null" } ] } } }
```

TransactionBlockResponseOptions

Object

```
bash "TransactionBlockResponseOptions": { "type": "object", "properties": { "showBalanceChanges": {
"description": "Whether to show balance_changes. Default to be False", "default": false, "type":
"boolean" }, "showEffects": { "description": "Whether to show transaction effects. Default to be
False", "default": false, "type": "boolean" }, "showEvents": { "description": "Whether to show
transaction events. Default to be False", "default": false, "type": "boolean" }, "showInput": {
"description": "Whether to show transaction input data. Default to be False", "default": false,
"type": "boolean" }, "showObjectChanges": { "description": "Whether to show object_changes. Default
to be False", "default": false, "type": "boolean" }, "showRawEffects": { "description": "Whether to
show raw transaction effects. Default to be False", "default": false, "type": "boolean" },
"showRawInput": { "description": "Whether to show bcs-encoded transaction input data", "default":
false, "type": "boolean" } } }
```

TransactionBlockResponseQuery

Object

```
bash "TransactionBlockResponseQuery": { "type": "object", "properties": { "filter": {
"description": "If None, no filter will be applied", "default": null, "anyOf": [ { "$ref":
"#/components/schemas/TransactionFilter" }, { "type": "null" } ] }, "options": { "description":
"config which fields to include in the response, by default only digest is included", "default":
null, "anyOf": [ { "$ref": "#/components/schemas/TransactionBlockResponseOptions" }, { "type":
"null" } ] } } }
```

TransactionDigest

A transaction will have a (unique) digest.

Digest

```
bash "TransactionDigest": { "description": "A transaction will have a (unique) digest.", "allOf": [
{ "$ref": "#/components/schemas/Digest" } ] }
```

TransactionEventsDigest

```
bash "TransactionEventsDigest": { "$ref": "#/components/schemas/Digest" }
```

TransactionFilter

One of

Object

CURRENTLY NOT SUPPORTED. Query by checkpoint.

Object

Query by move function.

Object

Query by input object.

Object

Query by changed object, including created, mutated and unwrapped objects.

Object

Query for transactions that touch this object.

Object

Query by sender address.

Object

Query by recipient address.

Object

Query by sender and recipient address.

Object

CURRENTLY NOT SUPPORTED. Query txs that have a given address as sender or recipient.

Object

Query by transaction kind

Object

Query transactions of any given kind in the input.

```
bash "TransactionFilter": { "oneOf": [ { "description": "CURRENTLY NOT SUPPORTED. Query by
checkpoint.", "type": "object", "required": [ "Checkpoint" ], "properties": { "Checkpoint": {
"$ref": "#/components/schemas/BigInt_for_uint64" } }, "additionalProperties": false }, {
"description": "Query by move function.", "type": "object", "required": [ "MoveFunction" ],
"properties": { "MoveFunction": { "type": "object", "required": [ "package" ], "properties": {
"function": { "type": [ "string", "null" ] }, "module": { "type": [ "string", "null" ] },
"package": { "$ref": "#/components/schemas/ObjectID" } } } }, "additionalProperties": false }, {
"description": "Query by input object.", "type": "object", "required": [ "InputObject" ],
"properties": { "InputObject": { "$ref": "#/components/schemas/ObjectID" } },
"additionalProperties": false }, { "description": "Query by changed object, including created,
mutated and unwrapped objects.", "type": "object", "required": [ "ChangedObject" ], "properties": {
"ChangedObject": { "$ref": "#/components/schemas/ObjectID" } }, "additionalProperties": false }, {
"description": "Query for transactions that touch this object.", "type": "object", "required": [
"AffectedObject" ], "properties": { "AffectedObject": { "$ref": "#/components/schemas/ObjectID" }
}, "additionalProperties": false }, { "description": "Query by sender address.", "type": "object",
"required": [ "FromAddress" ], "properties": { "FromAddress": { "$ref":
"#/components/schemas/SuiAddress" } }, "additionalProperties": false }, { "description": "Query by
recipient address.", "type": "object", "required": [ "ToAddress" ], "properties": { "ToAddress": {
"$ref": "#/components/schemas/SuiAddress" } }, "additionalProperties": false }, { "description":
"Query by sender and recipient address.", "type": "object", "required": [ "FromAndToAddress" ],
"properties": { "FromAndToAddress": { "type": "object", "required": [ "from", "to" ], "properties":
{ "from": { "$ref": "#/components/schemas/SuiAddress" }, "to": { "$ref":
"#/components/schemas/SuiAddress" } } } }, "additionalProperties": false }, { "description":
"CURRENTLY NOT SUPPORTED. Query txs that have a given address as sender or recipient.", "type":
```

```
"object", "required": [ "FromOrToAddress" ], "properties": { "FromOrToAddress": { "type": "object",
"required": [ "addr" ], "properties": { "addr": { "$ref": "#/components/schemas/SuiAddress" } } }
}, "additionalProperties": false }, { "description": "Query by transaction kind", "type": "object",
"required": [ "TransactionKind" ], "properties": { "TransactionKind": { "type": "string" } },
"additionalProperties": false }, { "description": "Query transactions of any given kind in the
input.", "type": "object", "required": [ "TransactionKindIn" ], "properties": {
"TransactionKindIn": { "type": "array", "items": { "type": "string" } } }, "additionalProperties":
false } ] }
```

TransferObjectParams

O bject

```
bash "TransferObjectParams": { "type": "object", "required": [ "objectId", "recipient" ],
"properties": { "objectId": { "$ref": "#/components/schemas/ObjectID" }, "recipient": { "$ref":
"#/components/schemas/SuiAddress" } } }
```

TypeOrigin

Identifies a struct and the module it was defined in

O bject

```
bash "TypeOrigin": { "description": "Identifies a struct and the module it was defined in", "type":
"object", "required": [ "datatype_name", "module_name", "package" ], "properties": {
"datatype_name": { "type": "string" }, "module_name": { "type": "string" }, "package": { "$ref":
"#/components/schemas/ObjectID" } } }
```

TypeTag

S tring

```
bash "TypeTag": { "type": "string" }
```

UpgradeInfo

Upgraded package info for the linkage table

O bject

ObjectID

SequenceNumber2

```
bash "UpgradeInfo": { "description": "Upgraded package info for the linkage table", "type":
"object", "required": [ "upgraded_id", "upgraded_version" ], "properties": { "upgraded_id": {
"description": "ID of the upgraded packages", "allOf": [ { "$ref": "#/components/schemas/ObjectID"
} ] }, "upgraded_version": { "description": "Version of the upgraded package", "allOf": [ { "$ref":
"#/components/schemas/SequenceNumber2" } ] } } }
```

ValidatorApy

O bject

```
bash "ValidatorApy": { "type": "object", "required": [ "address", "apy" ], "properties": {
"address": { "$ref": "#/components/schemas/SuiAddress" }, "apy": { "type": "number", "format":
"double" } } }
```

ValidatorApys

O bject

```
bash "ValidatorApys": { "type": "object", "required": [ "apys", "epoch" ], "properties": { "apys":
{ "type": "array", "items": { "$ref": "#/components/schemas/ValidatorApy" } }, "epoch": { "$ref":
"#/components/schemas/BigInt_for_uint64" } } }
```

ZkLoginAuthenticator

An zk login authenticator with all the necessary fields.

Object

```
bash "ZkLoginAuthenticator": { "description": "An zk login authenticator with all the necessary
fields.", "type": "object", "required": [ "inputs", "maxEpoch", "userSignature" ], "properties": {
"inputs": { "$ref": "#/components/schemas/ZkLoginInputs" }, "maxEpoch": { "type": "integer",
"format": "uint64", "minimum": 0 }, "userSignature": { "$ref": "#/components/schemas/Signature" } }
}
```

ZkLoginAuthenticatorAsBytes

```
bash "ZkLoginAuthenticatorAsBytes": { "$ref": "#/components/schemas/Base64" }
```

ZkLoginInputs

All inputs required for the zk login proof verification and other public inputs.

Object

```
bash "ZkLoginInputs": { "description": "All inputs required for the zk login proof verification and
other public inputs.", "type": "object", "required": [ "addressSeed", "headerBase64",
"issBase64Details", "proofPoints" ], "properties": { "addressSeed": { "$ref":
"#/components/schemas/Bn254FrElement" }, "headerBase64": { "type": "string" }, "issBase64Details":
{ "$ref": "#/components/schemas/Claim" }, "proofPoints": { "$ref":
"#/components/schemas/ZkLoginProof" } } }
```

ZkLoginIntentScope

String enum ["TransactionData" | "PersonalMessage"]

```
bash "ZkLoginIntentScope": { "type": "string", "enum": [ "TransactionData", "PersonalMessage" ] }
```

ZkLoginProof

The struct for zk login proof.

Object

```
bash "ZkLoginProof": { "description": "The struct for zk login proof.", "type": "object",
"required": [ "a", "b", "c" ], "properties": { "a": { "type": "array", "items": { "$ref":
"#/components/schemas/Bn254FqElement" } }, "b": { "type": "array", "items": { "type": "array",
"items": { "$ref": "#/components/schemas/Bn254FqElement" } } }, "c": { "type": "array", "items": {
"$ref": "#/components/schemas/Bn254FqElement" } } } }
```

ZkLoginPublicIdentifier

A wrapper struct to retrofit in [enum PublicKey] for zkLogin. Useful to construct [struct MultiSigPublicKey].

Base64

```
bash "ZkLoginPublicIdentifier": { "description": "A wrapper struct to retrofit in [enum PublicKey]
for zkLogin. Useful to construct [struct MultiSigPublicKey].", "allOf": [ { "$ref":
"#/components/schemas/Base64" } ] }
```

ZkLoginVerifyResult

Object

```
bash "ZkLoginVerifyResult": { "type": "object", "required": [ "errors", "success" ], "properties":
{ "errors": { "description": "The errors field captures any verification error", "type": "array",
"items": { "type": "string" } }, "success": { "description": "The boolean result of the
verification. If true, errors should be empty.", "type": "boolean" } } }
```