

Client

Queries

In this section we describe the queries required on grpc server.

```
// Query defines the gRPC querier service. service
```

```
Query
```

```
{ // Parameters queries the parameters of the module. rpc
```

```
Params ( QueryParamsRequest )
```

```
returns
```

```
( QueryParamsResponse )
```

```
{ option
```

```
( google . api . http ) . get =
```

```
"/neutron-org/neutron/contractmanager/params" ; }
```

```
// Queries a Failures by address. rpc
```

```
Failure ( QueryGetFailuresByAddressRequest )
```

```
returns
```

```
( QueryGetFailuresByAddressResponse )
```

```
{ option
```

```
( google . api . http ) . get =
```

```
"/neutron-org/neutron/contractmanager/failure/{address}" ; }
```

```
// Queries a list of failed addresses. rpc
```

```
AllFailures ( QueryAllFailureRequest )
```

```
returns
```

```
( QueryAllFailureResponse )
```

```
{ option
```

```
( google . api . http ) . get =
```

```
"/neutron-org/neutron/contractmanager/failure" ; } }
```

failures[address]

Returns list of all failures.

neutrond query contractmanager failures Example Returns info about all failures:neutrond query contractmanager failures
Output:

failures : -

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2ttr8m8f43vamudsqrdivs8qqvfwpj error :

'codespace: wasm, code: 5' id :

"1" sudo_payload : <serialized msg of MessageSudoCallback type

-

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2ttr8m8f43vamudsqrdivs8qqvfwpj error :

'codespace: wasm, code: 5' id :

"2" sudo_payload : <serialized msg of MessageSudoCallback type

-

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj error :

'codespace: contractmanager, code: 1103' id :

"3" sudo_payload : <serialized msg of MessageSudoCallback type

Returns list of all failures for specific contract address.

neutrnd query contractmanager failures [address] Example Returns failures for specific contract address:neutrnd query contractmanager failures neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj Output:

failures : -

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj error :

'codespace: wasm, code: 5' id :

"1" sudo_payload : <serialized msg of MessageSudoCallback type

-

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj error :

'codespace: wasm, code: 5' id :

"2" sudo_payload : <serialized msg of MessageSudoCallback type

-

address : neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj error :

'codespace: contractmanager, code: 1103' id :

"3" sudo_payload : <serialized msg of MessageSudoCallback type

Failure Details

Returns an exact error why contract failed to process certain ibc acknowledgement.

Note It is a CLI-like query, and you can not perform it onchain, e.g. you can not make a query from a contract. The reason is - cosmos-sdk do not store raw errors in a storage due to non-deterministic nature of errors. You can not consider the data which are not under your control as deterministic. But to make developers' life easier we incorporated the following mechanism. In case of error happened during the sudo call we emit an event with full error text under the key(contract_address,failure_id) . Events are not the part of the consensus, and it's safe to emit any data you want. The CLI command looks for [transaction](#) by a set of events.

Note If the node you are making query to either does not index transaction or already cleared the block with the wanted transaction, you get the error -detailed failure error message not found in node events . In this case you need to query a node, which:

1) Indexes transactions.

2) Keeps block with wanted height. neutrnd q contractmanager failure-details [address]

[failure_id] Example Query the detailed error related to a contract's sudo call failure:neutrnd q contractmanager failure-details neutron1nxshmmwrvxa2cp80nwvf03t8u5kvl2tr8m8f43vamudsqrds8qqvfwpj 1 Output:

dispatch :

submessages :

Generic error :

Integrations test mock submsg error : execute wasm contract failed[Previous Overview Next State](#)