

#

Legacy Amino JSON REST

The Irishub versions v1.0.0 (depends on Cosmos-SDK v0.41) and earlier provided REST endpoints to query the state and broadcast transactions. These endpoints are kept in Irishub v1.0, but they are marked as deprecated, and will be removed after a few versions therefore call these endpoints legacy REST endpoints.

Some important information concerning all legacy REST endpoints:

- Most of these endpoints are backwards-compatible. All breaking changes are described in the next section.
- In particular, these endpoints still output Amino JSON. Cosmos-SDK v0.41 introduced Protobuf as the default encoding library throughout the codebase, but legacy REST endpoints are one of the few places where the encoding is hardcoded to Amino.

#

API Port, Activation and Configuration

All routes are configured under the following fields in `~/iris/config/app.toml` :

- `api.enable = true|false`
- field defines if the REST server should be enabled. Defaults to `true`
- .
- `api.address = {string}`
- field defines the address (really, the port, since the host should be kept at `0.0.0.0`
-) the server should bind to. Defaults to `tcp://0.0.0.0:1317`
- .
- some additional API configuration options are defined in `~/iris/config/app.toml`
- , along with comments, please refer to that file directly.

#

Legacy REST API Routes

The REST routes present in Irishub v0.16 and earlier are marked as deprecated via [a HTTP deprecation header](#)[open in new window](#) . They are still maintained to keep backwards compatibility, but will be removed after a few versions.

For application developers, Legacy REST API routes need to be wired up to the REST server, this is done by calling the `RegisterRESTRoutes` function on the `ModuleManager`.

#

Legacy REST Endpoint

#

Breaking Changes in Legacy REST Endpoints (Contrast with Cosmos-SDK v0.39 and earlier versions)

Legacy REST Endpoint Description Breaking Change

`POST /txs Broadcast tx` Endpoint will error when trying to broadcast transactions that don't support Amino serialization (e.g. IBC txs)

`1 . POST /txs/encode ,POST /txs/decode` Encode/decode Amino txs from JSON to binary

Endpoint will error when trying to encode/decode transactions that don't support Amino serialization (e.g. IBC txs)

`1 . GET /txs/{hash}` Query tx by hash

Endpoint will error when trying to output transactions that don't support Amino serialization (e.g. IBC txs)

`1 . GET /txs` Query tx by events

Endpoint will error when trying to output transactions that don't support Amino serialization (e.g. IBC txs)

`1 . GET /gov/proposals/{id}/votes ,GET /gov/proposals/{id}/votes/{voter}` Gov endpoints for querying votes

All gov endpoints which return votes return `int32` in the `option` field instead of `string:1=VOTE_OPTION_YES, 2=VOTE_OPTION_ABSTAIN, 3=VOTE_OPTION_NO, 4=VOTE_OPTION_NO_WITH_VETO` .

`GET /staking/*` Staking query endpoints

All staking endpoints which return validators have two breaking changes. First, the `validator's consensus_pubkey` field returns an Amino-encoded struct representing an `Any` instead of a bech32-encoded string representing the pubkey. The `value` field of the `Any` is the pubkey's raw key as base64-encoded bytes. Second, the `validator's status` field now returns an `int32` instead of `string:1=BOND_STATUS_UNBONDED ,2=BOND_STATUS_UNBONDING ,3=BOND_STATUS_BONDED` .

`GET /staking/validators` Get all validators

`BondStatus` is now a protobuf enum instead of an `int32`, and JSON serialized using its protobuf name, so expect query parameters like `?status=BOND_STATUS_{BONDED,UNBONDED,UNBONDING}` as opposed to `?status={bonded,unbonded,unbonding}` .

`1` : Transactions that don't support Amino serialization are the ones that contain one or more `Msg s` that are not registered with the Amino codec. Currently in the IRISHub, only `IBCMsg s` fall into this case.

#

Migrating to New REST Endpoints (from Cosmos-SDK v0.39)

IRIShub API Endpoints

Legacy REST Endpoint Description New gRPC-gateway REST Endpoint GET /bank/balances/{address} Get the balance of an address GET /cosmos/bank/v1beta1/balances/{address} POST /bank/accounts/{address}/transfers Send coins from one account to another N/A, use Protobuf directly GET /bank/total Get the total supply of all coins GET /cosmos/bank/v1beta1/supply GET /bank/total/{denom} Get the total supply of one coin GET /cosmos/bank/v1beta1/supply/{denom} GET /auth/accounts/{address} Get the account information on blockchain GET /cosmos/auth/v1beta1/accounts/{address} GET /staking/delegators/{delegatorAddr}/delegations Get all delegations from a delegator GET /cosmos/staking/v1beta1/delegations/{delegator_addr} POST /staking/delegators/{delegatorAddr}/delegations Submit delegation N/A, use Protobuf directly GET /staking/delegators/{delegatorAddr}/delegations/{validatorAddr} Query a delegation between a delegator and a validator GET /cosmos/staking/v1beta1/validators/{validator_addr}/delegations/{delegator_addr} GET /staking/delegators/{delegatorAddr}/unbonding_delegations Get all unbonding delegations from a delegator GET /cosmos/staking/v1beta1/delegators/{delegator_addr}/unbonding_delegations POST /staking/delegators/{delegatorAddr}/unbonding_delegations Submit an unbonding delegation N/A, use Protobuf directly GET /staking/delegators/{delegatorAddr}/unbonding_delegations/{validatorAddr} Query all unbonding delegations between a delegator and a validator GET /cosmos/staking/v1beta1/validators/{validator_addr}/delegations/{delegator_addr}/unbonding_delegation GET /staking/redelegations Query redelegations GET /cosmos/staking/v1beta1/delegators/{delegator_addr}/redelegations POST /staking/delegators/{delegatorAddr}/redelegations Submit a redelegations N/A, use Protobuf directly GET /staking/delegators/{delegatorAddr}/validators Query all validators that a delegator is bonded to GET /cosmos/staking/v1beta1/delegators/{delegator_addr}/validators GET /staking/delegators/{delegatorAddr}/validators/{validatorAddr} Query a validator that a delegator is bonded to GET /cosmos/staking/v1beta1/delegators/{delegator_addr}/validators/{validator_addr} GET /staking/validators Get all validators GET /cosmos/staking/v1beta1/validators GET /staking/validators/{validatorAddr} Get a single validator info GET /cosmos/staking/v1beta1/validators/{validator_addr} GET /staking/validators/{validatorAddr}/delegations Get all delegations to a validator GET /cosmos/staking/v1beta1/validators/{validator_addr}/delegations GET /staking/validators/{validatorAddr}/unbonding_delegations Get all unbonding delegations from a validator GET /cosmos/staking/v1beta1/validators/{validator_addr}/unbonding_delegations GET /staking/pool Get the current state of the staking pool GET /cosmos/staking/v1beta1/pool GET /staking/parameters Get the current staking parameter values GET /cosmos/staking/v1beta1/params GET /slashing/signing_infos Get all signing infos GET /cosmos/slashing/v1beta1/signing_infos POST /slashing/validators/{validatorAddr}/unjail Unjail a jailed validator N/A, use Protobuf directly GET /slashing/parameters Get slashing parameters GET /cosmos/slashing/v1beta1/params POST /gov/proposals Submit a proposal N/A, use Protobuf directly GET /gov/proposals Get all proposals GET /cosmos/gov/v1beta1/proposals POST /gov/proposals/param_change Generate a parameter change proposal transaction N/A, use Protobuf directly GET /gov/proposals/{proposal-id} Get proposal by id GET /cosmos/gov/v1beta1/proposals/{proposal_id} GET /gov/proposals/{proposal-id}/proposer Get proposer of a proposal GET /cosmos/gov/v1beta1/proposals/{proposal_id} (Get proposer from Proposal struct) GET /gov/proposals/{proposal-id}/deposits Get deposits of a proposal GET /cosmos/gov/v1beta1/proposals/{proposal_id}/deposits POST /gov/proposals/{proposal-id}/deposits Deposit tokens to a proposal N/A, use Protobuf directly GET /gov/proposals/{proposal-id}/deposits/{depositor} Get depositor of a deposit GET /cosmos/gov/v1beta1/proposals/{proposal_id}/deposits/{depositor} GET /gov/proposals/{proposal-id}/votes Get votes of a proposal GET /cosmos/gov/v1beta1/proposals/{proposal_id}/votes POST /gov/proposals/{proposal-id}/votes Vote a proposal N/A, use Protobuf directly GET /gov/proposals/{proposal-id}/votes/{voter} Get voted information by voterAddr GET /cosmos/gov/v1beta1/proposals/{proposal_id}/votes/{voter} GET /gov/proposals/{proposal-id}/tally Get tally of a proposal GET /cosmos/gov/v1beta1/proposals/{proposal_id}/tally GET /gov/parameters/deposit Get governance deposit parameters GET /cosmos/gov/v1beta1/params/{params_type} GET /gov/parameters/tallying Query governance tally parameters GET /cosmos/gov/v1beta1/params/{params_type} GET /gov/parameters/voting Get governance voting parameters GET /cosmos/gov/v1beta1/params/{params_type} GET /distribution/delegators/{delegatorAddr}/rewards Get the total rewards balance from all delegations GET /cosmos/distribution/v1beta1/delegators/{delegator_addr}/rewards POST /distribution/delegators/{delegatorAddr}/rewards Withdraw all delegator rewards N/A, use Protobuf directly GET /distribution/delegators/{delegatorAddr}/rewards/{validatorAddr} Query a delegation reward GET /cosmos/distribution/v1beta1/delegators/{delegator_addr}/rewards/{validator_addr} POST /distribution/delegators/{delegatorAddr}/rewards/{validatorAddr} Withdraw a delegation reward N/A, use Protobuf directly GET /distribution/delegators/{delegatorAddr}/withdraw_address Get the rewards withdrawal address GET /cosmos/distribution/v1beta1/delegators/{delegator_addr}/withdraw_address POST /distribution/delegators/{delegatorAddr}/withdraw_address Replace the rewards withdrawal address N/A, use Protobuf directly GET /distribution/validators/{validatorAddr} Validator distribution information N/A, use Protobuf directly GET /distribution/validators/{validatorAddr}/outstanding_rewards Outstanding rewards of a single validator GET /cosmos/distribution/v1beta1/validators/{validator_addr}/outstanding_rewards GET /distribution/validators/{validatorAddr}/rewards Commission and self-delegation rewards of a single a validator GET /cosmos/distribution/v1beta1/validators/{validator_addr}/commission GET /cosmos/distribution/v1beta1/validators/{validator_addr}/outstanding_rewards POST /distribution/validators/{validatorAddr}/rewards Withdraw the validator's rewards N/A, use Protobuf directly GET

/distribution/community_pool Get the amount held in the community pool GET /cosmos/distribution/v1beta1/community_pool
GET /distribution/parameters Get the current distribution parameter values GET /cosmos/distribution/v1beta1/params
Tendermint API Endpoints

Legacy REST Endpoint Description New gRPC-gateway REST Endpoint GET /node_info Get the properties of the
connected node GET /cosmos/base/tendermint/v1beta1/node_info GET /syncing Get syncing state of node GET
/cosmos/base/tendermint/v1beta1/syncing GET /blocks/latest Get the latest block GET
/cosmos/base/tendermint/v1beta1/blocks/latest GET /blocks/{height} Get a block at a certain height GET
/cosmos/base/tendermint/v1beta1/blocks/{height} GET /validatorsets/latest Get the latest validator set GET
/cosmos/base/tendermint/v1beta1/validatorsets/latest GET /validatorsets/{height} Get a validator set a certain height GET
/cosmos/base/tendermint/v1beta1/validatorsets/{height} GET /txs/{hash} Query tx by hash GET
/cosmos/tx/v1beta1/txs/{hash} GET /txs Query tx by events GET /cosmos/tx/v1beta1/txs POST /txs Broadcast tx POST
/cosmos/tx/v1beta1/txs POST /txs/encode Encodes an Amino JSON tx to an Amino binary tx N/A, use Protobuf directly
POST /txs/decode Decodes an Amino binary tx into an Amino JSON tx N/A, use Protobuf directly

#

Breaking changes in High Priority Endpoints for Queries

Here are the High Priority Endpoints for Queries

- Staking* Validators
- - Delegators
- Bank* Balances
- Gov
- Auth
- Distributions

#

Bank

Endpoint Name: QueryBalance

- Endpoint Path:
- /bank/balances/{address}
- What Changed:
- - No Changes observed.
- - See[coin cross-chain transfer source tracing](#)[open in new window](#)
- - for details on how on non-native IBC coins will written into the denom value. This will include a hash of source trace for each coin. The core decision if the hash should replace the denom or be prepended to the denom.

#

Validators

Endpoint Name: QueryValidators

- Endpoint Path:
- /staking/validators
- What Changed:
- - The fieldsunbonding_height
- - andjailed
- - are no longer supported
- - The fields in description are now omit if empty. Rather than returning fields with empty strings. We now don't return the field if the validator has chosen not to configure it. For instance at launch, no validator will have a security contact filled out and the field will only appear once they do.
- Sample JSON:
- {
- "commission": {
- "commission_rates": {

- "max_change_rate": "0.000000000000000000",
- "max_rate": "0.000000000000000000",
- "rate": "0.000000000000000000"
- },
- "update_time": "1970-01-01T00:00:00Z"
- },
- "consensus_pubkey":
"cosmosvalconspub1zcjduepqwuxd2yevzmsrmrjx2su8kdlk44eqfdzeqx27zejuen6m0nkcpzps0qavpw",
- "delegator_shares": "0.000000000000000000",
- "description": {
- "details": "security",
- "identity": "identity",
- "moniker": "moniker",
- "security_contact": "details",
- "website": "website"
- },
- "min_self_delegation": "1",
- "operator_address": "cosmosvaloper1pcpl7xhxq0wm72e9ljs2srx5h3vqwytnq44sr",
- "status": 1,
- "tokens": "0",
- "unbonding_time": "1970-01-01T00:00:00Z"
- }

#

Delegators

Endpoint Name: QueryDelegatorDelegations

- Endpoint Path:
- /staking/delegators/delegations
- What Changed:
- - balance
- - now is no longer a number. It is a field with two values:amount
- - andDenom
- - delegator_address
- - is no longer a string. It's a field calleddelegation
- - with three values:delegator_address", "shares", "validator_address
- - The old fieldvalidator_address
- - is no longer used. Two new fieldsvalidator_dst_address
- - andvalidator_src_address
- - replace this in the newredelegation
- - field.
- Sample JSON:
- {
- "balance": {
- "amount": "5",
- "denom": "stake"
- },
- "delegation": {
- "delegator_address": "cosmos1n2k9ygw2ws9sg86mrx84pdcre5geqd5ugt44h0",
- "shares": "5.000000000000000000",
- "validator_address": "cosmosvaloper155998a4hv5kqvuxr9jryjxrtlnlydvqu8c0cy03"
- }
- }

Endpoint Name: QueryRedelegations

- Endpoint Path:
- /staking/redelegations
- What Changed:
- The following old fields are now sub fields of a new field called redelegation_entry
- :
- - completion_time
- - initial_balance
- - shares_dst
- The old field creation_height
- is no longer supported.
- The following are new fields:
- - redelegation
- - which holds the sub-fields.* delegator_address
- - - (new)
- - - entries
- - - (new)
- - - validator_dst_address
- - - validator_src_address
- Sample JSON:
- {
- "entries": [
- {
- "balance": "5",
- "redelegation_entry": {
- "completion_time": "1969-12-31T16:00:00-08:00",
- "initial_balance": "5",
- "shares_dst": "5.000000000000000000"
- }
- },
- {
- "balance": "5",
- "redelegation_entry": {
- "completion_time": "1969-12-31T16:00:00-08:00",
- "initial_balance": "5",
- "shares_dst": "5.000000000000000000"
- }
- }
-],
- "redelegation": {
- "delegator_address": "cosmos104yggz5x4ype50c59vu84ze2w36pc3swm2u698",
- "entries": null,
- "validator_dst_address": "cosmosvaloper1td8yl7g5662m0mpptaxjmcn9jtzvl0wgulvv23",
- "validator_src_address": "cosmosvaloper1gqv70e79a8q0yz5s5qhsjhd12c79496faer0vz"
- }
- }

Endpoint Name: QueryUnbondingDelegation

- Endpoint Path:
- /staking/unbondingDelegation
- What Changed:
- - The old field creation_height
-

- is no longer supported

#

Distributions

Endpoint Name: getQueriedValidatorOutstandingRewards

- Endpoint Path:
- /distribution/validators/{validatorAddr}/outstanding_rewards
- What Changed:
- - The new fieldrewards
- - is the new root level field for the output
- Sample JSON:
- {
- "rewards": [
- {
- "denom": "mytoken",
- "amount": "3.000000000000000000"
- },
- {
- "denom": "myothertoken",
- "amount": "0.000000300000000000"
- }
-]
- }

Endpoint Name: getQueriedValidatorCommission

- Endpoint Path:
- /distribution/validators/{validatorAddr}
- What Changed:
- - The new fieldcommission
- - is the new root level field for the output
- Sample JSON:
- {
- "commission": [
- {
- "denom": "token1",
- "amount": "4.000000000000000000"
- },
- {
- "denom": "token2",
- "amount": "2.000000000000000000"
- }
-]
- }

Endpoint Name: getQueriedValidatorSlashes

- Endpoint Path:
- /distribution/validators/{validatorAddr}
- What Changed:
- No change
- Endpoint Name:
- getQueriedDelegationRewards
- Endpoint Path:
- /distribution/delegators/{delegatorAddr}/rewards
- What Changed:
- No change

#

Generating and Signing Transactions (Fully Backward Compatible)

The same code as integrating with irishub-v0.16.3 mainnet. The transaction structure is as follows:

```
{ "type": "cosmos-sdk/StdTx", "value": { "msg": [ { "type": "cosmos-sdk/MsgSend", "value": { "from_address": "iaa1rkgdpj6fyuu7pnhmc3v7gw9uls4mnajvzdwkt", "to_address": "iaa1q6t5439f0rkvkzl38m0f43e0kpv3mx7x2shlq8", "amount": [ { "denom": "uiris", "amount": "1000000" } ] }, "fee": { "amount": [ { "denom": "uiris", "amount": "30000" } ], "gas": "200000" }, "signatures": null, "memo": "Sent via irishub client" } } ] } Where the IRIShub address prefix usesiaa instead, which affects the fields:
```

- value.msg.value.from_address
- value.msg.value.to_address

Denom usesuiris instead (1iris = 106 uiris), which affects fields:

- value.msg.value.amount.denom
- value.fee.amount.denom

#

Broadcasting a transaction (Fully Backward Compatible)

The same code as integrating with irishub mainnet, callPOST /txs to send a transaction, as the example below:

```
curl -X POST"http://localhost:1317/txs" -H "accept: application/json" -H "Content-Type: application/json" -d '{"tx":{"msg":{"type":"cosmos-sdk/MsgSend","value":{"from_address":"iaa1rkgdpj6fyuu7pnhmc3v7gw9uls4mnajvzdwkt","to_address":"iaa1q6t5439f0rkvkzl38m0f43e0kpv3mx7x2shlq8","amount":[{"denom":"uiris","amount":"1000000"}]},"fee":{"amount":[{"denom":"uiris","amount":"30000"}],"gas":"200000"},"signatures":[{"pub_key":{"type":"tendermint/PublicKeySecp256k1","value":"AxGagdsRTKni/h1+vCFzTpNltwoiU7SwIR2dg6JI5a/"},"signature":"Pu8yiRVO8oB2YDDHyB047dXNArbVImasmKBrm8Kr+6B08y8QQ7YG1eVgHi5OIYYclccCf3Ju/BQ78qsMWMniNQ=="},"memo":"Sent via irishub client"},"mode":"block"}}
```

#

Breaking Changes in Querying Transactions

#

Tx

- Endpoint Name:
- QueryTx
- Endpoint Path:
- GET /txs
- &&GET /txs/{hash}
- What Changed:
- - Tags are no longer used; use the events field instead
- - The result field is no longer used, and the field in the original result is moved to the first level
- - The coin_flow field is no longer used
- Sample JSON:
- {
- "height"
- :
- "5"
- ,
- "txhash"
- :
- "E663768B616B1ACD2912E47C36FEB7DB0E0974D6DB3823D4C656E0EAB8C679D"
- ,
- "data"
- :
- "0A060A0473656E64"
- ,
- "raw_log"
- :
- "[{"events":[{"type":"message","attributes":{"key":"action","value":"send"}},

```
{\"key\": \"sender\", \"value\": \"iaa18awn3k70u05tlcul8w2qnl64g002uj4kjin93rn\"}, {\"key\": \"module\", \"value\": \"bank\"}},  
{\"type\": \"transfer\", \"attributes\": [{\"key\": \"recipient\", \"value\": \"iaa1w976a5jrjsj06dqmrh2x9qxzel74qtcmapklxc\"},  
{\"key\": \"sender\", \"value\": \"iaa18awn3k70u05tlcul8w2qnl64g002uj4kjin93rn\"},  
{\"key\": \"amount\", \"value\": \"1000000uiris\"}]]}]\"
```

```
• ,  
• "logs"  
• :  
• [  
• {  
• "events"  
• :  
• [  
• {  
• "type"  
• :  
• "message"  
• ,  
• "attributes"  
• :  
• [  
• {  
• "key"  
• :  
• "action"  
• ,  
• "value"  
• :  
• "send"  
• }  
• ,  
• {  
• "key"  
• :  
• "sender"  
• ,  
• "value"  
• :  
• "iaa18awn3k70u05tlcul8w2qnl64g002uj4kjin93rn"  
• }  
• ,  
• {  
• "key"  
• :  
• "module"  
• ,  
• "value"  
• :  
• "bank"  
• }  
• ]  
• }  
• ,  
• {  
• "type"  
• :  
• "transfer"  
• ,  
• "attributes"  
• :  
• [  
• {  
• "key"  
• :  
• "recipient"  
• ,  
• "value"  
• :  
• "iaa1w976a5jrjsj06dqmrh2x9qxzel74qtcmapklxc"
```



```
• }
• ,
• {
• "key"
• :
• "sender"
• ,
• "value"
• :
• "iaa18awn3k70u05tlcul8w2qnl64g002uj4kjin93rn"
• }
• ,
• {
• "key"
• :
• "amount"
• ,
• "value"
• :
• "1000000uiris"
• }
• ]
• }
• ]
• }
• ]
• ,
• "gas_wanted"
• :
• "200000"
• ,
• "gas_used"
• :
• "69256"
• ,
• "tx"
• :
• {
• "type"
• :
• "cosmos-sdk/StdTx"
• ,
• "value"
• :
• {
• "msg"
• :
• [
• {
• "type"
• :
• "cosmos-sdk/MsgSend"
• ,
• "value"
• :
• {
• "from_address"
• :
• "iaa18awn3k70u05tlcul8w2qnl64g002uj4kjin93rn"
• ,
• "to_address"
• :
• "iaa1w976a5jrhsj06dqmrh2x9qxzel74qtcmapk1xc"
• ,
• "amount"
• :
• [
• {
```

```
• "denom"
• :
• "uiris"
• ,
• "amount"
• :
• "1000000"
• }
• ]
• }
• }
• ]
• ,
• "fee"
• :
• {
• "amount"
• :
• [
• {
• "denom"
• :
• "uiris"
• ,
• "amount"
• :
• "30000"
• }
• ]
• ,
• "gas"
• :
• "200000"
• }
• ,
• "signatures"
• :
• [
• ]
• ,
• "memo"
• :
• ""
• ,
• "timeout_height"
• :
• "0"
• }
• }
• ,
• "timestamp"
• :
• "2021-01-18T07:29:21Z"
• }
```