

## Get Pool by ID

GET /liquidity\_pools/<liquidity-pool-id>

Every pool has a unique pool id. Horizon will provide an endpoint to query the state of a liquidity pool by its id. The JSON representation of a liquidity pool is defined below:

```
{
  "id": "abcdef",
  "paging_token": "abcdef",
  "fee_bp": 30,
  "type": "constant_product",
  "total_trustlines": "300",
  "total_shares": "5000",
  "reserves": [
    {
      "amount": "1000.0000005",
      "asset": "EURT:GAP5LET0V6YIE62YAM56STDANPRD07ZFDBGSNHJQIYGGKSMOZAH00S2S"
    },
    {
      "amount": "2000.0000000",
      "asset": "PHP:GAP5LET0V6YIE62YAM56STDANPRD07ZFDBGSNHJQIYGGKSMOZAH00S2S"
    }
  ]
}
```

## Get All pools

GET /liquidity\_pools{?cursor,order,limit}

Horizon will provide an endpoint to enumerate all available liquidity pools on the Stellar network. The response will consist of a list of pool objects:

```
[
  {
    "id": "abcdef",
    "paging_token": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",
```

```
"total_trustlines": "300",
"total_shares": "5000",
"reserves": [
  {
    "amount": "1000.0000005",
    "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"  },
  {
    "amount": "2000.0000000",
    "asset": "PHP:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
  }
]
},
{
  "id": "fghij",
  "paging_token": "fghij",
  "fee_bp": 30,
  "type": "constant_product",
  "total_trustlines": "500",
  "total_shares": "6000",
  "reserves": [
    {
      "amount": "1000.0000005",
      "asset": "ETH:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
    },
    {
      "amount": "2000.0000000",
      "asset": "USD:GBUQWP3BOUZ34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
    }
  ]
}
]
```

## Get pools by reserves

```
GET
/liquidity_pools?reserves=EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNH
JQIYGGKSMOZAHOOS2S,PHP:GBUQWP3BOUZ34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLL
AW7YKGU6EP
```

Horizon will provide an endpoint to find all liquidity pools which contain a given set of reserve assets. The response will consist of a list of pool objects:

```
[
  {
    "id": "abcdef",
    "paging_token": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",
    "total_trustlines": "300",
    "total_shares": "5000",
    "reserves": [
      {
        "amount": "1000.0000005",
        "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
      },
      {
        "amount": "2000.0000000",
        "asset": "PHP:GBUQWP3BOUZ34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
      }
    ]
  }
]
```

## Get Account

```
GET
/accounts/GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U
```

Horizon already provides an endpoint to query account details. This endpoint will be modified in the case that the account is an LP for a liquidity pool. We will add an extra balance object in the response to describe how many shares of the liquidity pool are held by the account.

## BREAKING API CHANGE

In the account response there is a **balances** field which consists of a list of balance objects. We will introduce the following breaking changes on the balance object which is included in the **balances** list:

- **asset\_type** can be **liquidity\_pool\_shares**. Previously the **asset\_type** field was restricted to: **native**, **credit\_alphanum4**, or **credit\_alphanum12**.
- **buying\_liabilities**, **selling\_liabilities**, **asset\_code**, and **asset\_issuer** are removed from the response for pool shares because they are not relevant to liquidity pools.

```
{
  "id": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
  "account_id": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
  "sequence": "24739097524306468",
  "subentry_count": 3,
  "inflation_destination": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
  "last_modified_ledger": 23569316,
  "num_sponsoring": 0,
  "num_sponsored": 0,
  "thresholds": {
    "low_threshold": 5,
    "med_threshold": 0,
    "high_threshold": 0
  },
  "flags": {
    "auth_required": false,
    "auth_revocable": true,
    "auth_immutable": false,
    "auth_clawback_enabled": true
  },
  "balances": [
    {
      "balance": "1.0000005",
      "limit": "922337203685.4775807",
      "buying_liabilities": "0.0000000",
      "selling_liabilities": "0.0000000",
    }
  ]
}
```

```

    "last_modified_ledger": 22651481,
    "is_authorized": true,
    "is_clawback_enabled": false,
    "asset_type": "credit_alphanum4",
    "asset_code": "EURT",
    "asset_issuer": "GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
  },
  {
    "balance": "0.0000000",
    "limit": "922337203685.4775807",
    "buying_liabilities": "0.0000000",
    "selling_liabilities": "0.0000000",
    "last_modified_ledger": 7877447,
    "is_authorized": false,
    "is_clawback_enabled": false,
    "asset_type": "credit_alphanum4",
    "asset_code": "PHP",
    "asset_issuer": "GBUQWP3BOUZX34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
  },
  {
    "liquidity_pool_id": "abcdef",
    "asset_type": "liquidity_pool_shares",
    "balance": "10",
    "limit": "10000",
    "last_modified_ledger": 7877447,
    "is_authorized": false,
    "is_clawback_enabled": false
  }
],
"signers": [
  {
    "weight": 10,
    "key": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
    "type": "ed25519_public_key"
  }
],
}

```

## Get Accounts by pool

GET /accounts?liquidity\_pool=abcdef

Horizon already provides an endpoint to query accounts by one of three filters: signer, asset, or sponsor. We will add another filter, **liquidity\_pool**, which will return all accounts who have a trustline for the given pool.

```
[
  {
    "id": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
    "account_id": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
    "sequence": "24739097524306468",
    "subentry_count": 3,
    "inflation_destination": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
    "last_modified_ledger": 23569316,
    "num_sponsoring": 0,
    "num_sponsored": 0,
    "thresholds": {
      "low_threshold": 5,
      "med_threshold": 0,
      "high_threshold": 0
    },
    "flags": {
      "auth_required": false,
      "auth_revocable": true,
      "auth_immutable": false,
      "auth_clawback_enabled": true
    },
    "balances": [
      {
        "balance": "1.0000005",
        "limit": "922337203685.4775807",
        "buying_liabilities": "0.0000000",
        "selling_liabilities": "0.0000000",
        "last_modified_ledger": 22651481,
        "is_authorized": true,
        "is_clawback_enabled": false,
        "asset_type": "credit_alphanum4",
        "asset_code": "EURT",
        "asset_issuer": "GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOSS"
      },
      {
        "balance": "0.0000000",
        "limit": "922337203685.4775807",
```

```

    "buying_liabilities": "0.0000000",
    "selling_liabilities": "0.0000000",
    "last_modified_ledger": 7877447,
    "is_authorized": false,
    "is_clawback_enabled": false,
    "asset_type": "credit_alphanum4",
    "asset_code": "PHP",
    "asset_issuer": "GBUQWP3BOUZ34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
  },
  {
    "liquidity_pool_id": "abcdef",
    "asset_type": "liquidity_pool_shares",
    "balance": "10",
    "limit": "10000",
    "last_modified_ledger": 7877447,
    "is_authorized": false,
    "is_clawback_enabled": false
  }
],
"signers": [
  {
    "weight": 10,
    "key": "GDI73WJ4SX7LOG3XZDJC3KCK6ED6E5NBYK2JUBQSPBCNNWEG3ZN7T75U",
    "type": "ed25519_public_key"
  }
]
}
]

```

## Liquidity Pool Operations / Transactions

[CAP 38](#) introduces two new operations which act on liquidity pools. There is the deposit operation which allows an account to deposit reserve assets into a liquidity pool in exchange for pool shares. And there is the withdrawal operation which allows an account to claim reserve assets for the underlying pool shares.

Also, the change trust operation will be extended to allow the creation, modification, and deletion of pool share trust lines.

Horizon will provide an endpoint for fetching all operations which act on a specific pool:

```
GET /liquidity_pools/<liquidity-pool-id>/operations
```

The operations included in the response are the change trust, withdrawal, and deposit operations described above along with any path payment operations which transact against the liquidity pool.

Horizon will also provide an endpoint for fetching all transactions which contain operations that act on a specific pool:

```
GET /liquidity_pools/<liquidity-pool-id>/transactions
```

Below you can find the JSON definition for the withdrawal and deposit operation objects.

## Deposit Liquidity Operation JSON

```
{
  "id": "3697472920621057",
  "paging_token": "3697472920621057",
  "transaction_successful": true,
  "source_account": "GBB4JST32UWKOLGYYSCEYBHBFCOFL2TGBHVDVOMZP462ET4ZRD4ULA7S2L",
  "type": "liquidity_pool_deposit",
  "type_i": 22,
  "created_at": "2021-11-18T03:47:47Z",
  "transaction_hash": "43ed5ce19190822ec080b67c3ccbab36a56bc34102b1a21d3ee690ed3bc23378",
  "liquidity_pool_id": "abcdef",
  "reserves_max": [
    {
      "asset": "JPY:GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "1000.0000005"
    },
    {
      "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S",
      "amount": "3000.0000005"
    }
  ],
  "min_price": "0.2680000",
  "min_price_r": {
    "n": 67,
    "d": 250
  },
  "max_price": "0.3680000",
  "max_price_r": {
    "n": 73,
    "d": 250
  },
  "reserves_deposited": [
    {
      "asset": "JPY:GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "983.0000005"
    },
    {
      "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S",
      "amount": "2378.0000005"
    }
  ]
}
```



```
"shares_received": "1000"
}
```

## Withdraw Liquidity Operation JSON

```
{
  "id": "3697472920621057",
  "paging_token": "3697472920621057",
  "transaction_successful": true,
  "source_account": "GBB4JST32UWKOLGYYSCEYBHBFCOFL2TGBHVDVOMZP462ET4ZRD4ULA7S2L",
  "type": "liquidity_pool_withdraw",
  "type_i": 23,
  "created_at": "2021-11-18T03:47:47Z",
  "transaction_hash": "43ed5ce19190822ec080b67c3ccbab36a56bc34102b1a21d3ee690ed3bc23378",
  "liquidity_pool_id": "abcdef",
  "reserves_min": [
    {
      "asset": "JPY:GBVAOIAACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "1000.0000005"
    },
    {
      "asset": "EURT:GAP5LET0V6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAH00S2S",
      "amount": "3000.0000005"
    }
  ],
  "shares": "200",
  "reserves_received": [
    {
      "asset": "JPY:GBVAOIAACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "993.0000005"
    },
    {
      "asset": "EURT:GAP5LET0V6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAH00S2S",
      "amount": "2478.0000005"
    }
  ]
}
```

## Change Trust Operation JSON

### BREAKING API CHANGE

- **asset\_type** can be **liquidity\_pool\_shares**. Previously the **asset\_type** field was restricted to: **native**, **credit\_alphanum4**, or **credit\_alphanum12**.
- **trustee**, **asset\_code**, and **asset\_issuer** are removed from the response for pool shares because they are not relevant to liquidity pools.

```
{
  "id": "157639717969326081",
  "paging_token": "157639717969326081",
  "transaction_successful": true,
  "source_account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
}
```

```

"type": "change_trust",
"type_i": 6,
"created_at": "2021-08-04T20:01:24Z",
"transaction_hash": "941f2fa2101d1265696a3c7d35e7688cd210324114e96b64a386ab55f65e488f",
"asset_type": "liquidity_pool_shares",
"liquidity_pool_id": "abcdef",
"limit": "1000",
"trustor": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG"
}

```

## Liquidity Pool Effects

Horizon will introduce several new effect types to describe how the state of a liquidity pool changes whenever liquidity is deposited, withdrawn, or traded against.

Horizon will provide an endpoint for retrieving all effects for a given liquidity pool:

```
GET /liquidity_pools/<pool-id>/effects
```

Below you can find the JSON definitions for the liquidity pool effects:

## Deposit Liquidity Effect JSON

```

{
  "id": "0000000012884905986-0000000002",
  "paging_token": "12884905986-2",
  "account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
  "type": "liquidity_pool_deposited",
  "type_i": 81,
  "created_at": "2021-11-18T03:15:54Z",
  "liquidity_pool": {
    "id": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",
    "total_trustlines": "300",
    "total_shares": "5000",
    "reserves": [
      {
        "amount": "1000.0000005",
        "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
      },
      {
        "amount": "2000.0000000",
        "asset": "PHP:GBUQWP3BOUZX34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
      }
    ]
  }
}

```

```

    ]
  },
  "reserves_deposited": [
    {
      "asset": "JPY:GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "983.0000005"
    },
    {
      "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S",
      "amount": "2378.0000005"
    }
  ],
  "shares_received": "1000"
}

```

## Withdraw Liquidity Effect JSON

```

{
  "id": "0000000012884905986-0000000002",
  "paging_token": "12884905986-2",
  "account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
  "type": "liquidity_pool_withdrew",
  "type_i": 82,
  "created_at": "2021-11-18T03:15:54Z",
  "liquidity_pool": {
    "id": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",
    "total_trustlines": "299",
    "total_shares": "4000",
    "reserves": [
      {
        "amount": "7.0000005",
        "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
      },
      {
        "amount": "1.0000000",
        "asset": "PHP:GBUQWP3BOUZX34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
      }
    ]
  },
  "reserves_received": [
    {
      "asset": "JPY:GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "993.0000005"
    },
    {
      "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S",
      "amount": "2478.0000005"
    }
  ],
  "shares_redeemed": "1000"
}

```

## Liquidity Pool Trade Effect JSON

```

{

```

```

{id": "0000000012884905986-0000000002",
"paging_token": "12884905986-2",
"account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
"type": "liquidity_pool_trade",
"type_i": 83,
"created_at": "2021-11-18T03:15:54Z",
"liquidity_pool": {
  "id": "abcdef",
  "fee_bp": 30,
  "type": "constant_product",
  "total_trustlines": "300",
  "total_shares": "5000",
  "reserves": [
    {
      "amount": "1000.0000005",
      "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
    },
    {
      "amount": "2000.0000000",
      "asset": "PHP:GBUQWP3BOUZ34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
    }
  ]
},
"sold": {
  "asset": "JPY:GBVAOAIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
  "amount": "983.0000005"
},
"bought": {
  "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S",
  "amount": "2378.0000005"
}
}

```

## Liquidity Pool Created Effect JSON

```

{
  "id": "0000000012884905986-0000000002",
  "paging_token": "12884905986-2",
  "account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
  "type": "liquidity_pool_created",
  "type_i": 84,
  "created_at": "2021-11-18T03:15:54Z",
  "liquidity_pool": {
    "id": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",

```

```

"total_trustlines": "1",
"total_shares": "0",
"reserves": [
  {
    "amount": "0",
    "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
  },
  {
    "amount": "0",
    "asset": "PHP:GBUQWP3BOUZX34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
  }
]
}
}

```

## Liquidity Pool Removed Effect JSON

```

{
  "id": "0000000012884905986-0000000002",
  "paging_token": "12884905986-2",
  "account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
  "type": "liquidity_pool_removed",
  "type_i": 85,
  "created_at": "2021-11-18T03:15:54Z",
  "liquidity_pool_id": "abcdef"
}

```

## Liquidity Revoked Effect JSON

```

{
  "id": "0000000012884905986-0000000002",
  "paging_token": "12884905986-2",
  "account": "GAAZI4TCR3TY5OJHCTJC2A4QSY6CJWJH5IAJTGKIN2ER7LBNVKOCCWN7",
  "type": "liquidity_pool_revoked",
  "type_i": 86,
  "created_at": "2021-11-18T03:15:54Z",
  "liquidity_pool": {
    "id": "abcdef",
    "fee_bp": 30,
    "type": "constant_product",
    "total_trustlines": "299",
    "total_shares": "4000",
    "reserves": [
      {
        "amount": "7.0000005",
        "asset": "EURT:GAP5LETOV6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAHOOS2S"
      },
      {
        "amount": "1.0000000",
        "asset": "PHP:GBUQWP3BOUZX34TOND2QV7QQ7K7VJTG6VSE7WMLBTMDJLLAW7YKGU6EP"
      }
    ]
  },
  "reserves_revoked": [
    {
      "asset": "JPY:GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
      "amount": "993.0000005",
    }
  ]
}

```

```

    "claimable_balance_id": "cbid1235"
  },
  {
    "asset": "EURT:GAP5LET0V6YIE62YAM56STDANPRDO7ZFDBGSNHJQIYGGKSMOZAH00S2S",
    "amount": "2478.0000005",
    "claimable_balance_id": "idcbd1234"
  }
],
"shares_revoked": "1000"
}

```

## BREAKING API CHANGE FOR TRUSTLINE EFFECTS

- **asset\_type** can be **liquidity\_pool\_shares**. Previously the **asset\_type** field was restricted to: **native**, **credit\_alphanum4**, or **credit\_alphanum12**.
- **asset\_code**, and **asset\_issuer** are removed from the response for pool shares because they are not relevant to liquidity pools.

## Trustline Created Effect JSON

```

{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_created",
  "type_i": 20,
  "created_at": "2021-08-04T20:01:24Z",
  "asset_type": "liquidity_pool_shares",
  "liquidity_pool_id": "abcdef",
  "limit": "1000"
}

```

## Trustline Updated Effect JSON

```

{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_updated",
  "type_i": 22,
  "created_at": "2021-08-04T20:01:24Z",
  "asset_type": "liquidity_pool_shares",
  "liquidity_pool_id": "abcdef",
  "limit": "2000"
}

```

## Trustline Removed Effect JSON

```
{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_removed",
  "type_i": 21,
  "created_at": "2021-08-04T20:01:24Z",
  "asset_type": "liquidity_pool_shares",
  "liquidity_pool_id": "abcdef",
  "limit": "0.0000000"
}
```

## BREAKING API CHANGE FOR TRUSTLINE SPONSORSHIP EFFECTS

- If the sponsorship is for a pool shares trustline, the **asset** field will be omitted and instead there will be a **liquidity\_pool\_id** field present.

## Trustline Sponsorship Created Effect JSON

```
{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_sponsorship_created",
  "type_i": 63,
  "created_at": "2021-08-04T20:01:24Z",
  "asset_type": "liquidity_pool_shares",
  "liquidity_pool_id": "abcdef",
  "sponsor": "GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM"
}
```

## Trustline Sponsorship Updated Effect JSON

```
{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_sponsorship_updated",
  "type_i": 64,
  "created_at": "2021-08-04T20:01:24Z",

```

```
"asset_type": "liquidity_pool_shares",
"liquidity_pool_id": "abcdef",
"former_sponsor": "GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM",
"new_sponsor": "GB76HCCL7TVCM7XLWUC7ZK536UTLRLZG64TT5DKZDONXWBTJU7OXRDCR"
}
```

## Trustline Sponsorship Removed Effect JSON

```
{
  "id": "0157639717969326081-0000000001",
  "paging_token": "157639717969326081-1",
  "account": "GBBWI7TEVQBPEUXKYNGI3GBAH7EHFEREONKK3UK56ZSLJIDIYHQJCVSG",
  "type": "trustline_sponsorship_removed",
  "type_i": 65,
  "created_at": "2021-08-04T20:01:24Z",
  "asset_type": "liquidity_pool_shares",
  "liquidity_pool_id": "abcdef",
  "former_sponsor": "GBVAOIACNSB7OVUXJYC5UE2D4YK2F7A24T7EE5YOMN4CE6GCHUTOUQXM"
}
```

## Trades

Horizon already has a [trades endpoint](#) which lists all trades occurring on the Stellar DEX. We will need to modify the endpoint so that it also includes trades which occur on liquidity pools. We will add an extra query parameter to the trades endpoint, `trade_type`, which defaults to `all`.

```
GET /trades?trade_type={orderbook,liquidity_pool,all}
```

The `trade_type` filter will work in addition to the base and counter asset pair filters. However, Horizon will reject requests which attempt to set `trade_type=liquidity_pool` when using the offer id filter.

The JSON definition of a trade will contain an extra field `trade_type` which indicates whether the trade occurred against the DEX or on a liquidity pool.

## Orderbook Trade JSON

### BREAKING API CHANGE

- The `price` fraction will represent its numerator and denominator values as strings to be consistent with the liquidity pool trade JSON object.



```

{
  "id": "157235845014249474-0",
  "paging_token": "157235845014249474-0",
  "ledger_close_time": "2021-07-29T21:10:53Z",
  "offer_id": "738916928",
  "trade_type": "orderbook",
  "base_offer_id": "4768921863441637378",
  "base_account": "GAPP4CS5KSZSHF6A24JWZ2W6MZKPCQKCAW44Y7FFX46WS76WMBDYWKWH",
  "base_amount": "0.0002007",
  "base_asset_type": "native",
  "counter_offer_id": "738916928",
  "counter_account": "GDW634JZX3VMEF2RZTCJTT34RITIMNX46QOGTYHCJEJL3MM7BLOQ6HOW",
  "counter_amount": "0.0022300",
  "counter_asset_type": "credit_alphanum4",
  "counter_asset_code": "VZT",
  "counter_asset_issuer": "GBENYXZDFFR2J4F4DB3YPBBAM244TXYOTIOUQI5DBT3OKUU4ZJ2M7NO",
  "base_is_seller": false,
  "price": {
    "n": "10000000",
    "d": "899997"
  }
}

```

## Liquidity Pool Trade JSON

### BREAKING API CHANGE

- The **price** fraction will represent its numerator and denominator values as strings because they will be 64 bit integers.
- Fields that refer to offer ids (e.g. **base\_offer\_id** or **counter\_offer\_id**) are removed from the response object.
- Only one side of the trade will be executed by an account (either **base\_account** or **counter\_account**). The other side of the trade will be executed against a liquidity pool (either **base\_liquidity\_pool\_id** or **counter\_liquidity\_pool\_id**).

```

{
  "id": "157235845014249474-0",
  "paging_token": "157235845014249474-0",
  "ledger_close_time": "2021-07-29T21:10:53Z",
  "trade_type": "liquidity_pool",
  "base_liquidity_pool_id": "abcdef",
  "liquidity_pool_fee_bp": 30,
  "base_amount": "0.0002007",
  "base_asset_type": "native",
  "counter_account": "GDW634JZX3VMEF2RZTCJTT34RITIMNX46QOGTYHCJEJL3MM7BLOQ6HOW",
  "counter_amount": "0.0022300",
  "counter_asset_type": "credit_alphanum4",
  "counter_asset_code": "VZT",
  "counter_asset_issuer": "GBENYXZDFFR2J4F4DB3YPBBAM244TXYOTIOUQI5DBT3OKUU4ZJ2M7NO",
  "base_is_seller": false,
  "price": {
    "n": "10000000",
    "d": "899997"
  }
}

```

}

Horizon will also provide an endpoint for retrieving all trades for a given liquidity pool:

```
GET /liquidity_pools/<pool-id>/trades
```

The response for this endpoint will only contain trade objects of type “liquidity\_pool.”

## Trade Aggregations

There will be no modifications to the [trade aggregations endpoint](#). The format of the request and the response JSON will remain as is. Liquidity pool trades **will be included** in the aggregations. But, that means it will not be possible to filter trade aggregations by trade type. The reason for this limitation is due to the complexity of Horizon’s trade aggregation implementation.

Trade aggregations are derived from a precomputation which is updated during ingestion. It took a considerable amount of effort to make the current implementation efficient. Therefore we don’t want to add any extra complexity to trade aggregations by adding an extra trade type dimension.