`RootConfigData` JSON Documents

Evgeni Pandurksi

2023-08-23

Overview

This document specifies the format for `RootConfigData` documents.

In Swaptacular, the *currency issuers* are able to configure various parameters of their respective currencies (like the interest rate). In order for the chosen currency parameters to take effect, they must be put together in a machine-readable document, and sent to the *accounting authority node* which is responsible for managing the given currency. The `RootConfigData` document format, which will be specified here, is one of the standard machine-readable formats that can be used to relay the currency parameters chosen by the issuer, to the accounting authority node responsible for the currency.

Note that in Swaptacular's terminology, the word "debtor" means a Swaptacular currency, with its respective issuer.

Note: The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

The Root Account's `config_data` Field

The network protocol that governs the communication between accounting authority nodes and their peers, is the Swaptacular Messaging Protocol (SMP). Every SMP account is uniquely identified by a `(debtor_id, creditor_id)` number pair.

In SMP, a special account called "*The Root Account*" (or "the debtor's account")[1] is used to issue new currency tokens into existence, and to configure the currency parameters. Each currency issuer (aka debtor) should use the `config_data` text field of its root account, to configure the parameters of its currency.

[1] The `creditor_id` for each debtor's root account is `0` (zero).

That is: To set or update the parameters of its currency, the currency issuer (aka the debtor) should send a `ConfigureAccount` SMP message for the debtor's root account, and the `config_data` field of this message should contain the currency parameters, encoded in a standard machine-readable format.

The `RootConfigData` Machine-readable Format

`RootConfigData` documents are JSON documents whose structure and content can be correctly validated by the JSON Schema specified below. UTF-8 encoding MUST always be used for `RootConfigData` documents.

All compliant accounting authority node implementations SHOULD support the `RootConfigData` format, as a standard way of specifying currency parameters, in the `config_data` field of root accounts.[2] [3]

JSON Schema

Type: `object`

path: #

This schema accepts additional properties.

Properties

- **type** `required`
  - Type: `string`
  - path: #/properties/type
  - The value must match this pattern: `^RootConfigData(-v[1-9][0-9]{0,5})?$`
- **rate**

  Optional annual rate (in percents) at which interest accumulates on creditors' accounts. Accounting authority nodes MAY decide to limit the range of allowed values for this parameter. However, zero MUST always be allowed, and all values between -50 an 100 SHOULD be allowed. Note that values smaller than -100 do not make sense, and SHOULD not be allowed.

  - Type: `number`
  - path: #/properties/rate
  - Default: `0.0`

[2] Accounting authority nodes MAY support other machine-readable formats as well.

[3] Note that the SMP specification requires that an empty string must always be a valid value for the `config_data` field, which represents the default configuration settings. In the root account's case, the default configuration settings are: zero interest rate, and no issuing limits.

- **limit**

  Optional limit for the total issued amount. The balance on the debtor's root account will be allowed to go negative, as long as it does not exceed the configured `limit` (with a negative sign). This gives currency issuers the ability to reliably restrict the total amount that they allow themselves to issue.

  The value must be a non-negative 64-bit integer. Note that processing integers outside the safe range from `-(2 ** 53 - 1)` to `2 ** 53` could be a problem for the standard ECMAScript JSON parser and serializer.

  - Type: `integer`
  - path: #/properties/limit
  - Range: between 0 and 9223372036854775807
  - Default: `9223372036854775807`

- **info**

  Optional additional information about the debtor.

  - path: #/properties/info
  - &ref: #/definitions/DebtorInfo

Definitions

`DebtorInfo`

Type: `object`

path: #/definitions/DebtorInfo

This schema accepts additional properties.

Properties

- **type** `required`

  - Type: `string`
  - path: #/definitions/DebtorInfo/properties/type
  - The value must match this pattern: `^DebtorInfo(-v[1-9][0-9]{0,5})?$`

- **iri** `required`

  A link (Internationalized Resource Identifier) referring to a document containing information about the debtor.

  - Type: `string`
  - path: #/definitions/DebtorInfo/properties/iri
  - String format must be a "iri"
  - Length: between 1 and 200

- **contentType**

  Optional MIME type of the document that the `iri` field refers to.

  - Type: `string`
  - path: #/definitions/DebtorInfo/properties/contentType
  - Length: <= 100

- **sha256**

  Optional SHA-256 cryptographic hash (Base16 encoded) of the content of the document that the `iri` field refers to.

  - Type: `string`
  - path: #/definitions/DebtorInfo/properties/sha256
  - The value must match this pattern: ^[0-9A-F]{64}$

JSON Schema File

This is the JSON Schema file, for validating `RootConfigData` documents:

```
{
  "definitions": {
    "DebtorInfo": {
      "type": "object",
      "properties": {
        "type": {
          "type": "string",
          "pattern": "^DebtorInfo(-v[1-9][0-9]{0,5})?$"
        },
        "iri": {
          "type": "string",
          "minLength": 1,
          "maxLength": 200,
          "format": "iri",
        },
        "contentType": {
          "type": "string",
          "maxLength": 100,
        },
        "sha256": {
          "type": "string",
          "pattern": "^[0-9A-F]{64}$",
        }
      },
      "required": [
        "type",
        "iri"
```

```
      ],
      "additionalProperties": true
    }
  },
  "type": "object",
  "properties": {
    "type":  {
      "type": "string",
      "pattern": "^RootConfigData(-v[1-9][0-9]{0,5})?$"
    },
    "rate": {
      "type": "number",
      "format": "float",
      "default": 0.0,
    },
    "limit": {
      "type": "integer",
      "format": "int64",
      "minimum": 0,
      "maximum": 9223372036854775807,
      "default": 9223372036854775807,
    },
    "info": {
      "$ref": "#/definitions/DebtorInfo",
    }
  },
  "required": [
    "type"
  ],
  "additionalProperties": true
}
```