Input Files

Overview

The input file is a JSON file that contains the configuration for generating the state file. It specifies the overall workflow and individual worker configurations. Based on the provided examples, here's a detailed breakdown of the input file structure and its options.

Input File Structure

```
{
  "user_code": "string",
  "configuration_code": "string",
  "name": "string",
  "schedule": null,
  "workers": [
    {
      "order": "integer",
      "configuration_code": "string",
      "name": "string",
      "user_code": "string",
      "state_type": "string",
      "download_options": {
        "date_from": "string | null",
        "date_to": "string | null",
        "type": "string | null",
        "periodicity": "string | null",
        "portfolios": ["string"] | null,
        "secret": "string | null"
      },
      "data_options": {
        "global_status": "string | null",
        "source": "string | null",
        "type": "string | null",
        "portfolios": null
```

```
      },
      "import_options": {
        "scheme": "string | null",
        "import_type": "string | null",
        "pricing_policy": "string | null"
      },
      "calculation_options": {
        "date_from": null,
        "date_to": null,
        "portfolios": null
      },
      "state_options": {
        "input_path": "string | null"
      }
    }
  ]
}
```

Field Descriptions

1. **Root Level Fields**
   - `user_code`: Always "workflow-manager".
   - `configuration_code`: Always "com.finmars.standard-workflow".
   - `name`: A descriptive name for the workflow, e.g., "Exante Historical - Step 1 (Download): positions, transactions".
   - `schedule`: Deprecated. Always null.
2. **Worker Fields**
   - `order`: The execution order of the worker within the workflow (integer).
   - `configuration_code`: Usually "com.finmars.standard-workflow".
   - `name`: A descriptive name for the worker's task.
   - `user_code`: Identifies the specific task, e.g., "download-exante-positions", "preprocess-exante-transactions", "generate-state".
   - `state_type`: Can be "period", "files", or "fixed".
3. **Download Options**
   - `date_from`: Start date for data retrieval, e.g., "2024-01-01" or null.
   - `date_to`: End date for data retrieval, e.g., "2024-07-14" or null.
   - `type`: Can be "day", "period", or null.
   - `periodicity`: Can be "monthly" or null.
   - `portfolios`: An array of portfolio identifiers or null.
   - `secret`: A string identifier for authentication, e.g., "itech-demo" or null.
4. **Data Options**
   - `global_status`: Can be "initial_data", "to_import", or null.

- `source`: Usually "exante" or null.
- `type`: Can be "positions", "transactions", "instruments", or null.
- `portfolios`: Usually null in the provided examples.
- `sync_to`: Used in "sync-files" tasks, e.g., "preprocessed".

5. **Import Options**
   - `scheme`: Import scheme identifier, e.g., "com.finmars.standard-import-from-file:accounts.account:account".
   - `import_type`: Can be "simple", "transaction", or null.
   - `pricing_policy`: Usually "com.finmars.standard-pricing:standard" or null.
6. **Calculation Options**
   - `date_from`: Usually null in the provided examples.
   - `date_to`: Usually null in the provided examples.
   - `portfolios`: Usually null in the provided examples.
7. **State Options**
   - `input_path`: Path to the next workflow file, e.g., "/input-workflows/exante/exante_historical_step_2.json" or null.

Example Input File

Here's an example based on the "exante_historical_step_1.json" file:

```
{
  "user_code": "workflow-manager",
  "configuration_code": "com.finmars.standard-workflow",
  "name": "Exante Historical - Step 1 (Download): positions, transactions",
  "workers": [
    {
      "order": 1,
      "configuration_code": "com.finmars.standard-workflow",
      "name": "Download Positions",
      "user_code": "download-exante-positions",
      "state_type": "period",
      "download_options": {
        "date_from": "2024-01-01",
        "date_to": "2024-07-14",
        "type": "day",
        "periodicity": "monthly",
        "portfolios": ["Portfolio_007"],
        "secret": "secret-path-demo"
      },
      "data_options": {
```

```
        "global_status": null,
        "source": null,
        "type": null,
        "portfolios": null
      },
      "import_options": {
        "scheme": null,
        "import_type": null,
        "pricing_policy": null
      },
      "calculation_options": {
        "date_from": null,
        "date_to": null,
        "portfolios": null
      },
      "state_options": {
        "input_path": null
      }
    },
    // ... other workers ...
  ]
}
```

Notes on Input Files

- The structure of the input file remains consistent across different steps of the workflow.
- Each step (represented by a separate JSON file) focuses on specific tasks such as downloading, preprocessing, or importing data.
- The "Generate State" worker is typically the last worker in each step, setting up the next step in the workflow.
- Options that are not relevant for a particular worker are usually set to `null`.
- The `state_type` determines how the worker processes data: "period" for date-based operations, "files" for file-based operations, and "fixed" for single-execution tasks.
- The `user_code` in the worker configuration specifies the exact workflow to be performed, such as downloading, preprocessing, or importing specific types of data.

Revision #5
Created 2024-08-30 14:33:59 UTC by Anton Petrushkaneki
Updated 2024-08-30 14:36:59 UTC by Anton Petrushkaneki