hledger rule files for beginners

Francis Begyn

2020/10/05

Plain text accounting

- small recap: what is plain text accounting?
  - plain text file
  - double-entry
  - tooling to create reports

Automating the import

- Manual imports offer a lot of flexibility
  - they also take time
  - handling backlogs becomes slow
- luckily, `(h)ledger` has to option to import from csv files
  - fast import of backlog
- ledger: `convert`
- hledger: `import`
- We'll mainly focus on the `hledger import` command in this talk, as it is based on the same principles of the `ledger convert` only with additional features.

CSV files (or why I dislike Belgian banks)

- most downloaded csv files (from Belgian banks) need post-processing
  - banks use different, non-standard formats
  - some work in cents, others in euros
  - time formats are incorrect or can't be parsed by `hledger`
  - ...
- 1 transaction per row
- all transaction must have the same amount of columns
- most of this is easily scriptable
- must contain an amount and date
  - other fields can be set in the rules file or read from the csv file as well

hledger rule file

- rule files decribe to hledger how to read the csv file
- rule files are tied to the structure of the csv
- some basic keywords:
  - `skip n`: skips the first `n` lines
  - `fields ...`: comma seperate fied identifier (for classifying columns in the csv)
  - `currency`: can be set in the file, are identifeid from the csv file
  - `accountx`: account #`x` that takes part in the transaction
  - `amountx`: amount of funds transfered matching the `x` account
  - `description`: description of the transaction
  - `comment`: comment that will be added to the transaction

hledger rule file

```
skip 1

fields date, description, amount

# specify the date field's format - not needed here since d
# date-format %-d/%-m/%Y
# date-format %-m/%-d/%Y
# date-format %Y-%h-%d

currency $

account1 assets:bank:checking

if (TO|FROM) SAVINGS
  account2 assets:bank:savings
```

example: Zeus tab

- The processed csv file
  - amount are devided by 100 (cents -> euros)

```
"amount","creditor","debtor","id","issuer","message","time"
8.4,"Zeus","thecy",3932,"iepoev","frieten","2016-10-10T20:3
8.4,"thecy","Zeus",3933,"iepoev","GELD via bancontactapp","
8.4,"Zeus","thecy",3992,"Tap","1 Ice Tea","2016-10-12T17:00
8.4,"Zeus","thecy",4026,"Tap","1 Cola","2016-10-13T18:50:48
8.4,"Zeus","thecy",4033,"Tap","1 Ice Tea","2016-10-13T20:04
8.4,"Zeus","thecy",4057,"Tap","1 Ice Tea","2016-10-14T17:10
8.4,"Zeus","thecy",4062,"Tap","1 Cola and 1 XL paprika","20
8.4,"Zeus","thecy",4150,"thecy","Chinees 18/10","2016-10-18
```

example: Zeus tab

the rule file

```
skip 1
fields am, creditor, debitor,,, msg, date
date-format %FT%T.000%Ez
# equivalent of date-format %Y-%m-%dT%H:%M:%S.000+02:00
account1 assets:be:zeus:tab
account2 expenses:voedsel

description %debitor to %creditor: %msg
amount1 -%am EUR
amount2 %am EUR

if %creditor thecy
    amount1 %am EUR
    account2 income:zeus:tab
    amount2 -%am EUR
```