api-to-roefact by RENware Software Systems

- CHANGELOG 0.5.3rc0
  - 0.5.3rc0 invoice supplier from Excel (#TODO: ... in progress...)

CHANGELOG 0.5.3rc0

`0.5.3rc0` invoice supplier from Excel (#TODO: ... in progress...)

- `240327piu01` build & publish wheel on PyPi
  - ✅ 1. update xl2roefact README add link "API Reference" to DLD doc (to be accessible from PyPi)
  - ✅ 2. update xl2roefact `setup.py` and include documentation directory (`doc/`) in WHEEL & DIST packages
  - ✅ 3. `README_xl2roefact_library.md` review & clean of useless content ("rdinv module logic", "Working directories") and refer it in main xl2roefact README before "Referinta API" bullet link
  - ✅ 4. review xl2roefact README, installation section
  - ✅ 5. build doc & all deliverables & temp make a portal buil
  - ✅ 7. clean `xl2roefact/setup.py` drop imclude dirs that are not under `xl2roefact/xl2roefact/` as not being considered. Also clean `xl2roefact/data/...` line and keep only dir as beong in whole included in python wheel
  - ✅ 8. update `downloads.md` ref all `0.5.3rc0` deliverables, sections title and mark end of life support for 0.2 versions
  - publish on PyPi, build site & publish
- `240325piu-use-new-function` import new function `get_partner_data()` and use it in `rdinv.py` module
  - ✅ 1. include function `get_partner_data()` in `rdinv.py` and test for simple compilation errors => result: PASS
  - ✅ 2. rebuild DLD documentation
  - ✅ 3. use in code to replace actual existing CUSTOMER data retrieve. Test for no change vs previous functionality => result: PASS
  - ✅ 4. make a new call for SUPPLIER data. Test for raw getting data in "...excel...original...data" key => PASS
  - ✅ 5. chk new get data and make needed adjustments
  - ✅ 6. update `rdinv.py` to create all supplier final constructs (like those for customer after getting data). Test result: PASS
  - ✅ 7. update XML-JSON map using code refactored @ `rdinv.py lines 287-309`
  - ✅ 8. clean code of FIXME and other work comments and built DLD doc

- `240323piu-suppl-configs` rollout supplier configuration parameters from `.../xl2roefact/___wk_cust_area_function.py` :
  - ✅ 1. update `config_settings.py` with their definition. Tested PASS
  - ✅ 2. update `data/app_settings.yml` with their definition. Tested PASS. Cleaned file `.../xl2roefact/___wk_cust_area_function.py` to make easier its transport to `rdinv.py`
  - ✅ update `rdinv.py` with their import
  - ✅ test `xl2roefact` : automation on `test-xl2roefact` branch
  - ✅ build DLD (`pdm run build_doc`)
- `upd-shebang` update all xl2roefact modules, drop shebang statement
- `240320piu-invsuppl` code for `INV.SUPP` ... `xl2roefact` invoice supplier (`<cac:AccountingSupplierParty>`)
  - ✅ 1. extracted code to generalize in `.../xl2roefact/___wk_cust_area_function.py` to "engineer it"
  - ✅ 2. wrap code in new function `get_partner_data()` , set its first param `partner_type` for desired function operation and protect function against unknown values
  - ✅ 3. 1st raw refactoring: constants dedicated to CUSTOMER adjusted as UNIFied...PARTNER... and set for customer or supplier depending on partner_type
  - ✅ 4. 2nd raw refactoring: `customer_area` string. This is a keyword, so it is subject to unification based on partner_type resulting `partner_area` variable used as keyword instead of hard code "customer_area"
  - ✅ 5. mark & comment in `rdinv.py` place where call this function, start / end of code that should be dropped by using this function (rdinv.py start line 229, end line 367, marker #FIXME.UNIF.PARTNER.DATA)
  - ✅ 6. updated `invoice_header_area` with `param_invoice_header_area` and set las as function parameter
  - ✅ 7. updated `invoice_customer_info` with `invoice_partner_info`
  - ✅ 8. replace word "customer" with "partner" keeping original case
  - ✅ 9. set new parameter `wks` as replacement of `ws` variable (supposed by original code as already existing). Type of this parameter is `pylightxl.Worksheet` .

Last update: March 30, 2024