

OEO Developer Meeting #82

Pads:

- Notes from last meeting: <https://etherpad.wikimedia.org/p/oeo-dev-81>
- Pad to this meeting: <https://etherpad.wikimedia.org/p/oeo-dev-82>
- Pad for next meeting: <https://etherpad.wikimedia.org/p/oeo-dev-83>

Date: 27./28.6.2024 live in MD, room 018, FIN Gebäude

Participants:

- Moderator:
- Main reporter:
- Next meeting organiser:
- Developers with affiliation:
 - Mirjam (IEE, OVGU)
 - Nele (OvGU)
 - Eugenio (DLR)
 - Christian (RLI)
 - Lukas (ÖI)

Preparation:

- List of agenda topics: <https://etherpad.wikimedia.org/p/oeo-dev-79>
- Ontology interoperability meeting

Agenda:

- **Thursday**
 - 10:00 welcome
 - 10:30 preparation of interoperability discussion for later in the day
 - 11:00 was passiert nach SIROP
 - ab September SIROP zu ende
 - SLE, OvGU
 - NFDI läuft weiter, aber weniger OEO Expertise
 - DLR kann noch bis Ende von SEDOS (2024) bisschen weiter machen
 - Idee:
 - kleinere Ontologieprojekte beantragen
 - nachhaltiges dauerhaftes Ontologieprojekt (Stiftungen, Software, Förderungen für Open Source, Fraunhofer/Helmholtz für Grundfinanzierung?..)
 - zugehen auf BnetzA Angebot Aufbereitung aller ihrer Datensätze in einheitlicher Form (Beispiel: Eugenios Datensatz, RLI, DLR, OvGU, ÖI)

- 11:30 imports
 - Clean import by including it into makefile
 - Tasks :
 - 0. Bring oeo-tools up to date in the main (ontology) repo. @Mirjam
 - 1. Add the oeo-tools as submodule in the OEO repo (Important! to know which commit was used) @Eugenio
 - 2. Add the import tasks to the make file @Eugenio
 - 3 (?) Add an action to perform the imports in GitHub. (also important for people who can't run the makefile. <https://docs.github.com/en/actions/using-workflows/manually-running-a-workflow> @Eugenio
 - 4. Alle Skripte durchgehen und extraction Method auf Subset umstellen @Mirjam
 - Current issue: How to deal with the interoperability of scripts (make files need a lot of setup in windows.)
 - Create own unit ontology? Studi-Arbeit
- 12:00 lunch break
 - Lunch break discussion on units:
 - Bring back original definition of `UO:unit`
 - Use our adapted definition^[1] for `UO:unit` for a new class `OEO:unit`.
 - Add new class `OEO:physical unit` and make it equivalent to `UO:unit`.
 - Make `OEO:currency` a subclass of `OEO:unit`
- 13:30 interoperability discussion with externals (hybrid)
 - <https://etherpad.wikimedia.org/p/oeo-foundry>
- 15:00 coffee break
- 15:20 Diskussion zu Interoperabilität
 - OEO "zerhacken"? - ZB. energy carriers and fuels separieren, facilities als Dimension, policies
 - leider wenig Feedback und Diskussion
 - <https://etherpad.wikimedia.org/p/oeo-foundry>
 - Email: Folien, Poll mit neuem Termin, Feedback (da wenig Zeit für Diskussion) @Mirjam
 - Mögliche orthogonale Energien
 - energy carriers and fuels

- facilities (?)
- energy markets
- sectors
-

- 15:30 Imports II
 - Import mit Subset: Skripte updaten

- 15:50 continuing with quantity values
 - energy als process attribute
 - energy storage function --> energy storage als prozess
 - energy storage process has participant energy storage object
 - energy storage content ist Eigenschaft von energy storage object
 -

- **Friday:**
 - 9:00 - 13:00
 - 9:00 quantity values
 - independent continuants haben spezifische qualities/quantities
 - quantities haben eindeutige quantity values, die weniger ausdifferenziert sind (length value, energy values)
 - add data property has number to quantity value
 -
 - **Welche quantity values sind wirklich quantity values?** Bleibt so bzw. axiome hinzufügen
 - area value
 - coefficient of performance value
 - discount rate
 - fee
 - gross domestic product (später Bezug zu process attribute)
 - gross value added
 - inflation rate (Bezug zu process inflation)
 - marginal cost
 - markdown
 - market revenue
 - markup
 - purchasing power parity
 - efficiency value
 - emission value

- forecast error
 - fraction value (muss aber überarbeitet werden)
 - full load hours
 - global warming potential
 - length value
 - time span (incl. subclasses) label anpassen?
 - transport performance value
 - volume --> in volume value umbenennen? und volume als quantity einführen
 - volumetric flow rate
- **Welche quantity values sind eigentlich quantities?** Verschieben+mit quantity values verknüpfen
- cost
 - monetary price (incl. subclasses)
 - maximum values --> detaillierter anschauen
 - energy storage capacity: quality of energy storage object
 - potential: umbenennen in regional potential; quality of region of relevance
 - flow potential: quality
 - stock potential: quality
 - power rating: quality with quantity value power value
 - slope (--> quantity value angle value)
 - specific space requirement
 - surface azimuth angle --> Ausrichtung im Raum als quantity einführen?
- **Welche quantity values sind process attributes?**
- due
 - remuneration (Überklasse von due?)
 - maximum values
- **Was brauchen wir? Was muss geändert werden? Todos:**
- monetary value (quantity value of cost/price)
 - relational quantity value? (setzt zwei quantities ins verhältnis)
 - als neue Klasse einführen
 - efficiency value
 - fraction values subclass of relational quantity values
 - emission
 - brauch axiom 'has process attribute' some 'emission rate'
 - emission quantity value löschen (wurde zu emission value)

- energy
 - energy amount value wird zu energy value
 - electricity export/import value wird aufgelöst,
 - electricity export/import braucht process attribute was Verbindung zu quantity value hat
 - energy value braucht korrespondierendes process attribute (issue #1865 bleibt offen bis #1812 auch geklärt ist)
- Fraction Value mit oex verknüpfen und andere noch unter fraction value einsortieren
 - als äquivalente Klasse einführen
- Bezug von full load hours zu process über process attribute performance
- market share value zu fraction values
- maximum values in quantities und process attributes einsortieren
 - dann auch power value anschauen
- label von potential ändern
- population count überarbeiten
 - has quality amount?
 - quantity value count
 - population ist object aggregate --> braucht quantity
- klarere Definition von size (amount vs. size)
- angle value
- area per power value als relational quantity value für specific space requirement
- quality orientation einführen: quantity values: slope und surface azimuth angle
- label von time span + subclasses mit value anpassen
- Volume (quantity value + quantity)

- Dispositions sind in bestimmten qualities verankert, also sind nicht gleich, aber hängen zusammen
- Disposition kann Verbindung zwischen quality und process sein

- mehr äquivalente Klassen z.B. length value

topics

- energy transformations
- time stamps
- (has part vs. has proper part)
- policy instrument roles/qualities/functions #1826

- Questions for FN:
 - <https://github.com/OpenEnergyPlatform/oekg/issues/19>
 - relation "without"
 - minimum / maximum

Collection of Tasks:

- Add something @A

