

GreenDELTA

sustainability consulting + software

Linked Life Cycle Data beyond Ontologies

Michael Srocka
GreenDelta GmbH

March 30, 2017
64th Discussion Forum on Life Cycle Assessment, Zürich

Content

1. Data conversion and exchange
2. Linked Data
3. Examples
4. Conclusions

Data conversion and exchange

10 years ago ...

[Home](#) / [Browse](#) / [Development](#) / [Data Formats](#) / [openLCA](#) / Files



[Summary](#) | [Files](#) | [Reviews](#) | [Support](#) | [Wiki](#) | [Tickets](#) ▾ | [News](#) | [Discussion](#) | [Donate](#)

Looking for the latest version? [Download openlca-1.6.1.20170320-win-64bit.zip \(212.5 MB\)](#)

[Home](#) / [openlca_converter](#) / converter1.0

Name ↕	Modified ↕	Size ↕	Downloads / Week ↕
↑ Parent folder			
readme.txt	2007-05-04	1.0 kB	0 <input type="checkbox"/>
GPL_converter1.0.zip	2007-05-04	19.4 MB	1
GPL_converter1.0_source.zip	2007-05-04	1.1 MB	0 <input type="checkbox"/>
Totals: 3 Items		20.6 MB	1

Data conversion and exchange

```
<exchange dataSetInternalID="0">
  <referenceToFlowDataSet
    type="flow data set"
    refObjectId="56ced643-75c7-463b-aa9c-1b32cfd5!
    uri=" ../flows/56ced643-75c7-463b-aa9c-1b32cfd!
      <common:shortDescription xml:lang="en">Steel
    </referenceToFlowDataSet>
    <exchangeDirection>Output</exchangeDirection>
    <meanAmount>1.0</meanAmount>
    <resultingAmount>1.0</resultingAmount>
  </exchange>
```



```
<intermediateExchange
  id="71ee4d00-90a5-416e-9833-30e9369b44b1"
  unitId="487df68b-4994-4027-8fdc-a4dc298257b7"
  amount="1.0"
  intermediateExchangeId="56ced643-75c7-463b-aa9c-1b32cfd5!
  <name>Steel</name>
  <unitName>kg</unitName>
  <outputGroup>0</outputGroup>
</intermediateExchange>
```

Reference data

- Units
- Unit groups
- Quantities
- Elementary flows
- LCIA methods
- ...

Mapping files

"00c2562e-8e79-46a5-a0c5-7667c24d3e7f";"ee93d3b7-ccdc-4348-9c36-fe00f4d18ef4";1.0
"0131c82e-8971-439f-bf4d-3b2ab971b69f";"d07fb7e3-8cd7-4a9c-adbf-244e23f813ff";1.0
"013c4c10-abb1-42ed-8e2d-83cae782f6e7";"25e5eab5-52b8-4bd6-8143-90eefa058c45";1.0
"016cf6a4-41fe-4cd9-ba1a-ebe998e27c0e";"19e2eafe-5129-48b1-b4f2-ee62c2df59cb";1.0
"0182455d-898b-4964-bde1-5d4edf228fe1";"c1513682-45ad-444e-afb5-27c660714e88";1.0
"01ba5949-e357-4755-8385-62358ddffd01";"24d59513-f122-436d-ab94-e936d3bddbde";1.0
"021c2d46-60cc-4da9-8b8b-7dbbb59defba";"0caccb10-c146-4c59-94db-cb342c044636";1.0
"022700df-fdf8-456b-bcf2-1589a2d2e350";"d2823e63-e692-4fa3-b164-09e334237104";1.0
"02a5d443-47d2-4452-aff3-f546e14753e8";"9c2e78e0-38d7-4eb7-a40f-31c7308b1dd6";1.0
"02d3e0f3-a198-4ec3-a7cc-4d5d5f85843f";"4ad79e3d-8f8a-45ab-a43a-d22959687c1e";1.0

Data conversion and exchange

- More data
- Better tools
- Complex models

Data formats

- XML (EcoSpold 1&2, ILCD),
- CSV,
- MS Excel,
- Shapefile,
- ...
- PDF
- ...
- Binary formats

Data are more useful when provided ...

- in the most granular and disaggregated form
- in a machine and human readable format
- in a way that it is easy to connect them

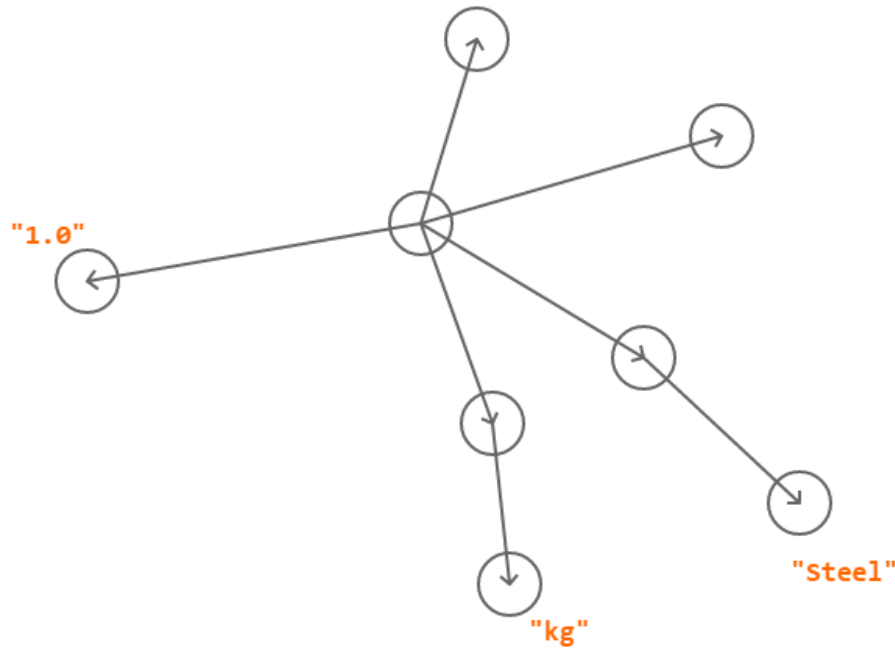
Linked Data

Linked Data

- “make structured data more usable”
- Identify things with URIs
- Use HTTP links
- Model your data in RDF

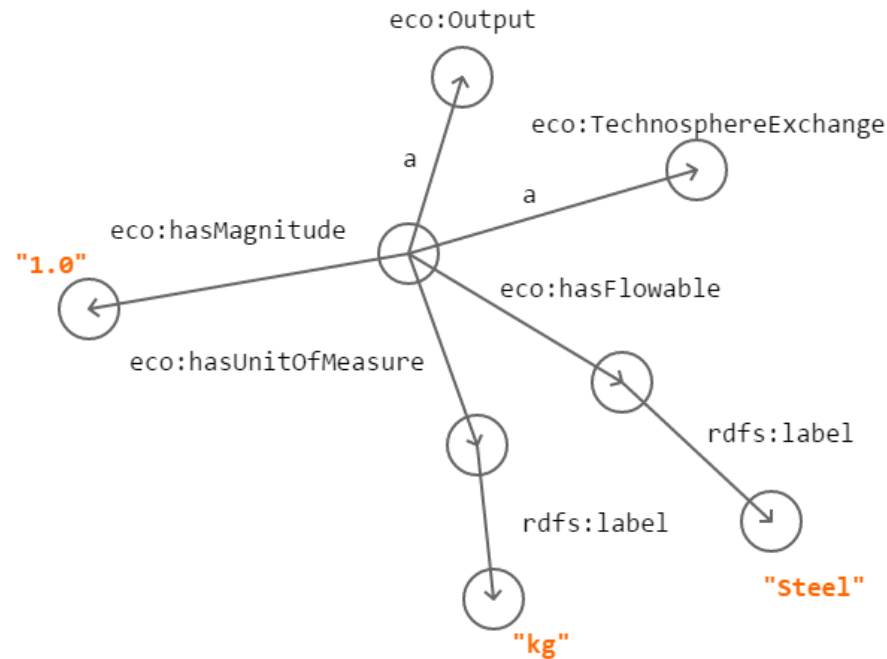
RDF

- A graph-based data model to describe things and their relations



Vocabularies / Ontologies

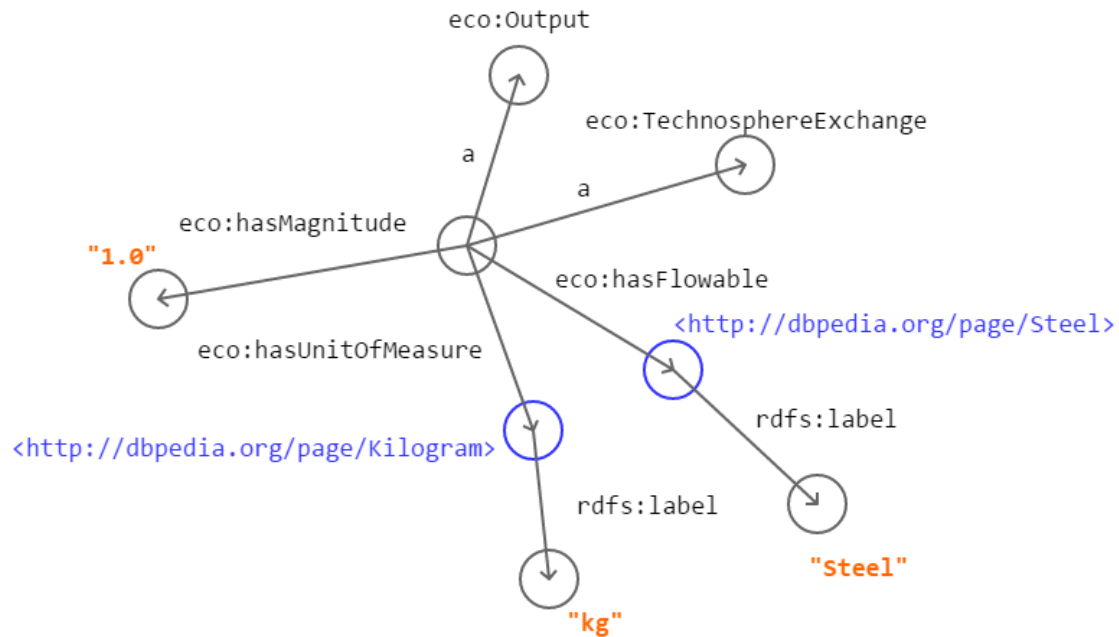
- Terms and definitions of types, properties, relations ...



```
@prefix eco: <http://ontology.earthster.org/eco/core#>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

HTTP Links

- Link to resources that can be retrieved from the web



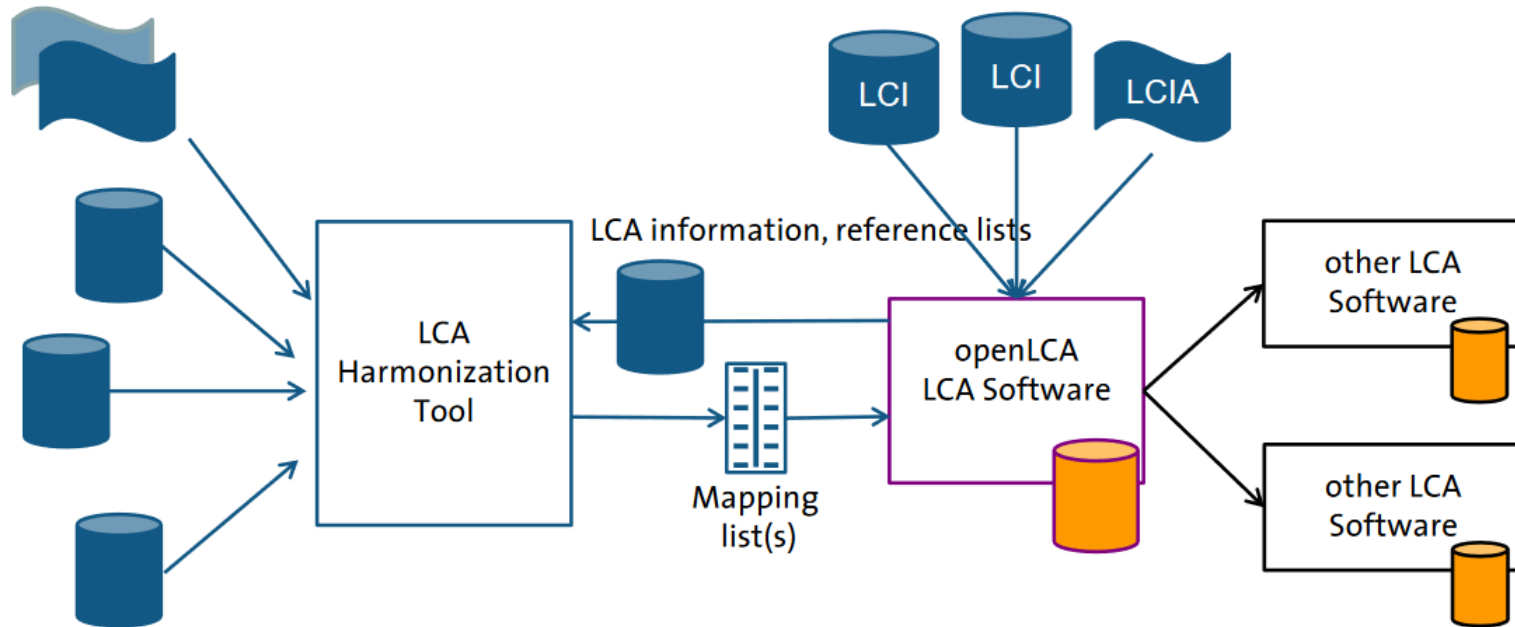
```
@prefix eco: <http://ontology.earthster.org/eco/core#>  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

Examples

The LCA Harmonization Tool and openLCA

Various non LCI data sources, flow lists

Various LCI & LCIA data sources



Source: Ingwersen W., Ciroth A.: Elementary Flow Harmonization with openLCA and the LCA Harmonization Tool. 4th Meeting of the International Forum on LCA cooperation, Shah Alam, Malaysia, March 2015.

Best way to build an RDF import and export for openLCA?

- RDF serialization formats: RDF/XML, Notation-3 (N3), Turtle, N-Triples, RDFa, RDF/JSON

```
@prefix eco: <http://ontology.earthster.org/eco/core#> .  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
@prefix dbp: <http://dbpedia.org/page/> .  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
_:steel_output  
  a eco:TechnosphereExchange ;  
  a eco:Output ;  
  eco:hasFlowable dbp:Steel ;  
  eco:hasUnitOfMeasure dbp:Kilogram ;  
  eco:hasMagnitude "1.0"^^xsd:decimal .
```

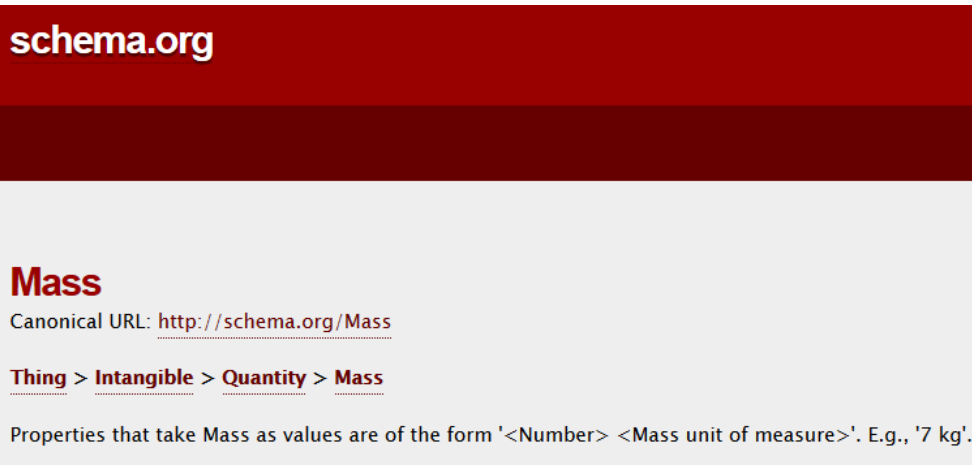
```
dbp:Steel rdfs:label "Steel" .
```

```
dbp:Kilogram rdfs:label "kg" .
```

schema.org & JSON-LD

... Over 10 million sites use Schema.org to markup their web pages and email messages

...



The screenshot shows the schema.org website. At the top, there is a dark red header with the text 'schema.org' in white. Below the header, the word 'Mass' is displayed in a large, bold, dark red font. Underneath 'Mass', the canonical URL is listed as 'http://schema.org/Mass'. A breadcrumb trail reads 'Thing > Intangible > Quantity > Mass'. At the bottom of the screenshot, a text block states: 'Properties that take Mass as values are of the form '<Number> <Mass unit of measure>'. E.g., '7 kg'.'

```
{
  "@context": "http://schema.org/",
  "@type": "MedicalWebPage",
  "publisher": {
    "@type": "Organization",
    "name": "Drugs.com",
    "logo": {
      "@type": "ImageObject",
      "url": "https://www.drugs.c",
      "width": 280,
      "height": 58
    }
  },
  "about": {
    "@type": "Drug",
    "name": "aspirin",
    "nonProprietaryName": "aspirin",
    "pregnancyCategory": "http://sc",
    "pregnancyWarning": "Not classi"
  },
  "name": "aspirin",
  "audience": "Patient",
  "description": "Aspirin is a salicy"
}
```

JSON-LD

- A RDF serialization format
- Just JSON + some annotations

...JSON is the lingua franca of exchanging data over the net and between applications...

<https://appliedgo.net/json/>

```
object
  { }
  { members }
members
  pair
  pair , members
pair
  string : value
array
  [ ]
  [ elements ]
elements
  value
  value , elements
value
  string
  number
  object
  array
  true
  false
  null
```

<http://json.org/>

olca-schema*

Class Flow

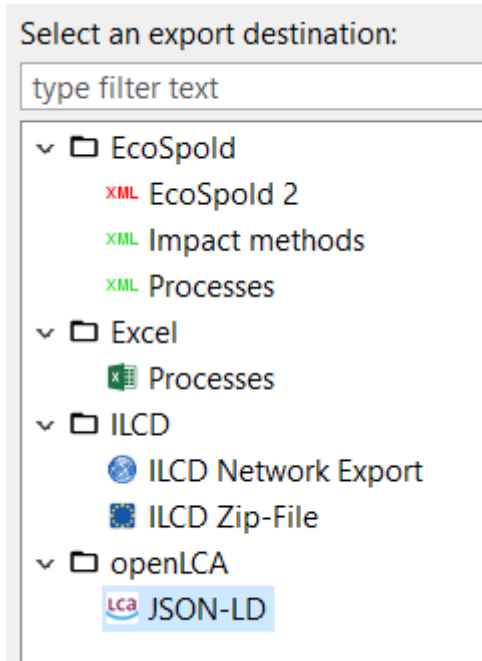
Everything that can be an input or output of a process (e.g. a substance, a product, a waste, a service etc.)

~

Properties:

<u>flowType</u>	FlowType	The type of the flow. Note that this type is more a descriptor of how the flow is handled in calculations.
<u>cas</u>	string	A CAS number of the flow.
<u>formula</u>	string	A chemical formula of the flow.
<u>flowProperties</u>	List[FlowPropertyFactor]	The flow properties (quantities) in which amounts of the flow can be expressed together with conversion factors between these flow flow properties.
<u>location</u>	Location	The location of the flow. Normally the location of a flow is defined by the process location where the flow is an input or output. However, some data formats define a location as a property of a flow.

openLCA JSON-LD import / export



```
{
  "@type": "Exchange",
  "avoidedProduct": false,
  "input": false,
  "amount": 1.0,
  "flow": {
    "@type": "Flow",
    "@id": "56ced643-75c7-463b-aa9c-1b32cfdd58d0",
    "name": "Steel",
    "flowType": "PRODUCT_FLOW"
  },
  "unit": {
    "@type": "Unit",
    "@id": "20aad24-a391-41cf-b340-3e4529f44bde",
    "name": "kg"
  },
  "flowProperty": {
    "@type": "FlowProperty",
    "@id": "93a60a56-a3c8-11da-a746-0800200b9a66",
    "name": "Mass"
  },
  "@id": "f60111a5-2f68-3740-a780-65ba8fd98d80",
  "quantitativeReference": true
}
```

openLCA JSON-LD: @context

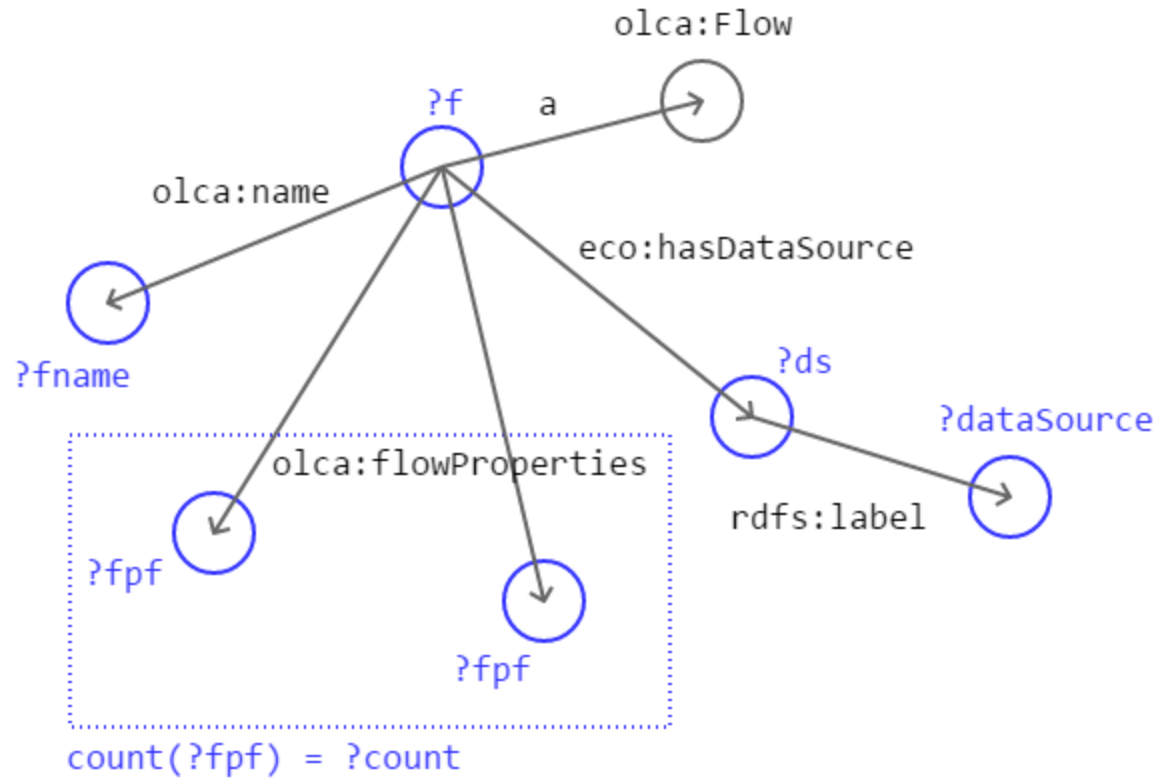
```
{  
  "@vocab": "http://openlca.org/schema/v1.1/",  
  "@base": "http://openlca.org/schema/v1.1/",  
  "modelType": {"@type": "@vocab"},  
  "flowPropertyType": {"@type": "@vocab"},  
  "flowType": {"@type": "@vocab"},  
  "distributionType": {"@type": "@vocab"},  
  "parameterScope": {"@type": "@vocab"},  
  "allocationType": {"@type": "@vocab"},  
  "defaultAllocationMethod": {"@type": "@vocab"},  
  "allocationMethod": {"@type": "@vocab"},  
  "processType": {"@type": "@vocab"},  
  "riskLevel": {"@type": "@vocab"}  
}
```

Using it in LCA-HT ... SPARQL

- select,
- insert,
- delete triples

```
1 PREFIX olca: <http://openlca.org/schema/v1.0/>
2 PREFIX eco: <http://ontology.earthster.org/eco/core#>
3 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4
5 select
6 ?f ?dataSource ?fname ?count
7 where {
8   {select ?f ?dataSource ?fname (count(?fpf) as ?count)
9     where {
10       ?f a olca:Flow .
11       ?f olca:name ?fname .
12       ?f olca:flowProperties ?fpf .
13       ?f eco:hasDataSource ?ds .
14       ?ds rdfs:label ?dataSource .
15     }
16     group by ?f ?fname ?dataSource
17   }
18   filter (?count > 1)
19 }
20 order by ?fname|
```


Using it in LCA-HT ... SPARQL



Or just query the JSON documents ...

```
import json
import os

results = []
for f in os.listdir('flows'):
    with open('flows/' + f, 'r', encoding='utf-8') as reader:
        flow = json.load(reader)
        if len(flow['flowProperties']) > 1:
            results.append(flow['name'])
```

Data pipelines ...

- We use it as an intermediate format to build databases ...
- <https://nexus.openlca.org/databases>



iomb - Input-Output Model Builder

- <https://github.com/USEPA/IO-Model-Builder>

Exporting models

Finally, a model can be exported into a JSON-LD package which can be imported into [openLCA](#):

```
In [19]: import iomb.olca as olca
         olca.Export(eeio_model).to('example_jsonld.zip')
```

openLCA Collaboration server

Workgroup_Displays/AU_Optronics - Data sets

Version Latest [Download Raw Data](#)

Adam A. (AU Optronics) on 08/29/2016 11:49:20

Manufacturing of display (screen) AU Optronics

AU Optronics Display

Display is manufactured in Taiwan by AU Optronics

Inputs/Outputs Documentation Allocation Social aspects Parameters

Flow	Category	Amount	Costs	Uncertainty	Prov
Fe Manufacture of glass and glass products	Commodities / South Korea	0.0734 USD		No distribution	
Fe Manufacture of plastic products	Commodities / China	0.359098096 USD		No distribution	
Fe Metal Products	Commodities / China	0.600849815 USD		No distribution	

Flow	Category	Amount	Costs	Uncertainty	Avol proc
Fe Manufactured	Notebook Production /	1 Item(s)		No distribution	

Version 00.00.015
Last change

Diff: biomass/fuels/Ethanol, 85%, at blending terminal, 2022

Local model	Remote model
<p>Name: Ethanol, 85%, at blending terminal, 2022 Description: transport of gasoline is accounted by using the ecoinver</p> <p>Category: Ethanol Process type: Unit process Location: RNA Infrastructure process: No Process documentation</p> <p>Inputs</p> <ul style="list-style-type: none">Fe 1: Energy, output, from gasolineFe 2: Ethanol, denatured, at refueling station, 2022Fe 3: CUTOFF Liquid storage tank, chemicals, organicsFe 4: Electricity, at grid, US, 2008Fe 5: Gasoline, at refinery6: <p>Outputs</p>	<p>Name: Ethanol, 85%, at blending terminal, 2022 Description: transport of gasoline is accounted by using the ecoinver</p> <p>Category: fuels Process type: Unit process Location: RNA Infrastructure process: No Process documentation</p> <p>Inputs</p> <ul style="list-style-type: none">Fe 1: Energy, output, from gasolineFe 2: Ethanol, denatured, at refueling station, 2022Fe 3: Dummy_liquid storage tank, chemicals, organicsFe 4: Electricity, at grid, US, 20085:Fe 6: Gasoline, at refinery <p>Outputs</p>

Mark as merged

Conclusions

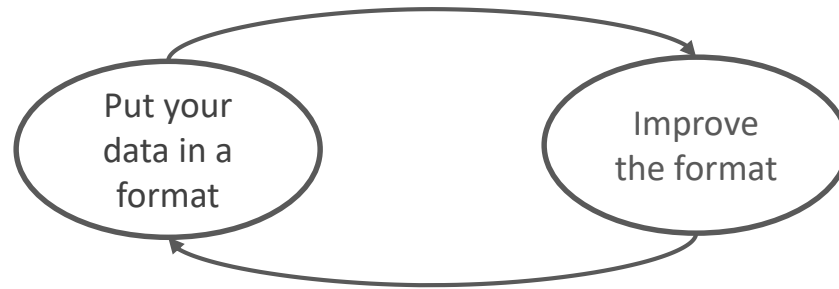
It could be further simplified ...

```
{
  "@type": "Exchange",
  "avoidedProduct": false,
  "input": false,
  "amount": 1.0,
  "flow": {
    "@type": "Flow",
    "@id": "56ced643-75c7-463b-aa9c-1b32cfdd58d0",
    "name": "Steel",
    "flowType": "PRODUCT_FLOW"
  },
  "unit": {
    "@type": "Unit",
    "@id": "20aad24-a391-41cf-b340-3e4529f44bde",
    "name": "kg"
  },
  "flowProperty": {
    "@type": "FlowProperty",
    "@id": "93a60a56-a3c8-11da-a746-0800200b9a66",
    "name": "Mass"
  },
  "@id": "f60111a5-2f68-3740-a780-65ba8fd98d80",
  "quantitativeReference": true
}
```



```
{
  "@type": "Exchange",
  "@id": "Steel_production/steel_output",
  "direction": "output",
  "amount": 1.0,
  "flow": "Steel",
  "unit": "kg"
}
```

It could be further simplified ...



Vocabularies and ontologies can be linked flexibly ...

```
"@context": {  
  "@base": "http://mydatabase.net/",  
  "eco": "http://ontology.earthster.org/eco/core#",  
  "flow": {"@id": "eco:hasFlowable", "@type": "@id"},  
  "unit": {"@id": "eco:hasUnitOfMeasure", "@type": "@id"}  
},
```

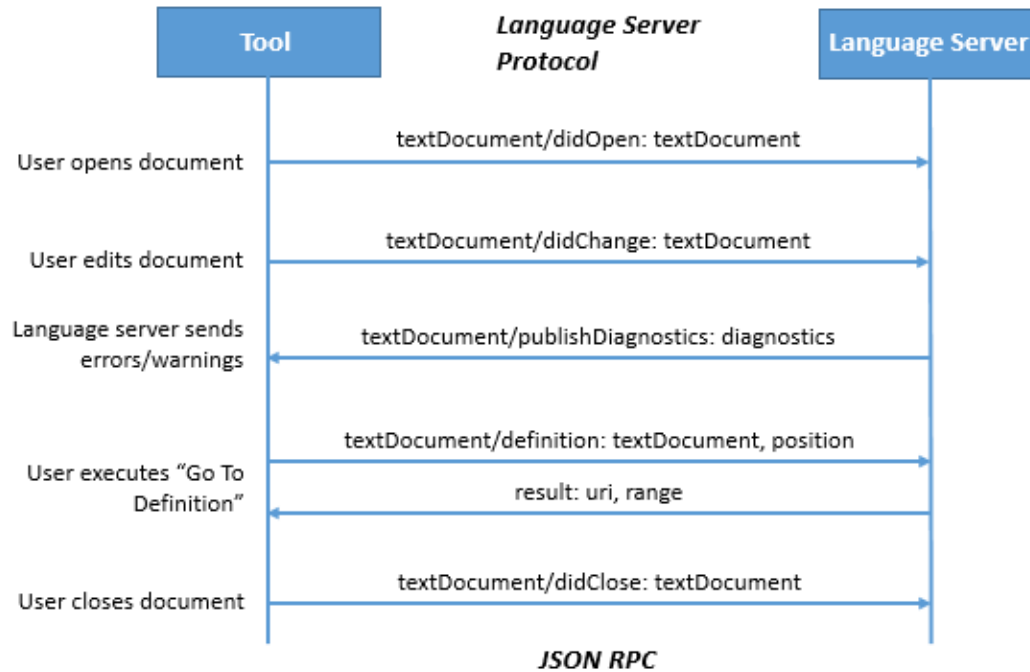
The power of plain text ...

```
steel_production.json (HEAD) → steel_production.json - Id_example - Visual Studio Code
File Edit Selection View Go Help

EXPLORER
├─ OPEN EDITORS
│   └─ steel_production.json (HEAD) → ...
├─ LD_EXAMPLE
│   ├── categories
│   ├── flow_properties
│   ├── flows
│   │   └─ steel.json
│   ├── processes
│   │   └─ steel_production.json
│   ├── unit_groups
│   ├── context.json
│   └─ script.py
└─ master* 0 0 0

steel_production.json (HEAD) → steel_production.json x
11  "processDocumentation": {
12      "@type": "ProcessDocumentation",
13      "copyright": false,
14      "creationDate": "2017-03-27T10:20:18.563",
15  },
16  "exchanges": [
17      {
18          "@type": "Exchange",
19 -      "avoidedProduct": false,
20          "input": false,
21  //
22          "amount": 1.0,
23          "flow": {
24              "@type": "Flow",
25              "@id": "01eb1598-129c-42eb-856c-...",
26              "name": "Steel",
27              "flowType": "PRODUCT_FLOW"
28          },
29          "unit": {
30              "@type": "Unit",
31              "@id": "20aad24-a391-41cf-b340-...",
32              "name": "kg"
33          },
34          "flowProperty": {
35              "@type": "FlowProperty",
36              "@id": "93a60a56-a3c8-11da-a746-..."
37          }
38      }
39  ]
40  }
41  }
```

The power of plain text ...



<https://github.com/Microsoft/language-server-protocol>

The power of plain text ...

The screenshot shows the GitHub interface for the repository `smartchicago/chicago-atlas`. At the top, there are navigation links for Features, Business, Explore, and Pricing, along with a search bar and links to Sign in or Sign up. The repository name is displayed as `smartchicago / chicago-atlas`. Below this, there are statistics for Watch (33), Star (136), and Fork (243). The main navigation bar includes Code, Issues (34), Pull requests (2), Projects (0), Pulse, and Graphs. The current file path is `Branch: master -> chicago-atlas / db / import / zipcodes.geojson`. A commit message from Derek Eder is shown: "Derek Eder merged 60633 in to 60827, closes #175" with a commit hash of 9b4a699 and a date of May 1, 2014. The file details show 7182 lines (7182 sloc) and 248 KB. The map view shows a map of Chicago and surrounding areas with zip codes highlighted in orange. The map includes labels for Elgin, Schaumburg, Bartlett, Saint Charles, Batavia, Aurora, Naperville, Lombard, Elk Grove Village, Cicero, Evanston, and Chicago. The map is powered by Mapbox and OpenStreetMap.

GreenDelta

sustainability consulting + software

Thank you!

Contact

Michael Srocka

srocka@greendelta.com

GreenDelta GmbH

Muellerstrasse 135, 13349 Berlin

www.greendelta.com