

ecoinvent v.3.2 in openLCA



Software version: openLCA 1.4.2

Report version: 2

Date: 1 March 2016

Author: Cristina Rodríguez, Franziska Möller, Andreas Ciroth

GreenDelta GmbH, Müllerstrasse 135, 13349 Berlin, Germany; gd@greendelta.com

GreenDELTA

Content

1	ecoinvent v.3.2 – What’s new?	3
2	ecoinvent v.3.2 in openLCA	3
2.1	Renaming of datasets	4
2.1.1	Activities	5
2.1.2	Intermediate exchanges.....	6
2.2	Update of UUIDs.....	8
2.2.1	Intermediate exchanges.....	8
2.2.2	Activities	9
2.3	Refactoring of categories	9
2.4	Mapping of elementary flows.....	10
2.5	ecoinvent LCIA methods.....	12
3	Converting own datasets from ecoinvent v.3.1 to ecoinvent v.3.2.....	12
4	Quality assurance and calculation times	14
5	Known issues, limitations.....	16
5.1	Database.....	16
5.2	openLCA.....	17
5.2.1	Preserving the product system layout after importing a *.zolca file containing product systems	17
5.2.2	Flow properties and descriptions specific of exchanges.....	18
6	Summary.....	19

1 ecoinvent v.3.2 – What’s new?

The ecoinvent v.3.2 database was released on the 30th November 2015 and included important updates and additions, such as:

- **Complete update of the electricity sector:** the newest grid mixes and the latest available data on electricity production were used in over 2,200 electricity datasets, covering 89% of the world's electricity production in 56 countries. Moreover, the annual production volumes for all technologies were also updated and new technologies are now also included.
- **New electricity data for 32 Chinese Regions:** the new version includes electricity production data for 32 Chinese regions, as well as mixes for different grids within China and for China as a whole.
- **New refrigerated transport data** for all different modes of transport (i.e. ship, train, aircraft and truck).
- **Other new international data** in the areas of the cement, wood, aluminium production and waste treatment.
- **Market groups:** this is a new type of dataset added to facilitate the readability and modification of datasets.

For more information about the content and methodology of ecoinvent v.3.2 database, please check the [report of changes](#) or the [ecoinvent website](#).

2 ecoinvent v.3.2 in openLCA

As in previous ecoinvent 3 versions, six data packs generated by the ecoinvent Centre from the undefined ecoinvent database are provided containing the three different system models, all as unit and aggregated processes:

- Allocation, allocation at the point of substitution¹ (unit and system² processes)
- Allocation, cut-off by classification (unit and system processes)
- Consequential long-term (unit and system processes)

In addition, access to older versions of the ecoinvent database is also provided when purchasing ecoinvent v.3.2 in [openLCA Nexus](#).

¹ In previous ecoinvent 3.1 versions, this system model was named “allocation, default”

² Named “LCI” in the data files

All the different data packs can be used as independent databases in openLCA or combined together, if necessary³. Moreover, in order to be able to reproduce the LCIA results reported by theecoinvent Centre, a pack with the LCIA methods as implemented by them is also available for download in openLCA Nexus.

In this new version of ecoinvent in openLCA, different improvements were carried out to make it more user-friendly. For instance, the names of the activities and products were extended to simplify the modelling of new process and results analysis. As in v.3.1, the identifiers of the flows and processes were also updated to allow the combination of different system models and process types in a single openLCA database. What is new in v.3.2 is that the identifiers were also harmonised to those in v.3.1 so that the conversion from one version to the other is much simpler now.

Further details about the implementation work done and the functionalities of the new ecoinvent v.3.2 in openLCA are explained in the next sections of this report.

2.1 Renaming of datasets

Some openLCA users pointed out for previous ecoinvent 3 versions that the fact that several processes existed with the same name, location and category but producing different reference products was a limitation for a fast modelling and results analysis. This issue was caused because when applying the different system models in ecoinvent 3, a single undefined activity is split into several new activities producing each of the co-products. However, the name of the original activity is maintained in all of them.









- ▲  0144:Raising of sheep and goats
 -  market for sheep fleece in the grease, alloc. default, S - GLO
 -  market for sheep for slaughtering, live weight, alloc. default, S - GLO
 -  sheep for slaughtering, live weight to generic market for red meat, live weight, alloc. default, S - GLO
 -  sheep production, for meat, alloc. default, S - RoW
 -  sheep production, for meat, alloc. default, S - RoW
 -  sheep production, for meat, alloc. default, S - US
 -  sheep production, for meat, alloc. default, S - US

Figure 1: Example of ecoinvent v.3.1, allocation default activities with the same name, location and category in openLCA v.1.4.2

³ It is recommended to always import the data pack of smaller size into the bigger one to reduce the time of the import (e.g. unit process data files into LCI data files).

Likewise, there were also several product flows with the same name, location and category but produced by several processes. This was caused by the current limitation in openLCA that a product flow cannot be used multiple times in the same process with different providers⁴. As this is the case in most of the non-aggregated activities of ecoinvent 3 (e.g. several electricity flows provided by different electricity markets), it was not possible to have a single product flow produced by multiple activities in the database in openLCA. In version 3.1 of June 2015, this issue was partially overcome by adding in the flow's description the name of the activity producing it in order to make the database more user-friendly. This information is shown in the status bar of openLCA when clicking on the product's name in the navigation pane, and also in the tooltip shown when hovering over the flow's name in the "add new flow" window of the process editor. However, we wanted to provide an even more user-friendly ecoinvent 3 database in openLCA.

Therefore, in the new ecoinvent v.3.2 databases, the names of the processes and products have been extended to allow a straightforward identification of each activity and intermediate exchange without the need of opening the correspondent dataset or checking their description information in the status bar or tooltips.

2.1.1 Activities

The name of the activities in openLCA follows this schema:

Activity name Reference product name System model, Process type

The nomenclature used for the system model information is:

- **APOS:** Allocation, allocation at the point of substitution
- **conseq. long-term:** consequential long-term
- **cut-off:** Allocation, cut-off by classification

The process types are indicated with "U" for unit processes (i.e. non-aggregated datasets) or "S" for system processes (i.e. LCI results).

⁴ It is planned to solve this limitation in future releases of openLCA

- 014:Animal production
 - 0141:Raising of cattle and buffaloes
 - cattle for slaughtering, live weight to generic market for red meat, live weight | red meat, live weight | APOS, U - GLO
 - market for cattle for slaughtering, live weight | cattle for slaughtering, live weight | APOS, U - GLO
 - market for cow milk | cow milk | APOS, U - GLO
 - market for operation, housing system, cattle, loose, per animal unit | operation, housing system, cattle, loose, per animal unit | APOS, U - GLO
 - market for operation, housing system, cattle, tied, per animal unit | operation, housing system, cattle, tied, per animal unit | APOS, U - GLO
 - market for red meat, live weight | red meat, live weight | APOS, U - GLO
 - milk production, from cow | cattle for slaughtering, live weight | APOS, U - CA-QC
 - milk production, from cow | cattle for slaughtering, live weight | APOS, U - RoW
 - milk production, from cow | cow milk | APOS, U - CA-QC
 - milk production, from cow | cow milk | APOS, U - RoW

Figure 2: Example of ecoinvent v.3.2, allocation at the point of substitution activities in openLCA v.1.4.2

The information of the system model and the process type was added because users can combine different data packs together in a single openLCA database (e.g. consequential long-term unit and system processes; three system models together; etc.). More information is provided in [section 2.2](#).

Note: In some cases, it was necessary to truncate the name of the reference product due to the limitation of 255 characters per name.

2.1.2 Intermediate exchanges

The name of the intermediate exchanges (i.e. product flows) in openLCA follows this schema:

Flow name | Producer name

- 3510a: Electric power generation based on liquid fuels
 - diesel, burned in diesel-electric generating set, 10MW | diesel, burned in diesel-electric generating set, 10MW - GLO
 - diesel, burned in diesel-electric generating set, 10MW | market for diesel, burned in diesel-electric generating set, 10MW - GLO
 - diesel, burned in diesel-electric generating set, 18.5kW | diesel, burned in diesel-electric generating set, 18.5kW - GLO
 - diesel, burned in diesel-electric generating set, 18.5kW | market for diesel, burned in diesel-electric generating set, 18.5kW - GLO
 - electricity, high voltage | electricity production, oil - ASCC
 - electricity, high voltage | electricity production, oil - AT
 - electricity, high voltage | electricity production, oil - AU
 - electricity, high voltage | electricity production, oil - BA
 - electricity, high voltage | electricity production, oil - BE

Figure 3: Example of ecoinvent v.3.2, consequential long-term intermediate exchanges in openLCA v.1.4.2

There is no system model or process type information included in the product flow's name because the product flows are harmonised between system models and process types. Check [section 2.2](#) for further clarifications.

Developer box 1: Rewriting activities and intermediate exchanges to their originalecoinvent names

In case you were interested in using the original ecoinvent names, that is, without the additional information described in the sections above, you can use the following Python script:

- **Rewriting activities and intermediate exchanges to their originalecoinvent names:**

```
import org.openlca.io.KeyGen as keygen

def change_process_name(process):
    old_name = process.name
    i = old_name.find('|')
    new_name = old_name[0:i-1]
    process.name = new_name
    olca.updateProcess(process)

def change_flow_name(flow):
    if flow.flowType.name()=='PRODUCT_FLOW':
        old_flow_name = flow.name
        j = old_flow_name.find('|')
        new_flow_name = old_flow_name[0:j-1]
        flow.name = new_flow_name
        olca.updateFlow(flow)

olca.eachProcess(change_process_name)
olca.eachFlow(change_flow_name)
```

Note: This code will delete also the information regarding the system model and process type (i.e. U, S). If you want to add later that information, you can use the following Python script:

- **Adding system model and process type information to originalecoinvent activity names:**

```
import org.openlca.io.KeyGen as keygen

# define the system model and process type here
system_model = 'conseq. long-term'
process_type = 'U'

def change_name(process):
    old_name = process.name
    new_name = '%s, %s, %s' % (old_name, system_model, process_type)
    process.name = new_name
    olca.updateProcess(process)

olca.eachProcess(change_name)
```

If you want to have the additional information in the description fields like in version 3.1, you can use the next script to include the reference product name in the activity's description and the producing activity name in the intermediate exchange's description. Please, be aware that the provider names will be added as they are in that moment in the database; thus, if they contain also e.g. the system model and flow type, this will be shown in the product flow's description. Consequently, it is recommended to run this script before adding the system model and process type details to the process name.

- Add reference product and provider information to the activity and product descriptions, respectively:

```
import org.openlca.io.KeyGen as keygen

def extend_descriptions(process):
    process_name = process.name
    process_location_code=process.location.code
    e = process.quantitativeReference
    flow_description = e.flow.description
    flow_name = e.flow.name
    flow_location_code = e.flow.location.code
    new_flow_description = 'Produced by: "%s - %s". %s' %(process_name,
process_location_code, flow_description)
    e.flow.description = new_flow_description
    olca.updateFlow(e.flow)
    process_description = process.description
    new_process_description = 'Reference product: "%s - %s". %s' %(flow_name,
flow_location_code, process_description)
    process.description = new_process_description
    olca.updateProcess(process)

olca.eachProcess(extend_descriptions)
```

2.2 Update of UUIDs

During the implementation of ecoinvent v.3.1 the possibility of combining different system models and process types within a single openLCA database was already included, as explained in [this video](#). This feature has been maintained and extended in openLCA's ecoinvent v.3.2 so that users can easier convert their processes with ecoinvent v.3.1 exchanges to ecoinvent v.3.2.

2.2.1 Intermediate exchanges

The Universally Unique Identifier (UUID) of product flows existing in different system models were harmonised so that the users can easily change from one system model to the other, as well as combine both of them in a single openLCA database. A product flow was considered equal if the flow name, producer name, location and reference flow property were the same. Moreover, the identifiers of the product flows were also harmonised between unit and system processes, so that the users can decide in each case which type of process to use in the supply chains without the necessity of modifying their process datasets.

Flow	Category	Flow property	Unit	Amount	Uncertainty	Default provider	Pedigree uncertainty
⚡ electricity, high voltage electricity pr...	351:Electric power generatio...	Energy	MJ	1.0	none	None	
						None	
						electricity production, hard coal electricity, high voltage cut-off, U - AT	
						electricity production, hard coal electricity, high voltage APOS, U - AT	

Figure 4. Example of selection of default provider for an input when different system models are included in a database

In addition, the products UUIDs were mapped to the identifiers used in ecoinvent v.3.1 following the [correspondence file](#) provided by the ecoinvent Centre, so that the user can easily convert their own processes with ecoinvent exchanges from v.3.1 to v.3.2. For more information about the conversion from one version to the other, check [section 3](#).

2.2.2 Activities

The UUIDs of the activities in openLCA were changed to be different between system models and process types. This is necessary because if a process with the same identifier exists already in the database, it will not be added during the import. Thus, for being able to combine system models in a single openLCA database, the process UUIDs are now unique between data packs.

Moreover, the UUIDs were mapped to the identifiers used in ecoinvent v.3.1, as explained for the intermediate exchanges. This implies that it is not possible to have ecoinvent v.3.1 and v.3.2 processes in the same openLCA database.

Developer box 2: Making the process UUIDs unique in order to combine ecoinvent v.3.1 and v.3.2 in the same openLCA database

In case you were interested in using ecoinvent v.3.1 and v.3.2 datasets together in an openLCA database, it would be necessary to make the process identifiers unique. For that, you can run the following Python script in the databases you want to combine:

```
import org.openlca.io.KeyGen as keygen

# define the database version here
# these values are used to generate the new UUID
ecoinvent_version = '3.1'

def change_refid(process):
    old_refid = process.refId
    new_refid = keygen.get(old_refid, ecoinvent_version)
    process.refId = new_refid
    olca.updateProcess(process)

olca.eachProcess(change_refid)
```

Note: Once you have changed the process UUIDs, it will not be possible to automatically convert default providers and product systems from v.3.1 to v.3.2 as explained in section 3.

2.3 Refactoring of categories

The activities and product flows are categorized following the correspondent sectors IDs as included in ecoinvent's metadata, which allows a very easy navigation through the database, as shown in Figure 5.

Moreover, in the allocation system models, there were activities and their correspondent reference products without any category stated. Most of them were electricity activities, which are now placed in the general category “D:Electricity, gas, steam and air conditioning supply”. In the cut-off system model, there were also activities without category defined as “Recycled Content cut-off” processes. For those, a new category “Recycled content cut-off” was created.

- ▾ 📁 Processes
 - ▾ 📁 A:Agriculture, forestry and fishing
 - 📁 01:Crop and animal production, hunting and related service activities
 - ▾ 📁 02:Forestry and logging
 - ▾ 📁 021:Silviculture and other forestry activities
 - 📁 0210:Silviculture and other forestry activities
 - 📁 022:Logging
 - 📁 023:Gathering of non-wood forest products
 - 📁 B:Mining and quarrying
 - 📁 C:Manufacturing
 - 📁 D:Electricity, gas, steam and air conditioning supply
 - 📁 E:Water supply; sewerage, waste management and remediation activities
 - 📁 F:Construction
 - 📁 G:Wholesale and retail trade; repair of motor vehicles and motorcycles
 - 📁 H:Transportation and storage
 - 📁 J:Information and communication
 - 📁 L:Real estate activities
 - 📁 M:Professional, scientific and technical activities
 - 📁 N:Administrative and support service activities
 - 📁 Recycled Content cut-off
 - 📁 S:Other service activities

Figure 5: Overview of some process categories in ecoinvent v.3.2, allocation, cut-off by classification in openLCA v.1.4.2

2.4 Mapping of elementary flows

As with almost all the databases provided in openLCA Nexus, the elementary flows used in the ecoinvent v.3.2 database were mapped to openLCA reference data. This allows more consistency in the elementary flows when combining different inventory databases from Nexus, as well as the possibility of using the [openLCA LCIA methods pack](#).

In case no match could be found, the unmapped elementary flows were added to the openLCA reference list. However, it should be noted that the openLCA LCIA methods pack has not yet been updated to include them. The new elementary flows are listed in Table 1.

If you are interested in having the ecoinvent v.3.2 data pack provided in openLCA Nexus without openLCA reference data, please contact [us](#).

Table 1. New elementary flows added to openLCA reference list

Elementary flow name	Compartment	Sub-compartment(s)
2,4-D dimethylamine salt	air	low population density
	soil	agricultural
	water	ground water, surface water
2,4-D polypropoxybutyl ester	air	low population density
	soil	agricultural
	water	ground water, surface water
2,4-DB	air	low population density
	soil	agricultural
	water	ground water, surface water
Atrazine	water	surface water
Bentazone	water	surface water
Bromoxynil	water	surface water
Carbaryl	water	surface water
Carbon dioxide, non-fossil, from calcination	air	unspecified
Cobalt, Co 5.0E-2%, in mixed ore, in ground	resource	in ground
Copper, Cu 6.8E-1%, in mixed ore, in ground	resource	in ground
Dicamba	water	surface water
Dichlorprop	air	low population density
	soil	agricultural
	water	ground water, surface water
Dimethenamid	water	surface water
Ethephon	water	surface water
Glyphosate	water	surface water
Gold, Au 1.0E-7%, in mixed ore, in ground	resource	in ground
Herbicides, unspecified	soil	agricultural
Lambda-cyhalothrin	water	surface water
Mandipropamid	soil	agricultural
MCPA	water	surface water
MCPB	water	surface water
Methomyl	water	surface water
Metolachlor	water	surface water
Nickel, Ni 2.5E+0%, in mixed ore, in ground	resource	in ground
Palladium, Pd 1.6E-6%, in mixed ore, in ground	resource	in ground
Pendimethalin	water	surface water
Platinum, Pt 4.7E-7%, in mixed ore, in ground	resource	in ground
Prothioconazol	air	low population density
	water	ground water, surface water
Pyraclostrobin	water	ground water, surface water
Rhodium, Rh 1.6E-7%, in mixed ore, in ground	resource	in ground
Silver, Ag 1.8E-6%, in mixed ore, in ground	resource	in ground
Tefluthrin	water	surface water
Trifloxystrobin	water	ground water, surface water

2.5 ecoinvent LCIA methods

The ecoinvent LCIA methods packs provided in openLCA Nexus are intended to be used ONLY with the correspondent ecoinvent database in order to reproduce the LCIA results reported by the ecoinvent Centre. **They should not be used with other databases available in Nexus as not all the openLCA elementary flows are characterised in them**, only those used by the ecoinvent databases.

In the pack for version 3.2, the methods IPCC 2007 and 2013 were updated, as explained in the [Report of Changes](#) provided by the ecoinvent Centre in their website. It should be noted that ecoinvent recommends not to use the IPCC 2013 results reported for version 3.1.

For more information regarding the implementation of LCIA methods by the ecoinvent Centre, please check:

Hischier R., Weidema B., Althaus H.-J., Bauer C., Doka G., Dones R., Frischknecht R., Hellweg S., Humbert S., Jungbluth N., Köllner T., Loerincik Y., Margni M., and Nemecek T. (2010) [Implementation of Life Cycle Impact Assessment Methods](#). Final report ecoinvent v2.2 No. 3. Swiss Centre for Life Cycle Inventories, Dübendorf, CH.

3 Converting own datasets from ecoinvent v.3.1 to ecoinvent v.3.2

As mentioned previously in this report, the ecoinvent v.3.2 database has been implemented in openLCA in a way that allows the users to easily convert their processes from ecoinvent v.3.1 to v.3.2. For the conversion, the **openLCA database/zolca-File import** option is recommended. The procedure would be:

1. Download from openLCA Nexus the ecoinvent v.3.2 data pack you are interested in (e.g. ecoinvent 3.2 – consequential long-term LCI) and import the file into openLCA.
2. Import the old openLCA database containing your own processes into the new ecoinvent 3.2 database (i.e. File→Import→Database import→Existing database). It is also possible to first export the old database as *.zolca (i.e. right click on the database name → Export database) and then import it into the new database (i.e. File→Import→Database import→From exported zolca-File).

During the import, as the products and process UUIDs were previously harmonised between ecoinvent versions, those processes and flows existing in both versions will be mapped to v.3.2.

The [correspondence file](#) provided by the ecoinvent Centre was used for the mapping. However, it should be noted that there are some activities/products that could not be included in the automatic mapping during the import. Consequently, they will be imported to the new database, and further conversion work would be needed. Those are:

- Activities/products which have been deleted in v.3.2
- Activities/products with “amount of the replacement”⁵ different than 1
- Activities/products which should be replaced by other activities/products already existing in v.3.1. For example, “aluminium production, primary, liquid, prebake – RoW” should be mapped to “aluminium production, primary, liquid, prebake – GLO”. However, this activity exists in v.3.1 and it is mapped to the correspondent “aluminium production, primary, liquid, prebake – GLO” in v.3.2. As it is not possible to have two UUIDs specified for the same activity, the “aluminium production, primary, liquid, prebake – RoW” UUID cannot be used in the v.3.2 database.
- Activities/products from different locations mapped to a single v.3.2 global activity/product. In those cases, only one of the activities/products could be mapped to the v.3.2 global dataset. This is the case for “market for sawnwood, hardwood, raw | sawnwood, hardwood, raw”, in which the UUID of the v.3.1 European activity has been used for the global activity in v.3.2. Thus, the Rest-of-World activity remains unmapped.

If your database contains also product systems or projects, these will also be imported.

However, check [section 5.2](#) for solving issues in the display of the product systems graphs.

It is also possible to export the processes or product systems, for example, as **ILCD format** and import them into the new database. The flows and processes existing in both databases will be mapped to the ecoinvent v.3.2 datasets. However, due to limitations in the format, the information regarding the default providers in the process datasets will be lost. Moreover, as commented for the *.zolca import, if a product flow used in the process dataset is not included in v.3.2 or there is not a direct match to the new product flows, there will not be a provider for

⁵ As explained by ecoinvent, if the cell “amount of replacement” in the correspondence file has a value of 1 it means the ecoinvent v3.1 activity is replaced by exactly one activity in ecoinvent v3.2. If this value is different (below one) it represent the share of the replacement activity. For example, if the value is 0.25, it means that 25% of the ecoinvent v3.1 activity should be replaced by the specific activity in ecoinvent v3.2. The share of the ecoinvent v3.1 matches of course sum up to 1.

that flow unless the correspondent ecoinvent v.3.1 producing process is also included in the ILCD file imported.

If you were interested in GreenDelta carrying out the complete conversion of your datasets or assist you with the conversion, please contact [us](#).

4 Quality assurance and calculation times

The inventory and impact assessment results calculated for product systems using unit process datasets in openLCA were compared to the LCI and LCIA results published by the ecoinvent Centre. As it can be observed in the example plotted in Figure 6, the LCI results obtained in openLCA were almost equal⁶ to the ecoinvent system processes, with the exception of the social flows in the consequential model; however, as explained in [section 5.1](#), ecoinvent recommends to ignore those flows in the current database.

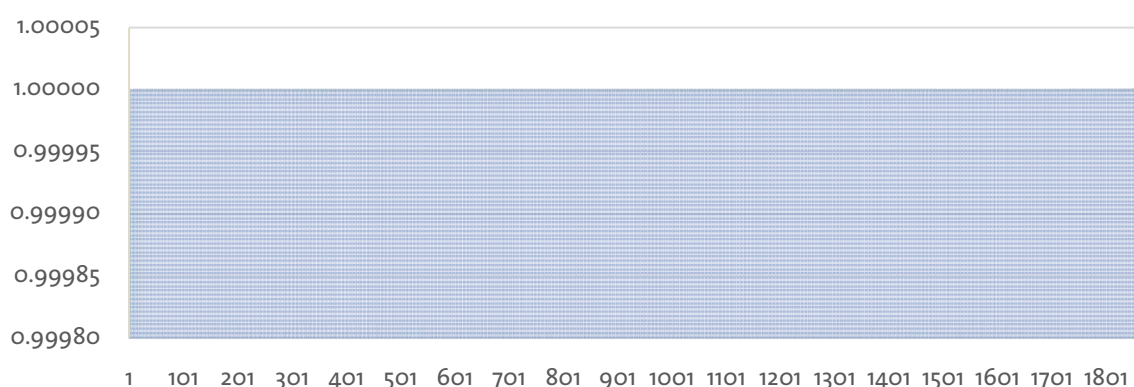


Figure 6. Ratio unit process LCI results to LCI process for "electricity, high voltage, production mix | electricity, high voltage | APOS - DE", ecoinvent v.3.2, openLCA v.1.4.2 (i.e. ratio of 1 represents equal inventory values)

Moreover, LCIA results calculated using the ecoinvent LCIA methods pack were also checked resulting in almost equal⁶ results as those reported in the ecoinvent website (Figure 7).

The times required for creating product systems using unit processes and calculating LCI and LCIA results were also measured. The product systems were created very quickly in less than 10 seconds in all models⁷. The "Quick results" calculation time varied between around 10 and 25 seconds depending on the system model used due to the differences in number of processes and links per product system. The time necessary for the "Analysis" calculation was a bit

⁶ Slight differences might be caused by rounding during the calculation

⁷ Times might increase for the first product system created after opening the openLCA database

higher but still very fast, around 30 seconds for the consequential model and less than 80 seconds for the allocation models. Figure 8 shows some examples of the times measured.

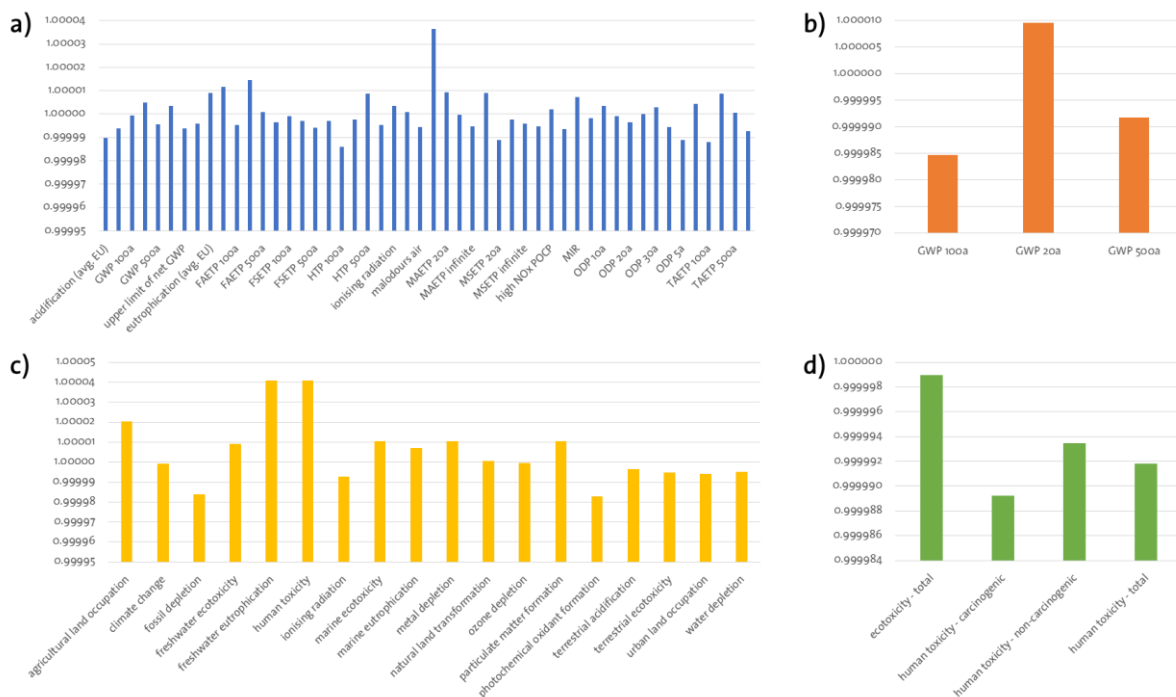


Figure 7. Ratio openLCA LCIA results (with ecoinvent methods) to ecoinvent v.3.2 LCIA reported results (i.e. ratio of 1 represents equal LCIA results). a) market for coating powder | coating powder | conseq. long-term - GLO; CML 2001. b) market for transport, passenger car, small size, diesel, EURO 5 | transport, passenger car, small size, diesel, EURO 5 | cut-off - GLO; IPCC 2007. c) iron mine operation, crude ore, 46% Fe | iron ore, crude ore, 46% Fe | APOS - GLO; ReCiPe Midpoint (H). d) furnace production, wood chips, with silo, 5000kW | furnace, wood chips, with silo, 5000kW | APOS - CH; UseTox

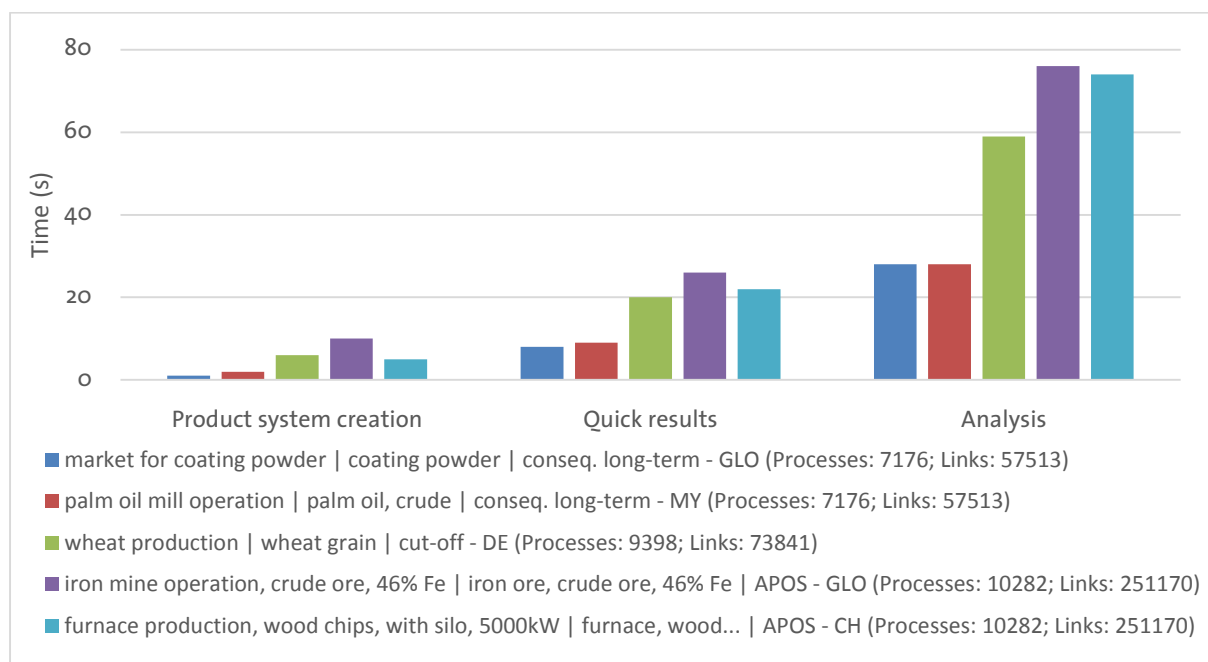


Figure 8. Example of product system creation times and calculation times for five product systems of ecoinvent v.3.2

Note: If the amount of memory allocated to openLCA is too low, it might lead to errors during the calculation due to the high amount of processes and links per product system. If that occurs, you can increase the memory usage in: File→ Preferences→ Configuration→ Maximum memory usage.

5 Known issues, limitations

5.1 Database

- The **input exchanges listed in Table 2 were deleted from the consequential long-term system model** following indications from the ecoinvent Centre.

Table 2. Deleted input exchanges in the consequential long-term system model

Intermediate exchange	Activity	Location	Amount	Unit
residual hardwood, wet	market for residual hardwood, wet	GLO	1	m ³
residual softwood, wet	market for residual softwood, wet	GLO	1	m ³
residual wood, dry	market for residual wood, dry	GLO	1	m ³
venting of nitrogen, liquid	market for venting of nitrogen, liquid	GLO	1	kg

This was necessary due to an error in the original EcoSpold2 files, where the default providers indicated for the exchanges were activities not producing those product flows. This issue will be fixed by the ecoinvent Centre in next versions of the ecoinvent database.

- According to ecoinvent, the following **social flows included in the consequential, long-term system model should be ignored**: “residual hardwood, wet”, “residual softwood, wet”, “residual wood, dry”, “venting of argon, crude, liquid” and “venting of nitrogen, liquid”. The inventory results for these flows obtained using product systems with unit processes will not match the LCI results provided by ecoinvent.
- The **markets for “heat, from municipal waste incineration/CA-” are missing**. Consequently, the v.3.1 activities “heat, from municipal waste incineration to generic market for heat district or industrial, other than natural gas | heat, for reuse in municipal waste incineration only” in locations CA-AB, CA-NB, CA-NS, CA-ON and CA-PE have no direct match in the consequential model of v.3.2. This is not intended and will be corrected by ecoinvent in future versions of the database.

- **No mathematical relations (i.e. parameters) are included** in the openLCA data packs because they refer to the undefined activities, that is before allocation, and may not be reflecting the amounts in the dataset any more. If you are interested in checking them anyway, you can review the EcoSpold2 files available in the ecoinvent website⁸.

Other [known issues of the ecoinvent v.3.2](#) database are listed in the ecoinvent website.

5.2 openLCA

5.2.1 Preserving the product system layout after importing a *.zolca file containing product systems

The model graph of a product system existing in a database to which a *.zolca file or “openLCA existing database” containing also product systems is imported cannot be displayed properly without deleting the imported “layouts” file from the openLCA-data folder. Moreover, the layout of the product systems contained in the imported file cannot not be preserved anyway after the import. This issue, caused by the use of database internal IDs instead of reference UUIDs in the “layouts” file, will be fixed in future releases of openLCA. However, for the time being, the following approach is suggested when product systems are included in the imported file/database (Figure 9):

1. If there are already product systems in the database in which you are going to import the *.zolca file containing product systems, make a backup of the “layouts” folder before the import. This folder will be located in the correspondent database folder in your openLCA-data directory. By default, this directory is: `C:\Users\User name\openLCA-data-1.4\databases\database name_olca_layouts`
If you do not have any product system created in the target database, or you do not mind if their current layout is lost (i.e. position of processes in the model graph), skip steps 1 and 4.
2. Import the *.zolca file into the database in openLCA.
3. Delete the content of the “layouts” folder of the database where the file was imported.
4. Copy the content of the backup made in step 1 into the “layouts” folder.
5. Open the product system again, the display should work properly now.

⁸ Login information to the ecoinvent website is provided after the ecoinvent license is granted in Nexus

If you are not interested in maintaining the information of the default providers in the process datasets, you can also export the product system as ILCD format.

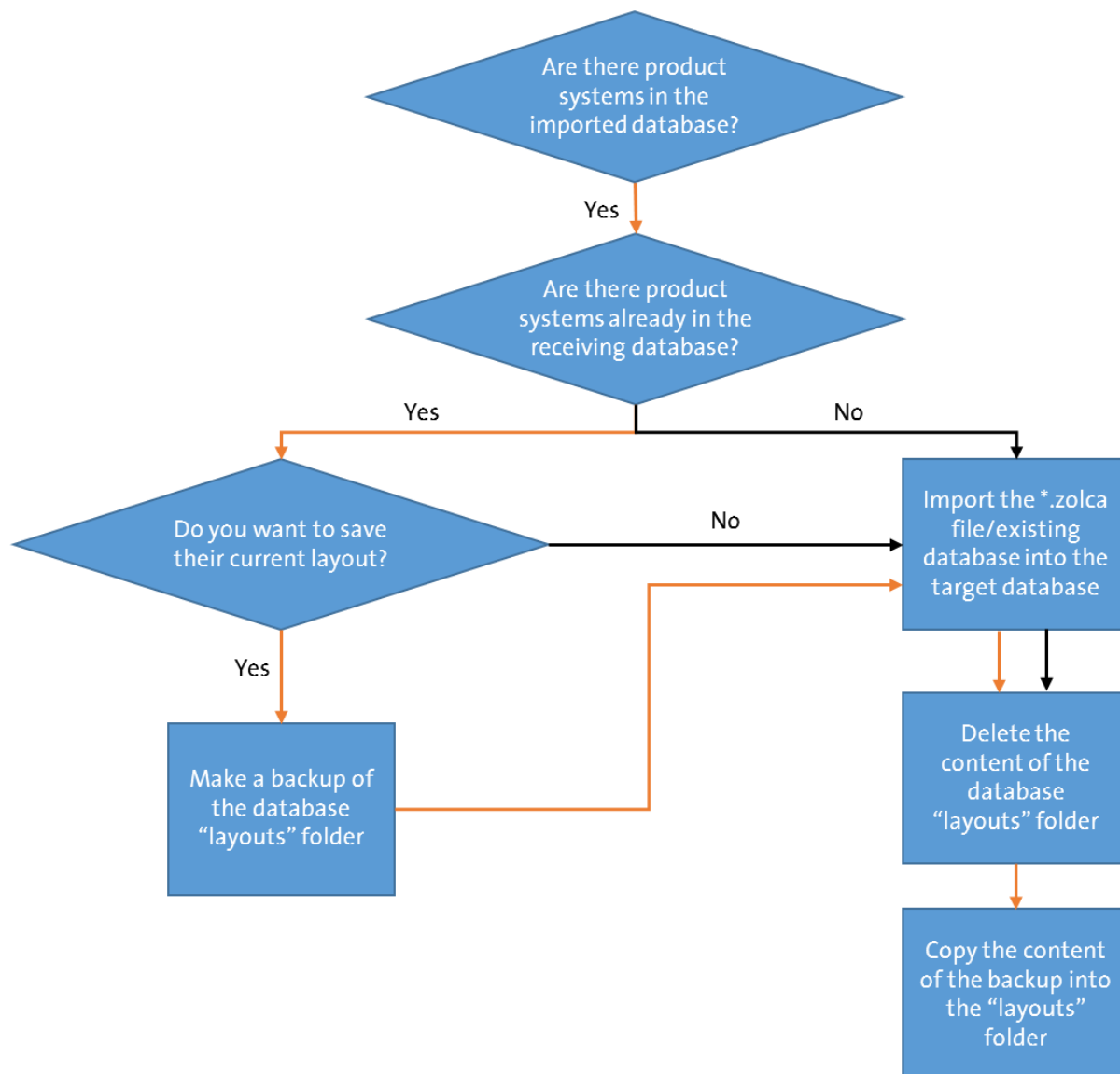


Figure 9. Procedure for solving the issue of the display of model graphs in product systems imported from *.zolca file or existing database in openLCA

5.2.2 Flow properties and descriptions specific of exchanges

In many cases, there are different flow properties specified for the different exchanges which are not flow specific, but exchange specific in the original EcoSpold2 files (e.g. carbon content, water content, price, etc.). Moreover, descriptions for some exchanges are also included in them. Currently, it is not possible to display in openLCA this information, so it is recommended to the users interested in it to review the EcoSpold2 files available for download in the

ecoinvent website. However, it is planned for future releases of openLCA to add a comment field per exchange, in which all this information could be displayed.

6 Summary

The ecoinvent v.3.2 in openLCA contains all the numerous updates and new data from the ecoinvent Centre, as well as the following benefits:

- Access to the three system models as both unit and aggregated processes in separate data files, which can be imported quickly in openLCA.
- Possibility of combining different system models and process types in a single openLCA database.
- Simplification of the conversion of processes using ecoinvent v.3.1 intermediate exchanges to ecoinvent v.3.2.
- More user-friendly naming of activities and product flows.
- Access to the pack of LCIA methods as implemented by the ecoinvent Centre, in order to reproduce their LCIA results.
- Possibility of using the openLCA LCIA methods pack thanks to the mapping of elementary flows to the openLCA reference list.

If you have other questions not addressed by this report, need further clarifications on any of the points commented, or have comments about the ecoinvent v.3.2 database in openLCA, please contact [us](#).