openLCA 1.10

Comprehensive User Manual

Software Version:	1.10.1
Manual Version:	October 2019
Authors:	Dr A. Ciroth, C. Di Noi, T. Lohse, M. Srocka

GreenDelta

Table of Contents

1 Introduction 1
1.1 Introduction to openLCA 1
1.1.1 What is new in version 1.10.1 1
1.2 Introduction to GreenDelta 2
1.3 Introduction to openLCA Nexus 2
2 Installation 3
2.1 Windows 3
2.2 Mac 5
2.3 Linux 6
3 Welcome to openLCA 6
3.1 Main menu functions 7
3.2 Tabs 11
4 Databases 11
4.1 Creating a new, empty database 11
4.2 Restoring a database 13
4.2.1 Accessing databases from openLCA Nexus 13
4.3 Database scheme update 15
4.4 Importing and combining databases 15
4.4.1 Importing a database from exported zolca-File 15
4.4.2 Importing an existing database 17
4.4.3 Importing databases in ecospold, Excel, ILCD, SimaPro CSV, and JSON-LD formats 17
4.5 Creating a remote database 20
4.6 Database elements 21
4.7 Exporting data 22
4.7.1 Exporting data in EcoSpold formats 22
4.7.2 Exporting data in Excel format 24
4.7.3 Exporting data in ILCD ZIP-file format 25
4.7.4 Exporting data as html file 26
4.7.5 Exporting data as CSV-Matrix 27
4.7.6 Export data as JSON-LD file 27
4.7.7 "Copy" function for all openLCA tables 29

4.8 Link with Collaboration Server 30

5 Flows 35

5.1 Creating a new flow 35

5.2 Flow tab contents 35

5.2.1 General information 35

5.2.2 Flow properties 36

5.3 Waste modelling with the actual flow direction 37

6 Processes 39

6.1 Creating a new process 39

6.2 Process tab contents 41

6.2.1 General information 41

6.2.2 General information - Data quality 41

6.2.3 Inputs/Outputs 43

6.2.4 Administrative information 44

6.2.5 Modelling and validation 44

6.2.6 Parameters 45

6.2.7 Allocation 46

6.2.8 Social aspects 49

7 LCIA methods 51

7.1 Importing LCIA methods into openLCA 51

7.2 Creating a new impact assessment method 53

7.3 Impact methods tab contents 53

7.3.1 General information 53

7.3.2 Impact factors 54

7.3.3 Normalization/Weighting 54

7.3.4 Parameters 54

7.3.5 Shape file parameters 55

8 Product systems 56

8.1 Creating a new product system 57

8.2 Product system tabs contents 59

8.2.1 General information 59

8.2.2 Parameters 60

8.2.3 Model graph 60

8.2.4 Statistics 65
8.3 Nested product systems 66
8.4 Calculating a product system 67
9 Result Analysis 70
9.1 Quick results tab contents 70
9.1.1 General information 71
9.1.2 Inventory results 72
9.1.3 Impact analysis 73
9.1.4 LCIA checks 74
9.1.5 Locations 77
9.1.6 Grouping 77
9.2 Analysis tab contents 80
9.2.1 General information 80
9.2.2 Inventory results 80
9.2.3 Impact analysis 80
9.2.4 Process results 80
9.2.5 Contribution tree 81
9.2.6 Grouping 82
9.2.7 Locations 82
9.2.8 Sankey diagram 83
9.2.9 LCIA checks 83
9.3 Regionalized LCA 83
9.3.1 Locations 84
9.3.2 Calculation framework 85
9.3.3 Parameterization of LCIA methods 85
9.3.4 Calculation of regionalized LCA 86
9.4 Monte Carlo Simulation 88
9.4.1 Adding uncertainty information 89
9.4.2 Starting the Monte Carlo Simulation 90
9.4.3 Monte Carlo Results 92
9.5 Life Cycle Costing in OpenLCA 93
9.5.1 Available Data 95
9.5.2 Results and analysis available 95

10 Projects 96
10.1 Creating a new project 96
10.2 Project tab contents 97
10.2.1 Project setup 97
10.2.2 Report sections 100
10.3 Exporting a project report 100
11 Advanced Topics 101
11.1 Expanding memory on a MacOS system 101
11.2 Scripting in openLCA 102
11.2.1 The Python interpreter 105
11.2.2 The JavaScript interpreter 106
11.2.3 The olca-object and the inspection function 106
11.2.4 Modifying database content 108
11.2.5 Running Calculations 108
11.2.6 Using more functions from the API 109
12 Quality assurance and performance 111
12.1 Quality assurance 111
12.2 Performance 113
13 Contact 114
14 Acknowledgements 115

Figures

Figure 1: openLCA Nexus search engine 3
Figure 2: Decompressed ZIP file 4
Figure 3: Setup screen for installation in Windows 4
Figure 4: Installation file for Mac OS 5
Figure 5: openLCA Welcome page 6
Figure 6: Search function in openLCA 7
Figure 7: Options from the "Window" tab 9
Figure 8: openLCA Formula Interpreter 10
Figure 9: Right-clicking a tab 11
Figure 10: Empty Navigation window following openLCA installation 11
Figure 11: Creating a new database, step 1 12
Figure 12: Creating a new database, step 2 (data creation wizard) 12
Figure 13: New database containing openLCA reference data only 12
Figure 14: openLCA reference data 13
Figure 15: openLCA Nexus Map 14
Figure 16: Downloading a database from openLCA Nexus 15
Figure 17: Importing an existing zolca database, step 1 16
Figure 18: Importing an existing zolca database, step 2 16
Figure 19: Importing an existing zolca database, step 3 16
Figure 20: Importing an existing database into an active database, step 3 17
Figure 21: Importing a database, step 1 18
Figure 22: Importing a database, step 2 18
Figure 23: Importing a database, step 3 19
Figure 24: Copy data from excel 19
Figure 25: Paste it into the “Inputs” or “Outputs” section 20
Figure 26: Data is copied 20
Figure 27: Connecting to a remote database 21
Figure 28: Database element structure and flow of information 21
Figure 29: Exporting as EcoSpold1 and EcoSpold2, step 1 23
Figure 30: Exporting as EcoSpold1 and EcoSpold2, step 2 23
Figure 31: Exporting as EcoSpold1 and EcoSpold2, step 3 23
Figure 32: Exporting results as excel file 24
Figure 33: Exporting product system as excel file, step 1 25
Figure 34: Exporting product system as excel file, step 2 25
Figure 35: Exporting as ILCD, step 1 26
Figure 36: Exporting as ILCD, step 2 26
Figure 37: Exporting Project results as html file 27
Figure 38: Exporting project results in SimaPro CSV format, step 1 27
Figure 39: Exporting project results in SimaPro CSV format, step 2 27

Figure 40: Exporting as JSON-LD file, step 1 28
Figure 41: Exporting as JSON-LD file, step 2 28
Figure 42: Exporting as JSON-LD file, step 3 29
Figure 43: Copying data from openLCA tables 29
Figure 44: Collaboration Server, exemplary use case 30
Figure 45: Collaboration server – connecting local database to repository 31
Figure 46: Collaboration Server - User 1 connects locale database to repository 31
Figure 47: Collaboration Server - User 1 commits changes to repository 32
Figure 48: Collaboration server - User 2 connects to the same repository and fetches data 33
Figure 49: Collaboration server - summary of differences to local database 33
Figure 50: Collaboration server - Diff tool 34
Figure 51: Collaboration server - Commit history in openLCA, step 1 34
Figure 52: Collaboration server - Commit history in openLCA, step 2 34
Figure 53: Creating a new flow 35
Figure 54: Flow editor - General information tab 36
Figure 55: Flow editor - Flow properties tab 36
Figure 56: Creation of a waste treatment process 37
Figure 57: Waste flow as input in a waste treatment process 37
Figure 58: Waste input flow marked as avoided waste 37
Figure 59: Product system, model graph – System expansion: waste flow as avoided waste on the output side of a process. 38
Figure 60: Waste modelling. Waste treatment as a service for the process 38
Figure 61: Difference between unit process and system process 39
Figure 62: Unit processes (purple font colour) and system processes (white font colour on purple background) 39
Figure 63: Creating a new process, step 1 40
Figure 64: Creating a new process, step 2 40
Figure 65: Creation of waste treatment process 41
Figure 66: Process tab - General Information, data quality 41
Figure 67: Process tab - Inputs/Outputs, Data quality for flows 42
Figure 68: Data quality systems directory in an active database 42
Figure 69: Data quality systems, indicators and scores 42
Figure 70: Create new data quality system, step 1 43
Figure 71: Create new data quality system, step 2 43
Figure 72: Flow provider 44
Figure 73: Adding a reviewer or source to processes 44
Figure 74: Global, Input and Dependent parameters 45
Figure 75: Creation of a global parameter 46
Figure 76: Parameter usage throughout the database 46
Figure 77: Physical, causal and economic allocation 48
Figure 78: Flow property mass 48
Figure 79: System expansion, marking an avoided product in a process 48

Figure 80: System expansion - avoided product in model graph 49
Figure 81: Social indicators in "Indicators and parameters" directory 49
Figure 82: Social indicators, General information tab 50
Figure 83: Process, social aspects 50
Figure 84: Process - Social aspects, modify social indicators 51
Figure 85: Process - Social aspects, add a social indicator 51
Figure 86: Importing LCIA methods, step 1 52
Figure 87: Importing LCIA methods, step 2 52
Figure 88: Importing LCIA methods, step 3 53
Figure 89: LCIA methods in openLCA 53
Figure 90: LCIA methods - General information tab 54
Figure 91: Shape files tab, import files 55
Figure 92: Shape files tab, import files, show in map 55
Figure 93: Map displaying the values of the selected parameter for all the features included in the imported shape file 56
Figure 94: Shape files tab, Evaluate for existing locations selection 56
Figure 95: Creating a product system, step 1 57
Figure 96: Creating a product system, step 2 58
Figure 97: Product system - General information tab 60
Figure 98: Product system model graph - example 61
Figure 99: Model graph - miniature view 62
Figure 100: Model graph - build next tier 62
Figure 101: Model graph - search providers for, step 1 63
Figure 102: Model graph - search providers for, step 2 63
Figure 103: Model graph- search providers for, connected process 63
Figure 104: Model graph - drawing connections, step 1 64
Figure 105: Model graph - drawing connections, step 2 64
Figure 106: Model graph - deleting connections 65
Figure 107: Product system statistics 66
Figure 108: Drag-and-drop of a product system into another product system 67
Figure 109: Drag-and-drop of a product system into the input flows of a process 67
Figure 110: Calculating a product system, step 1 68
Figure 111: Calculating a product system, step 2 69
Figure 112: Calculating a product system, assessing data quality, step 1 69
Figure 113: Calculating a product system - assessing data quality, step 2 70
Figure 114: Quick results - General information tab 71
Figure 115: Quick results - Inventory results tab 72
Figure 116: Quick results - Inventory results, inputs section 72
Figure 117: Quick results - Inventory analysis - data quality information 73
Figure 118: Quick results - inventory analysis, added value calculation 73
Figure 119: Quick results - LCIA Results tab 73

Figure 120: Quick analysis - Impact Analysis, data quality 74
Figure 121: LCIA checks tab 74
Figure 122: Impact analysis tab 75
Figure 123: Impact factors tab 76
Figure 124: Quick results - Locations tab 77
Figure 125: Creating a new group, step 1 78
Figure 126: Creating a new group, step 2 78
Figure 127: Quick results - Grouping tab 79
Figure 128: Saving groups 79
Figure 129: Analysis - Process results tab 81
Figure 130: Analysis - Contribution tree tab 82
Figure 131: Analysis - Contribution tree tab, cost category 82
Figure 132: Analysis - Sankey diagram 83
Figure 133: Regionalized LCA – locations 84
Figure 134: Regionalized LCA - KML editor 85
Figure 135: Regionalized LCA - Calculation framework 85
Figure 136: Shape file parameter in the "Parameters" tab of the LCIA method editor 86
Figure 137: Regionalized LCA - Parameters applied to CFs 86
Figure 138: Calculation properties for Regionalized LCA performance 87
Figure 139: Regionalized LCA - Result map 88
Figure 140: Monte Carlo simulation – creation of a process subtracting one process to the other for future comparison in Monte Carlo simulation 89
Figure 141: Adding uncertainty information, step 1 89
Figure 142: Adding uncertainty information, step 2 90
Figure 143: Adding uncertainty information to LCIA characterisation factors 90
Figure 144: Monte Carlo Simulation, step 1 90
Figure 145: Monte Carlo Simulation, step 2 91
Figure 146: Monte Carlo Simulation, step 3 91
Figure 147: Monte Carlo Simulation results with the progress of the calculation 92
Figure 148: Pin/unpin processes or subsystems 93
Figure 149: Specification of costs and revenues for input and output flows in a process 94
Figure 150: LCC - Choice of allocation method when performing LCC 94
Figure 151: Uncertainty given to a price parameter 95
Figure 152: Calculation of total and stage added value 95
Figure 153: Performance of LCC stand-alone, no impact assessment method is selected 96
Figure 154: Contribution tree referred to the selected cost category 96
Figure 155: Creating a new project, step 1 97
Figure 156: Creating a new project, step 2 97
Figure 157: Project setup, LCIA method selection 98
Figure 158: Adding product systems to a project 98
Figure 159: Project setup, Variants 99

Figure 160: Parameter definition in processes 99
Figure 161: Project setup, Parameters 99
Figure 162: Project setup, Process contributions result example 100
Figure 163: Exporting a report in html format 100
Figure 164: Expanding memory on a mac, step 1 101
Figure 165: Expanding memory on a mac, step 2 101
Figure 166: Expanding memory on a mac, step 3 102
Figure 167: Accessing scripting 102
Figure 168: openLCA logger for scripting 102
Figure 169: Running script example 103
Figure 170: Error popup 103
Figure 171: Run button in the toolbar 104
Figure 172: Output of code 104
Figure 173: Read and write csv files in openLCA 106
Figure 174: Loading JavaScript files 106
Figure 175: Method declarations in API documentation 110
Figure 176: Classes and dependencies of core model 111
Figure 177: Comparison of inventory results for SimaPro and openLCA 112
Figure 178: PC features 113

Tables

Table 1: Ratio of results from SimaPro and openLCA in a calculation comparison, 5 lowest 112
Table 2: Ratio of results from SimaPro and openLCA in a calculation comparison, 5 highest 113
Table 3: openLCA and SimaPro performance comparison in network/analysis calculation 113

1 Introduction

1.1 Introduction to openLCA

openLCA is an open source software for Life Cycle Assessment (LCA) and sustainability assessment. It has been developed by GreenDelta since 2006 (www.greendelta.com). As an open source software, it is freely available, without license costs (www.openlca.org). The source code can be viewed and changed by anyone. Furthermore, the open source nature of the software makes it very suitable for use with sensitive data. The software as well as any models created can be shared freely, if the database license allows it. openLCA can be used for several different applications, for example:

- LCA, Life Cycle Costing (LCC), Social Life Cycle Assessment (S-LCA)
- Carbon & water footprints
- Environmental Product Declaration (EPD)
- United States Environmental Protection Agency (EPA) Design for the Environment label
- Integrated Product Policy (IPP)

This text focuses on the 1.9 version of openLCA. It explains how to carry out the first steps in working with openLCA such as installation and importing databases. This document then provides an overview of openLCA operations and features including descriptions of how to use them.

www.openLCA.org offers many different services to openLCA potential and current users. The website provides links to download the software, the source code, the openLCA LCIA (Life Cycle Impact Assessment) Method Pack, case studies, and user manuals, among other things. There are also links to instructional videos and documents in the "Learning and Support" section (www.openlca.org/learning). The ask.openLCA website (https://ask.openlca.org) is a question and answer website which serves as a public support platform. Furthermore, the openLCA team manages a twitter account (@openLCA) and a LinkedIn group (openLCA: free, professional Life Cycle Assessment (LCA) and Footprint software) to keep users up-to-date on news and recent developments.

1.1.1 What is new in version 1.10.1

Images in process description

Images (for example flowcharts) that support the information given in the process description and are linked by the database provider are now visible in the ‘general information’ tab of processes.

openLCA for MacOS and Linux improvements

Several improvements and bug fixes for Linux and MacOS users have been implemented. More technically-oriented users can learn more by reading about these improvements on the openLCA GitHub page. The main improvements are that openLCA can now run on Linux using GTK3, fixing the memory allocation issue for MacOS and accessing the map view in the results tap for all users.

1.2 Introduction to GreenDelta

GreenDelta was founded by Dr Andreas Ciroth in 2004. Since the beginning, the core idea has been to provide life cycle-based consulting to businesses world-wide. GreenDelta has been developing openLCA since 2006. We continuously improve the software, expand its features and capabilities and keep it up-to-date with current LCA practice. We are at the forefront of LCA knowledge, offering specialized features such as regionalized LCIA as well as the ecoinvent 3.1 database. Next to software development, GreenDelta also offers Sustainability Consulting services such as guided case studies, research, critical reviews, EPDs and data management solutions. GreenDelta has an international reputation for pioneering professional yet free open source solutions for LCA and for our role as a common denominator in the international LCA community.

1.3 Introduction to openLCA Nexus

openLCA Nexus (https://nexus.openlca.org) is an online repository for LCA data. It combines data offered by world-leading LCA data providers such as the ecoinvent centre (ecoinvent database), PE International (GaBi databases), and the Joint Research Centre from the European Commission (ELCD database).

Datasets provided in Nexus can be easily imported into the openLCA software. openLCA and Nexus databases share a common set of elementary flows and other reference data which have been harmonized in coordination with the respective data providers to overcome methodological differences, for example concerning the modelling of waste.

Nexus contains free and “for purchase” data sets. For ordering and downloading databases from Nexus you need to sign up using a valid email address. The procedure here is very similar to a web shop: simply add the database into a shopping cart and order it. To find out more on how to download databases from openLCA Nexus and import them into openLCA, please see section 4.4.

The Nexus website contains a powerful search engine for LCA data that allows you to search for data sets in Nexus. It is also possible to filter data sets by the data provider, location, category, price and year of validity. We hope this makes it easy to identify the LCA data that you need.

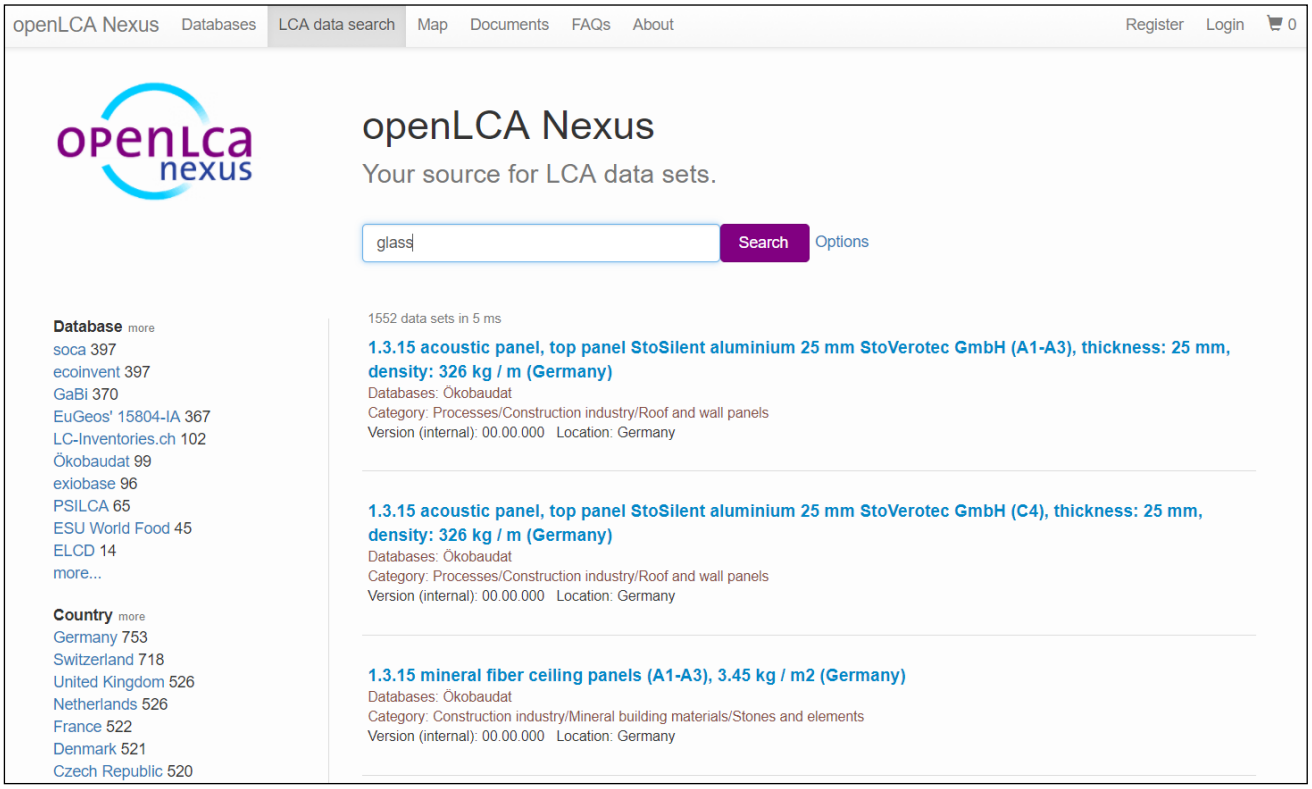


Figure 1: openLCA Nexus search engine

2 Installation

The installation of openLCA is slightly different for Windows, Mac and Linux, respectively. The following sections will explain how to install the software on different systems.

2.1 Windows

For windows, a 64-bit version is available.

Hardware, required:

- CPU with 2 GHz or higher
- 1 GB RAM (for analysing product systems with ~2500 processes, such as ecoinvent 2)
- >3 GB RAM (for analysing product systems such as ecoinvent 3)
- 6 GB RAM (for analysing product systems such as ecoinvent 3.4 or Psilca for social LCA)
- 500 MB free hard disk space + space for databases (e.g. ecoinvent 3 requires ~250MB)

Software required for the “Projects” feature:

- Windows 64 bit (for modern browser support): Microsoft Visual C++ 2010 Redistributable package (x64) (http://www.microsoft.com/de-de/download/details.aspx?id=14632)

The fastest way to get openLCA is to download the ZIP-archive. After decompressing, simply launch the executable file (Figure 2) and the program starts. No installation is needed.

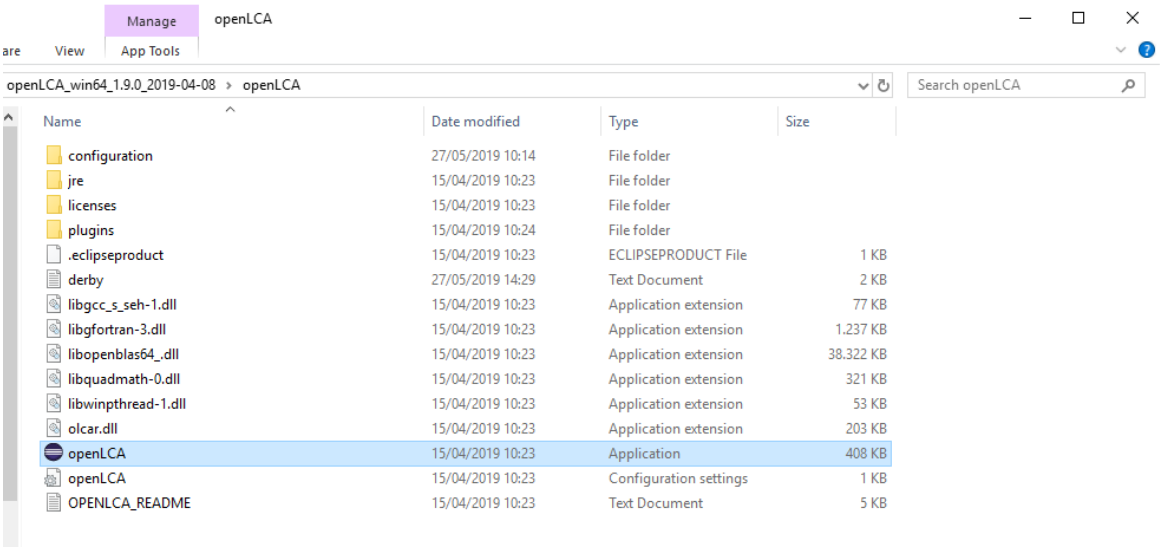


Figure 2: Decompressed ZIP file

Alternatively, the software can be installed. Start by downloading the installer file from the downloads page of the openLCA website. As usual in Windows installation, you can select whether openLCA should be used only by the user who installs (i.e. by you), or by anybody working on the computer (Figure 3).

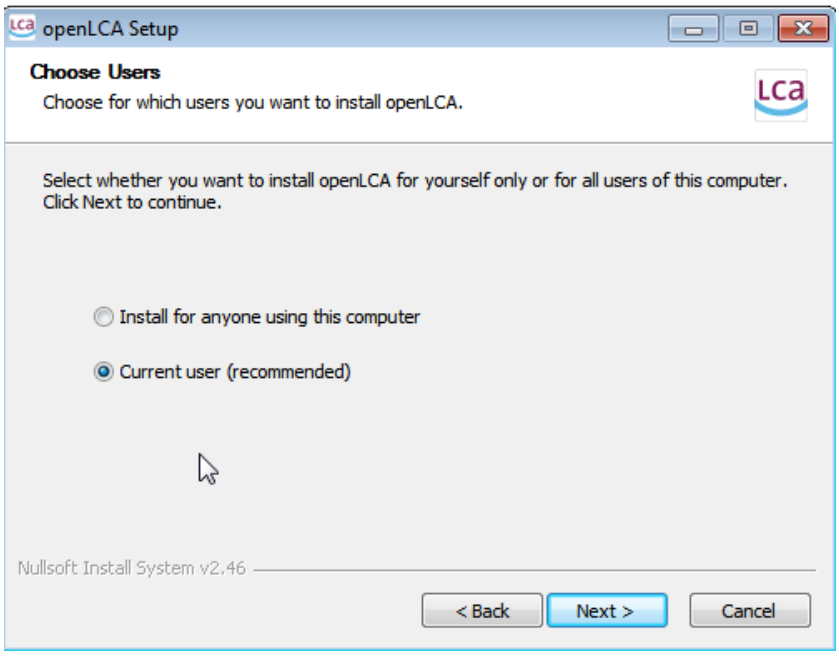


Figure 3: Setup screen for installation in Windows

Follow the installation steps to the end and you can begin working with openLCA.

2.2 Mac

A 64-bit version is available for Mac OS.

Hardware, required:

- CPU with 2 GHz or higher
- 1 GB RAM (for analysing product systems with ~2500 processes, such as ecoinvent 2)
- >3 GB RAM (for analysing product systems such as ecoinvent 3)
- 500 MB free hard disk space + space for databases (e.g. ecoinvent 3 requires ~250MB)

Software, required:

- Java in version 8; install Java before beginning with the openLCA installation (Java SE Development Kit for Mac OS available under http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html).

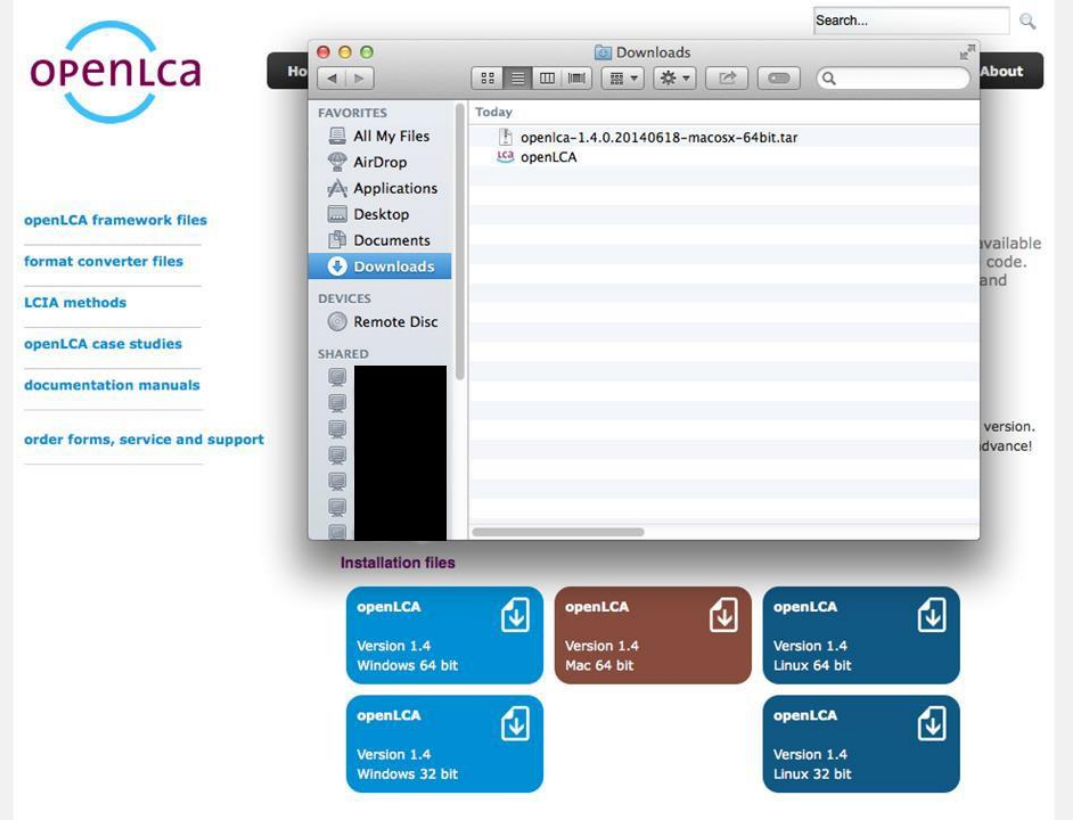


Figure 4: Installation file for Mac OS

Start by downloading the correct file from the downloads page of the openLCA website. Once the download is complete find the program in the "Downloads" folder (Figure 4) and transfer it into "Applications". Double-click on the program to open it. A warning message will appear because the application was downloaded from the internet and not from the App Store. Select "open". This message

will only appear the first time you open the program.

2.3 Linux

For Linux, a 64-bit version is available.

Hardware, required:

- CPU with 2 GHz or higher
- 1 GB RAM (for analysing product systems with ~2500 processes, such as ecoinvent 2)
- >3 GB RAM (for analysing product systems such as ecoinvent 3)
- 500 MB free hard disk space + space for databases (e.g. ecoinvent 3 requires ~250MB)

We recommend installing libgfortran3 for high performance calculations.

3 Welcome to openLCA

When you start openLCA for the first time, it does not contain any data. On the left-hand side, you see an empty Navigation field. On the right you see the Welcome page in the so-called "Editor".

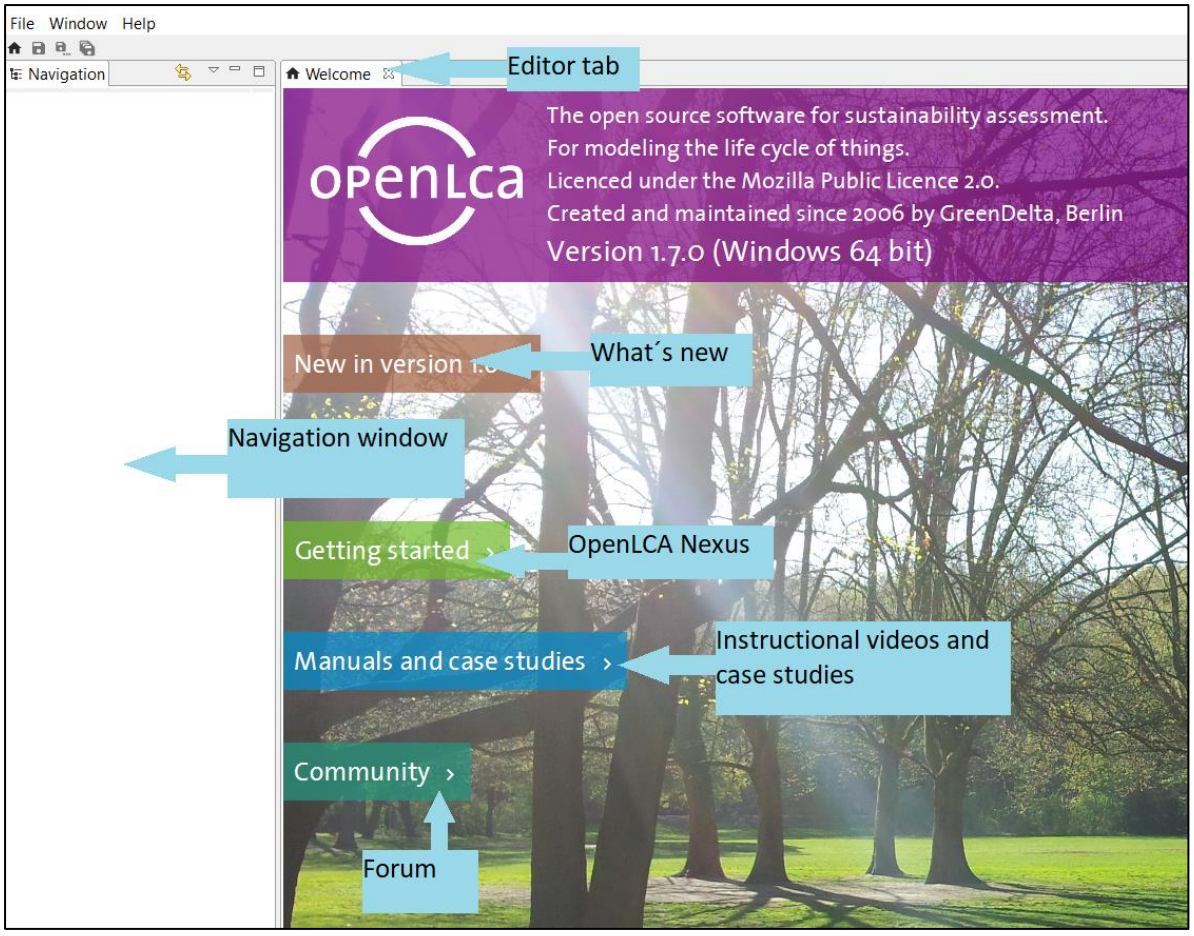


Figure 5: openLCA Welcome page

The Welcome page provides quick links to openLCA Nexus, instructional videos, case studies, this user manual, the openLCA download page where you can download the latest version of the software as well as LCIA methods and, finally, a link to more information on the openLCA network and its users.

On the top-right part of the page the “Search” function gives you the possibility to look for anything you like in openLCA (e.g. flows, processes, social indicators, currencies, etc.) searching in all or in one specific section.

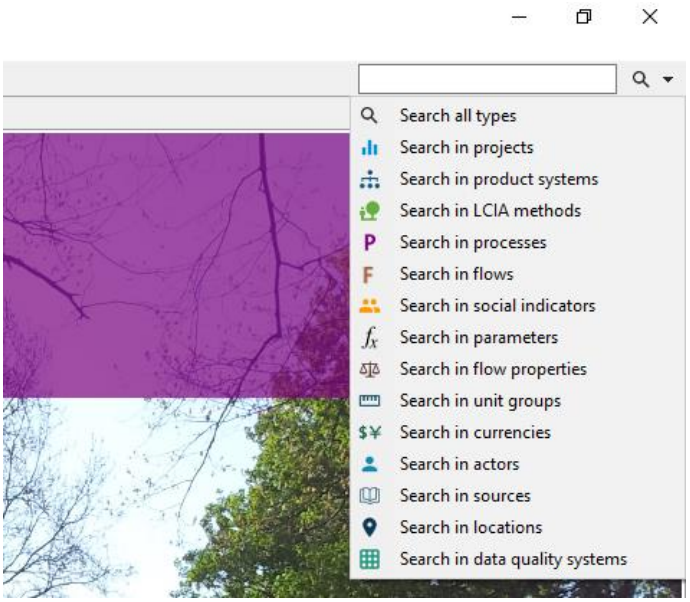


Figure 6: Search function in openLCA

3.1 Main menu functions

The following options are available under "File":

- "Save" / "Save As..." / "Save All": save current work open in editor tabs
- "Close" / "Close All": close the current/all windows open in the editor.
- "Settings":
    - “Collaboration”: set server configurations.
    - “Configuration”: select one of the eleven available languages you would like to work in (Arabic, Bulgarian, Catalan, Chinese, English, French, German, Italian, Portuguese, Spanish or Turkish). Here you can also select the maximum memory usage. It is recommended to increase this value for calculating very complex product systems (e.g. to ~4,000MB). This feature does not work for Mac OS. If you would like to expand your memory on a Mac OS operating system, please see section 11.1. Please note: you need to restart openLCA to activate configuration changes.

    - “Experimental features”: experimental features are features that are still in the beta stage of development, but we want to make them available to you and welcome any feedback.
    - “Import/export”: set ILCD Network URL, User, Password and language.
    - “Logging”: Set logging settings.
    - “Number format”: Set the number of decimal places for numbers displayed.
- "Import" / "Export": For more information on openLCA Import and Export options, see sections 4.4 and 4.7 respectively.
- “Exit”: Closes openLCA

The following options are available under “Database” when a database is opened:

- “New Database”: For creating a new database, see section 4.1. It is also possible to activate the “New Database” function by right-clicking the navigation window.
- “Restore Database”: For restoring a database, see section 4.2. It is also possible to activate the “Restore Database” function by right-clicking the navigation window.
- “Backup Database”: Copy the database into an archive file to save it.
- “Validate”: Checks whether all linkages within the database are working.
- “Copy”: create a copy of the active database
- “Rename”: rename the active database
- “Delete Database”: Removes the database from the main window
- “Close Database”: Closes the active database
- “Check linking properties”: A comprehensive provider check of the active database that displays the results in a table, showing if processes without a default provider exist, if product or waste flows exist with multiple providers, if and which product flows have multiple providers and what provider linking options are uncritical when working with the active database.
- “Properties”: Shows extended information about the database such as its location on the PC.
- “Compress database”: Compresses the database to save space on the hard drive.
- “Open Update Manager”: With the update manager it is possible to update the openLCA software

The following options are available under "Window" after clicking on “show views” and “other”:

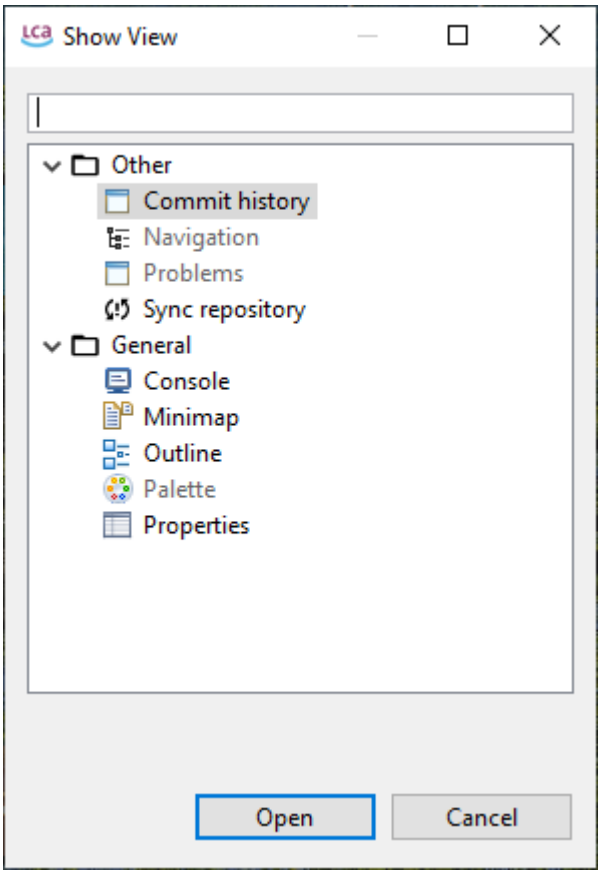


Figure 7: Options from the "Window" tab

- “Commit History”: Shows the commit history of the synchronisation with the collaboration server, see section 4.8.
- “Navigation”: The Navigation window displays the databases you have imported into openLCA and all the data sets they include
- “Problems”: List all synchronisation problems with the collaboration server, see section 4.8
- “Sync repository”: Synchronise with the collaboration server, see section 4.8
- “Console”: Displays the log
- “Minimap”: Not available, a residue from creating openLCA with eclipse
- “Outline”: Displays a list of all the processes of a product system, including all its background processes. It is only applicable after you’ve created a product system. Open the product system’s Model Graph (tab) and choose the “Outline” option from “Views”. The outline allows you to choose the processes you wish to show or hide from the Model Graph.
- “Palette”: Not available, a residue from creating openLCA with eclipse
- “Properties”: Not available, a residue from creating openLCA with eclipse

Furthermore, the following options are available under “Window”:

- “Parameters”: The parameter overview lists all parameters in the active database regardless of whether they are local or global. Upon opening the parameter overview, right-click -> Evaluate *formulas* allows to evaluate the underlying formulas. The filter field allows to search for specific parameters or to filter for errors. Via right-click -> Edit, formulas can be changed directly from the parameter overview.
- "Developer Tools": (to find out more details on scripting in openLCA, see Section 12.2)
    - “SQL”: A tool that can be used to carry out SQL queries in openLCA.
    - “JavaScript”: openLCA supports the possibility to run JavaScript programs directly in openLCA. With this feature you can automate calculations in openLCA, write your own data imports or exports, perform sensitivity analysis calculations by varying parameter values, and much more. See section 11.2.2 for more information.
    - “Python”: openLCA supports the possibility to run Python programs directly in openLCA. With this feature you can automate calculations in openLCA, write your own data imports or exports, perform sensitivity analysis calculations by varying parameter values, and much more. See section 11.2.1 for more information.
    - “IPC Server”: Inter Process Communication is a platform-independent data exchange interface via *http*. *IPC Server* allows to run openLCA services via the Python’s standard library.
- "Bulk-replace": it is a tool that allows replacement of a flow or product provider with another flow or provider.
- "Formula interpreter": Complex formulas for parameters must be typed in using the correct format. Use this interpreter to check if your formulas are correct. Start by opening the formula interpreter and then type 'help' to display the help message (Figure 8)

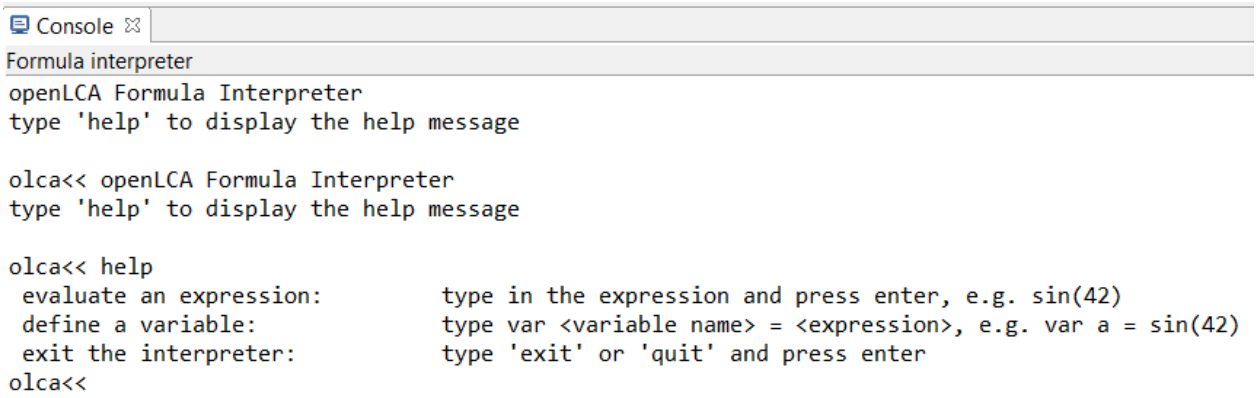


Figure 8: openLCA Formula Interpreter

Under "Help" you can find information on the openLCA copyright and openLCA log file as well as a link to

this user manual.

3.2 Tabs

openLCA sorts each new window into a tab next to the welcome icon in the main window. A right-click on a tab offers various tab management options that improve the usability of openLCA (Figure 9).

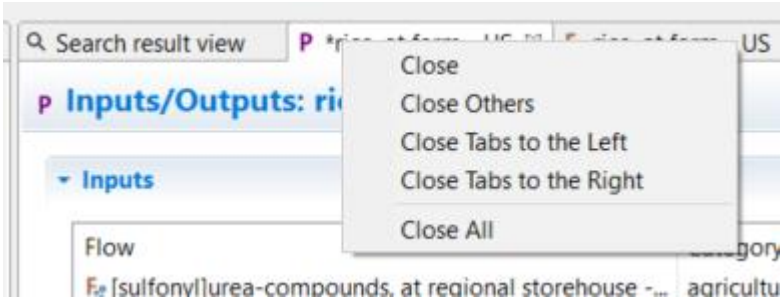


Figure 9: Right-clicking a tab

Tabs can be detached and moved around on the screen. Dragging and dropping a detached tab next to an existing tab in openLCA reverses the detaching. Detaching tabs allows you to run openLCA in a single-window and multi-window mode, especially when working with multiple screens

4 Databases

Following installation openLCA does not contain any data, therefore the 'navigation' section on the left is empty (Figure 10). It is possible to have more than one database. Databases are independent of each other and only one database is "active" at a time. All the others are "inactive". It is also possible to combine databases (section 4.4)

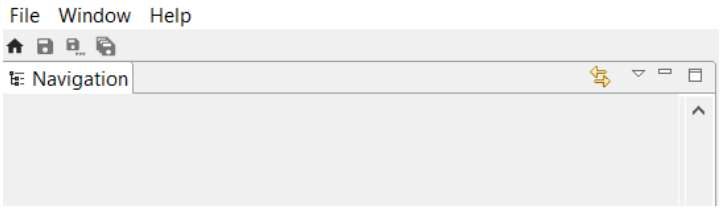


Figure 10: Empty Navigation window following openLCA installation

To change this, you can either:

- Create a new, empty database, or
- Restore a database.

4.1 Creating a new, empty database

Right-click in the navigation window and select "New database":

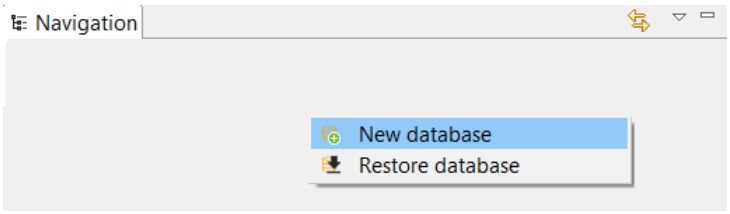


Figure 11: Creating a new database, step 1

The data creation wizard will then appear where you can select your settings for the new database:

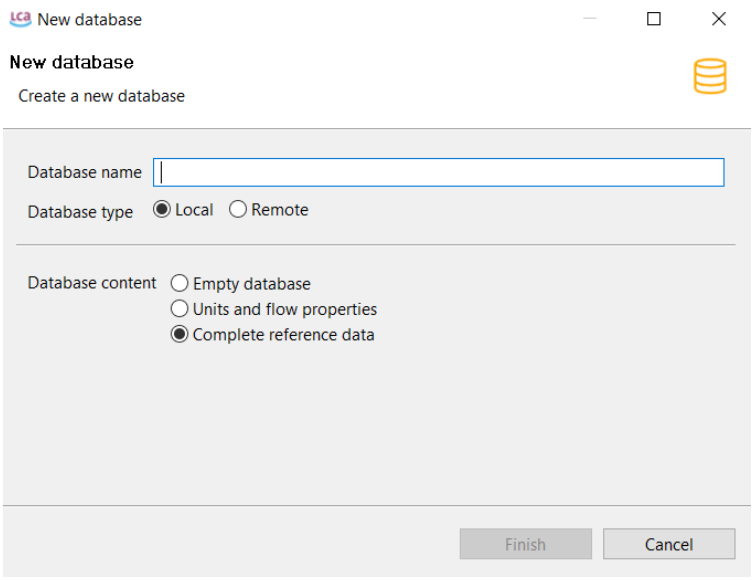


Figure 12: Creating a new database, step 2 (data creation wizard)

For the beginning, it is recommended to create the database with the settings ‘local database’ and ‘complete reference data’. Then select "Finish". After a few seconds, you can have a look at the newly created database:

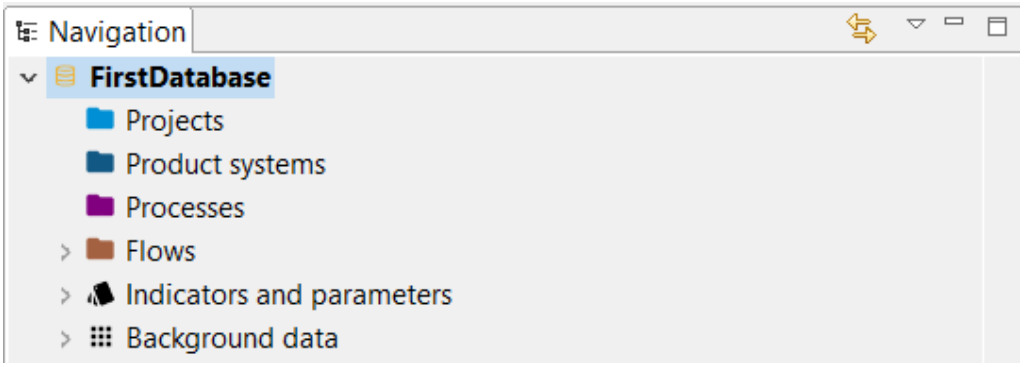


Figure 13: New database containing openLCA reference data only

The elements of the database will be familiar to you if you have worked with previous versions of openLCA. They will be explained in more detail in section 4.6.

The openLCA reference database contains flows, indicators and parameters, and background data. All the

databases available in openLCA Nexus are mapped to ensure that the software recognizes and applies all elements of each database correctly (for example, to ensure no double flows are created during import and that the impact assessment methods available deliver correct results).

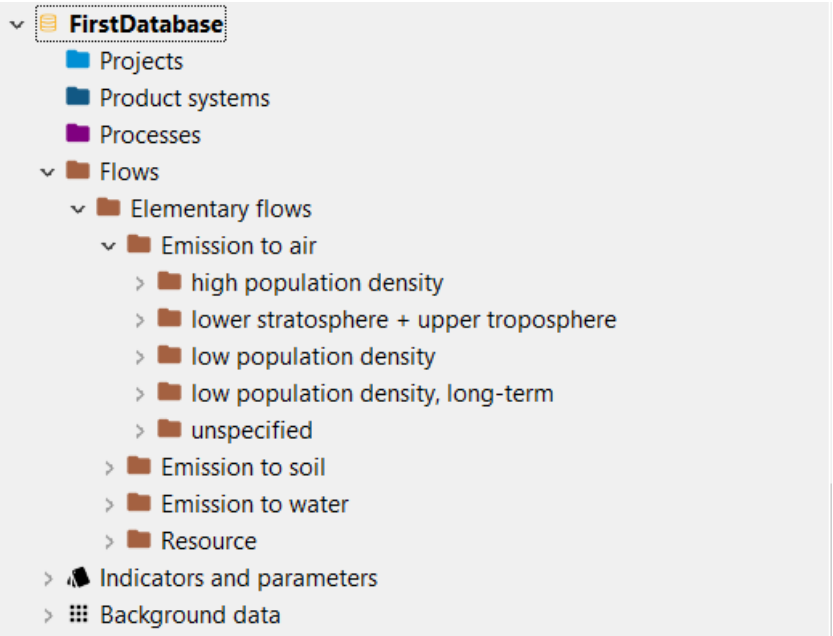


Figure 14: openLCA reference data

This database does not contain any process data sets. You can start creating your own processes using the flows and units that are already present.

4.2 Restoring a database

As a second option, next to creating a new empty database, you can restore a database. Databases can for example be a database downloaded from the openLCA Nexus website in zolca format. All the databases available in openLCA Nexus are mapped to ensure that the software recognizes and applies all elements of each database correctly. To restore a database, first create a new database as shown in Figure 11 & Figure 12, but select “empty database”. Then right-clock again and select “restore database” as shown in Figure 11 and select the database (zolca-format) from a folder. The progress bar is in the bottom right corner.

4.2.1 Accessing databases from openLCA Nexus

As mentioned in section 1.3, openLCA Nexus (https://nexus.openlca.org) is an online repository for LCA data. Nexus contains for free and for purchase databases. You can browse through the content of the offered databases using the search engine in the sections "LCA data search" and "Map". Using the LCA data search feature, you can search for specific data sets according to name or category (i.e. database, country, other location, Type of data, Category, Price and Start of validity). The openLCA Nexus Map is interactive and illustrates how many data sets are available in different locations. The darker the shade on the map, the more data sets are available for that region. The search can also be reduced by name and

by type.

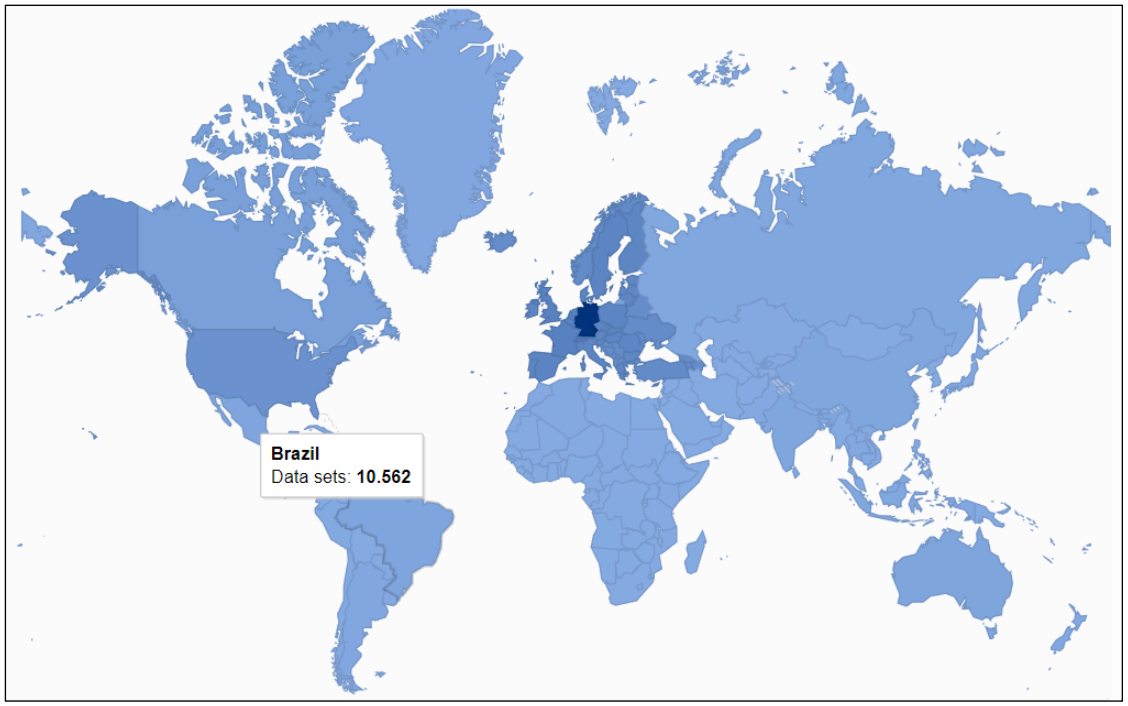


Figure 15: openLCA Nexus Map

To download a database from the openLCA Nexus site you need to first order it. Some databases in openLCA Nexus are available for free and some for charge. To order a database, begin by registering an account.[1] Once you are logged in, go to the "Databases" section and select the license you are interested in. Then add the license you want to purchase to the cart and order it. To find out more about licenses and how to place an order in openLCA Nexus, please see the website's FAQs.

Once you have made an order and it is approved, you can download the database in the "Downloads" section which will appear in the top right-hand corner of the Nexus website when you are logged in. In this section, you will see all the data files available to you for download. Select the files you would like to download as well as the format (e.g. openLCA 1.5). Once you have read and agreed with the licenses and the EULA, check off the two boxes at the bottom, then select "Download" (Figure 16).

[1] Registration is quick and helps us to understand and meet the needs of the user base. We are bound to our high data protection principles and protect your privacy.

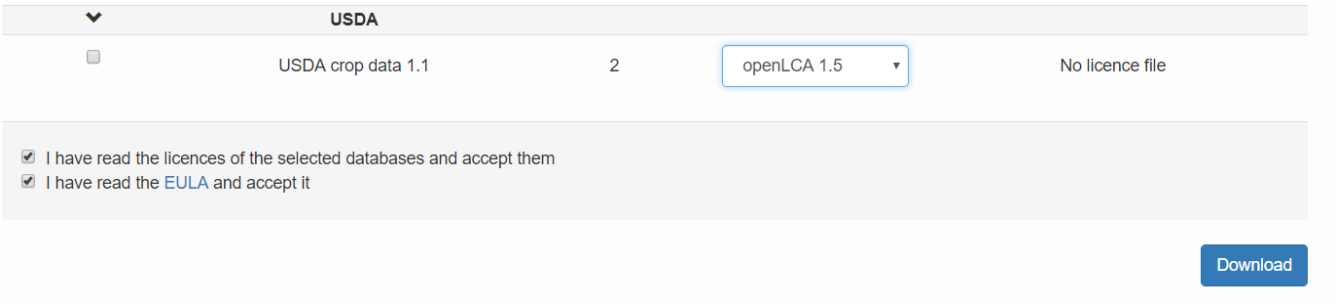


Figure 16: Downloading a database from openLCA Nexus

4.3 Database scheme update

openLCA features an updated database scheme. Upon opening databases with an old database scheme, openLCA will automatically prompt for an update. For very critical data it is recommended to create a backup before changing to a new database scheme.

4.4 Importing and combining databases

It is possible to combine databases into one openLCA database. Databases can for example be a database downloaded from the openLCA Nexus website. All the databases available in openLCA Nexus are mapped to ensure that the software recognizes and applies all elements of each database correctly (for example, to ensure no double flows are created during import and that the impact assessment methods available deliver correct results).

To combine databases, import the first database (we recommend importing the largest database first as otherwise the time it takes to compile the databases can be much longer). Then activate the database by double-clicking on it. At this point you can import the other databases. The software will then combine the two databases. Dependent on the size of the databases, this can take a minute or two. Click here to see an instructional video on combining ecoinvent 3.1 databases.

openLCA supports the following import formats:

- zolca
- Ecospold1
- Ecospold2
- Excel
- ILCD
- SimaPro CSV
- JSON-LD

4.4.1 Importing a database from exported zolca-File

Once you have a zolca file saved on your computer, you can import it directly to openLCA. To do this, select

“File”, “Import”, “Import entire database” and finally the second option "From exported zolca-File":

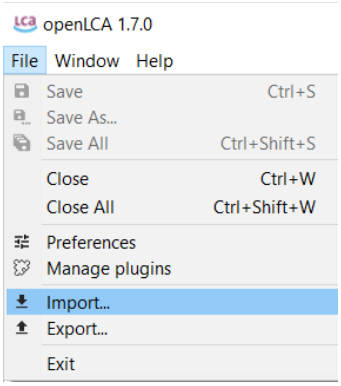


Figure 17: Importing an existing zolca database, step 1

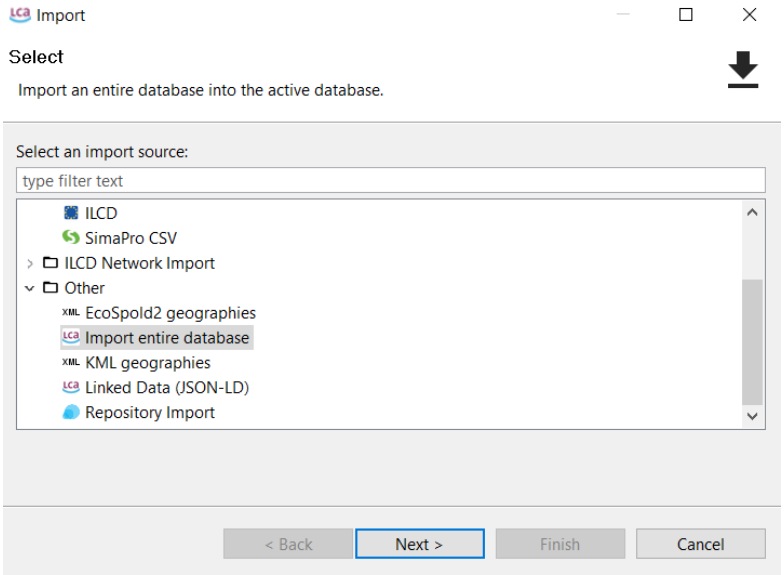


Figure 18: Importing an existing zolca database, step 2

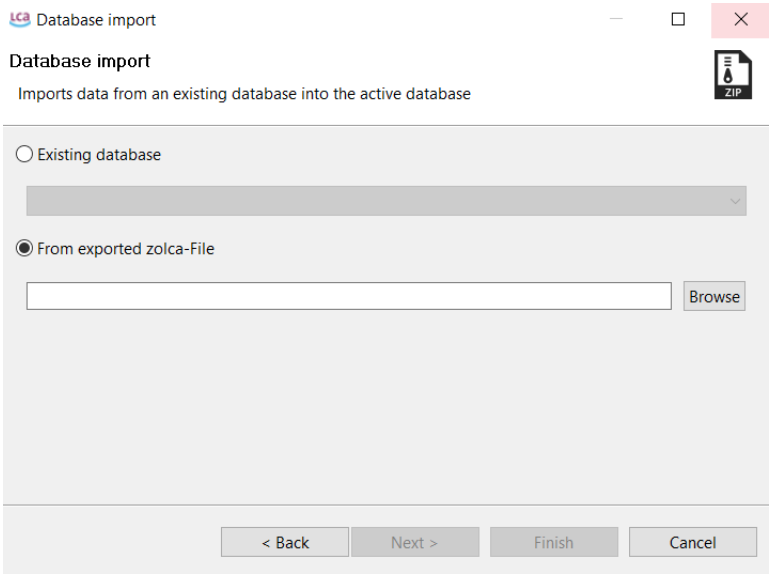


Figure 19: Importing an existing zolca database, step 3

Then find the ‘zolca’ file you would like to import in your browser and select "Open". The database will be

'inactive' at first. To activate a database, double-click on it. Then you will have access to all the flows, processes, etc. the database contains.

It is faster to get zolca-format databases by using the “restore database” strategy (see section 4.2).

4.4.2 Importing an existing database

It is possible to import data from an existing database into the active database. To do this, select “File”, “Import”, “Import entire database” and finally the first option "Existing database". Step 1 and 2 are the same as in the previous section.

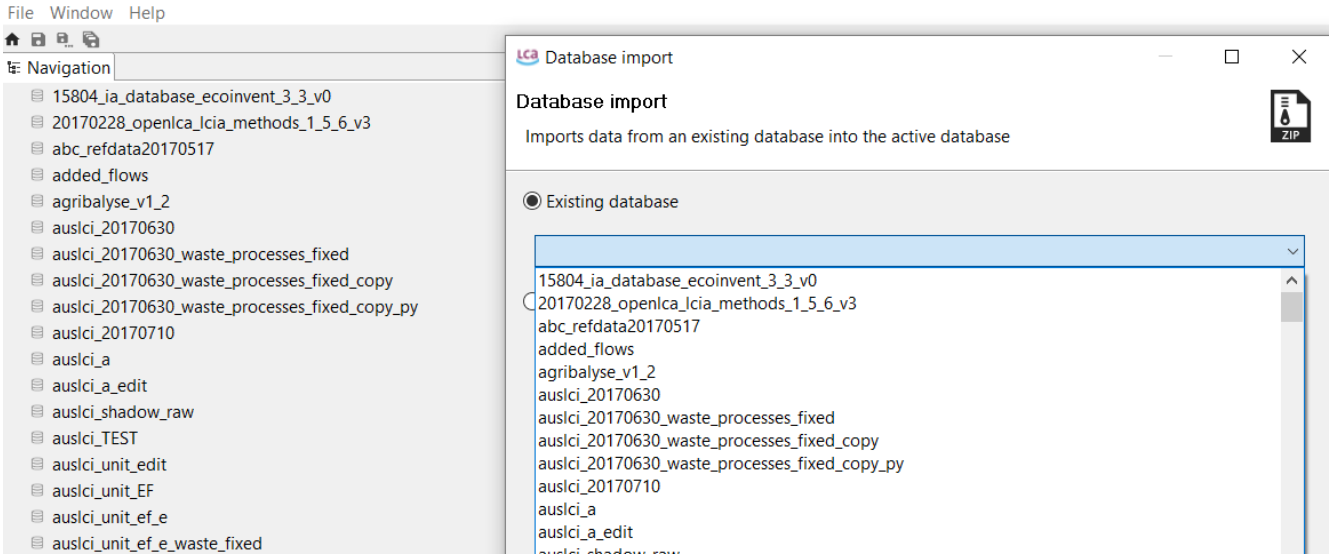


Figure 20: Importing an existing database into an active database, step 3

4.4.3 Importing databases in ecospold, Excel, ILCD, SimaPro CSV, and JSON-LD formats

These file types can be imported into existing openLCA databases. If necessary, create a new, empty database. Before importing, double-click on the target database to activate it. Then go to "File" --> "Import". Then select the format and select the import file from the directory. For ecospold 1&2 it is necessary to check and assign units. Select "Finish" to import the data (depending on the data, the import can take a couple of minutes). Note: databases in ILCD format must be imported as zip files. See Figure 21, Figure 22, and Figure 23 as examples. JSON-LD is the in-house format for openLCA. It is possible to import whole databases or just certain product systems.

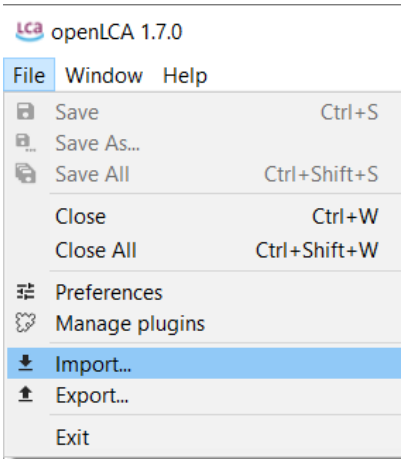


Figure 21: Importing a database, step 1

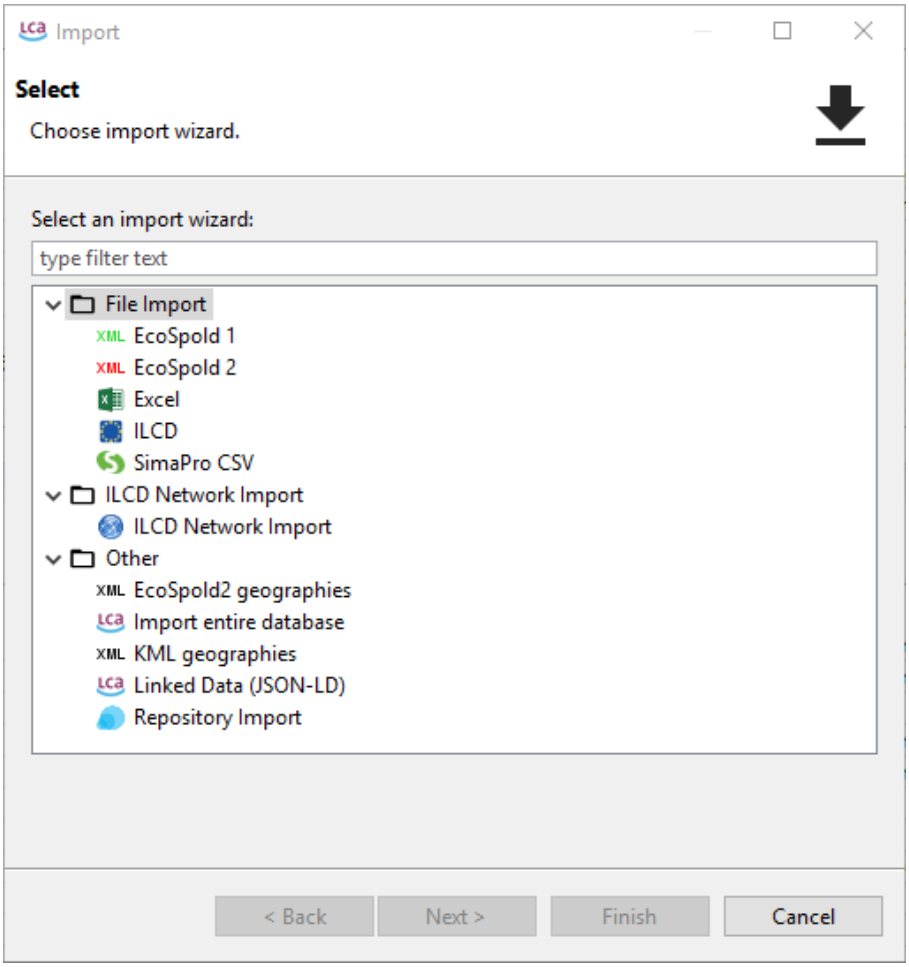


Figure 22: Importing a database, step 2

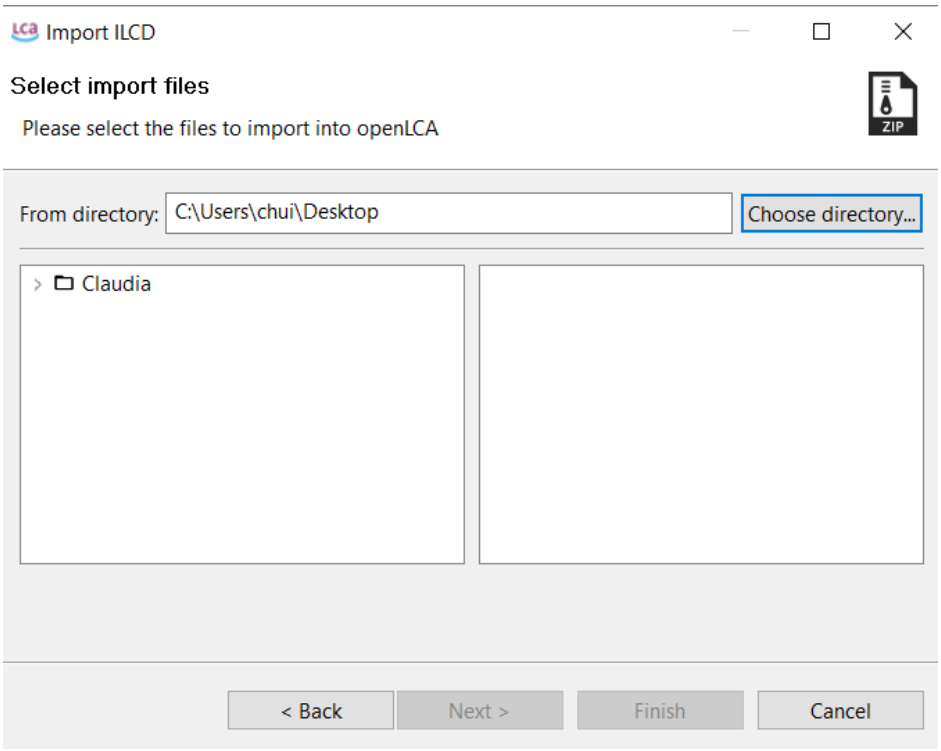


Figure 23: Importing a database, step 3

It is also possible to simply copy data from excel into openLCA, as shown in Figure 24, Figure 25, and Figure 26.

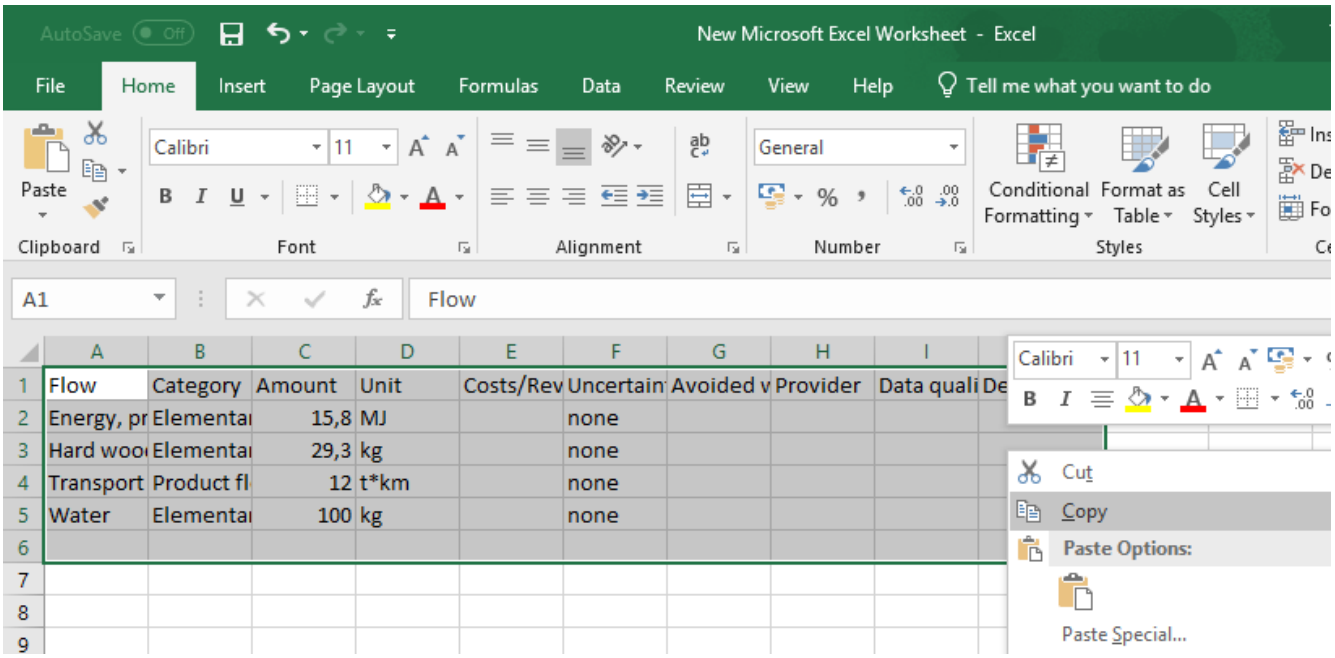


Figure 24: Copy data from excel

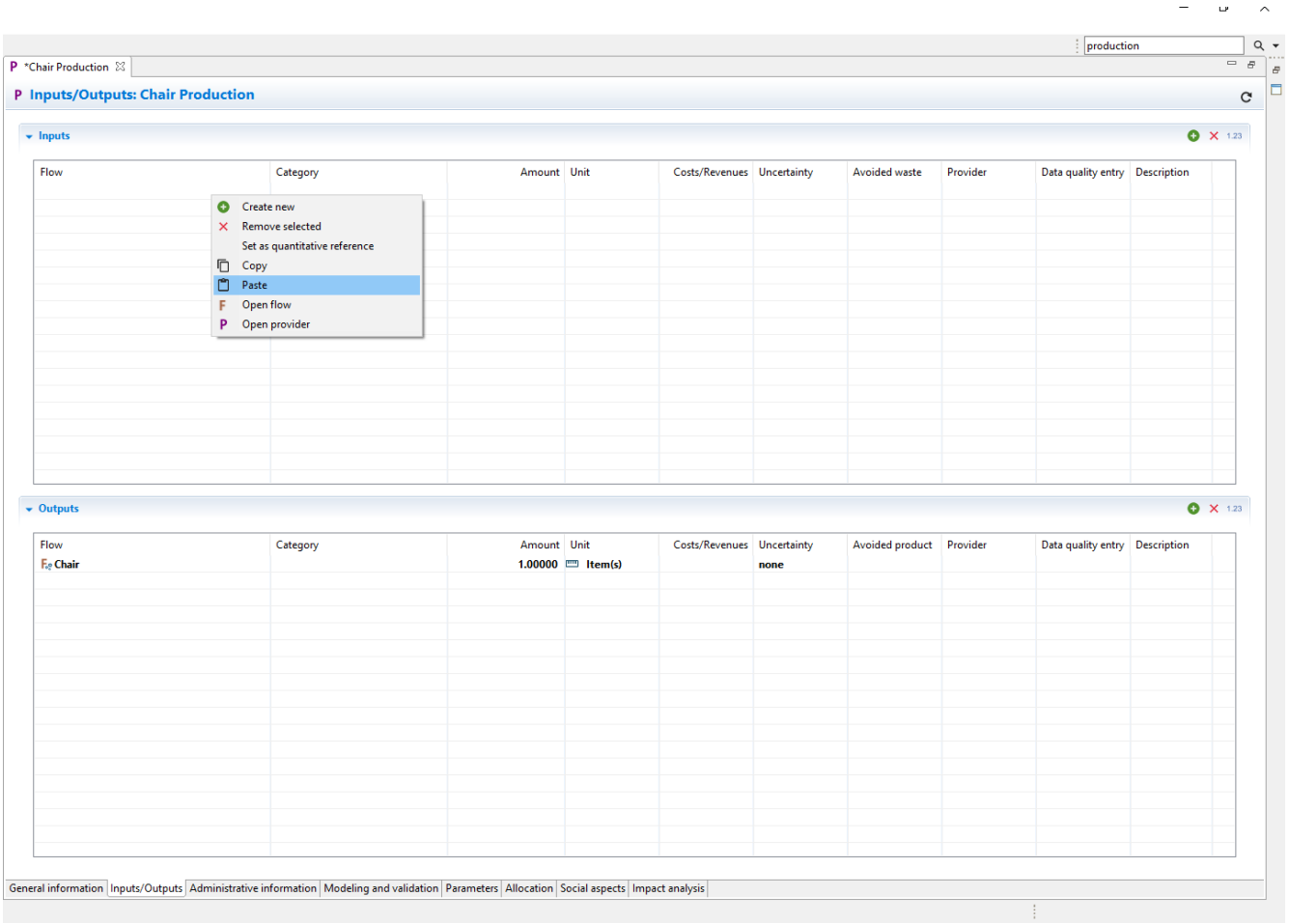


Figure 25: Paste it into the “Inputs” or “Outputs” section

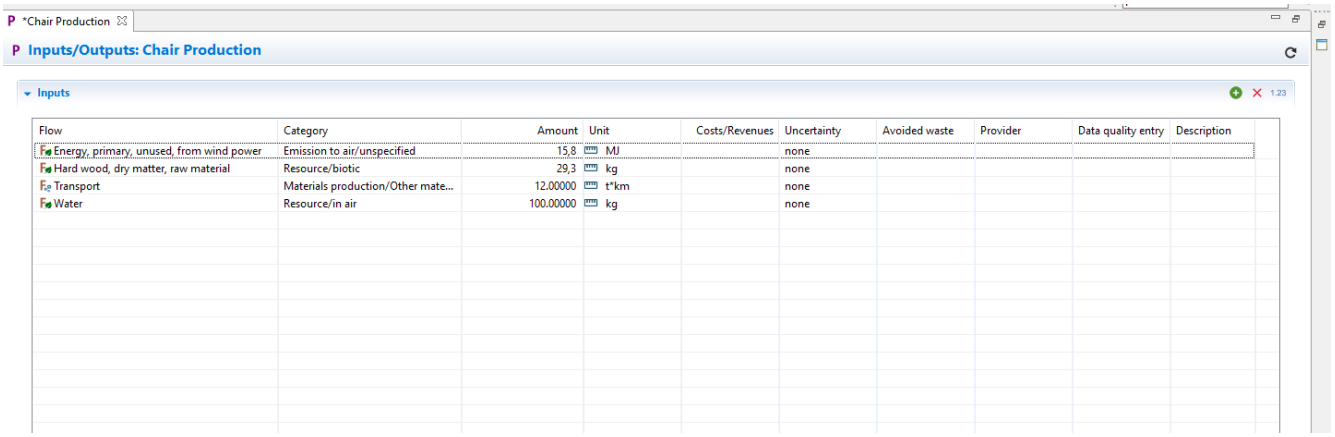


Figure 26: Data is copied

4.5 Creating a remote database

You can also create a remote database. The name should be the same as in the remote database, and you need to type in the information of the “Host”, “Port” and “User” from the remote database you want to connect (Figure 27).

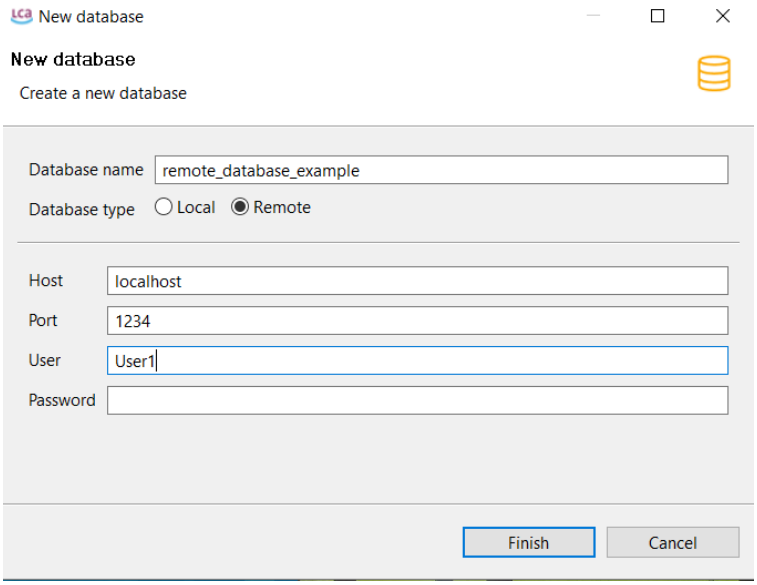


Figure 27: Connecting to a remote database

4.6 Database elements

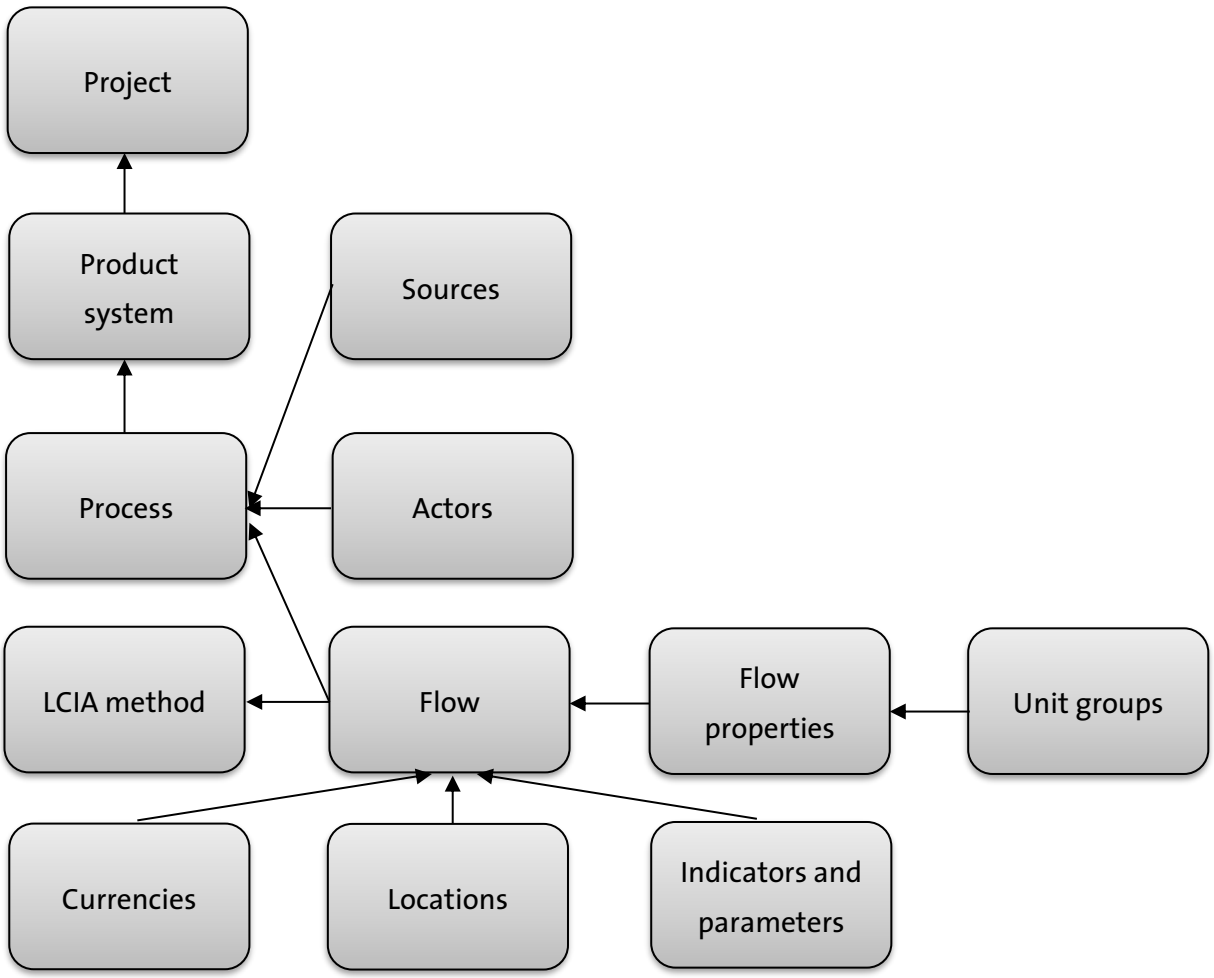


Figure 28: Database element structure and flow of information

The databases in openLCA contain the following elements:

- Actors: people who have provided data or modified models
- Currencies: cost can be assigned to flows and Life Cycle Costing can be performed
- Locations: important for regionalized LCA
- Sources: literature referenced
- Unit groups: groups of units (e.g. units of area include m^2, ft^2, sq. yd, etc.)
- Flow properties: properties of flows (e.g. length, mass, etc.)
- Flows: products and materials
- Processes: production or modification of products and materials
- Impact methods: impact assessment methods imported into openLCA
- Product systems: process networks (necessary to calculate inventory results and impact assessment)
- Projects: can be created to compare product system variants
- Indicators and parameters: social indicators, global parameters, data quality systems

4.7 Exporting data

openLCA supports data export in the following formats:

- Ecospold2 (entire databases, impact methods, processes,)
- ILCD Zip-file (actors, flow properties, flows, LCIA methods, processes, product systems, sources, unit groups)
- ILCD Network Export (entire databases, processes)
- Excel (processes, quick results, analysis results, Monte Carlo simulation results, product systems)
- JSON-LD
- "Copy" function for all openLCA tables

How to export these data types will be explained in the following sections.

4.7.1 Exporting data in EcoSpold formats

Processes and impact assessment methods can be exported in EcoSpold1 format. EcoSpold2 files can be created for processes only. To export these file types, activate the database from which you would like to export processes/impact assessment methods. Then click on "File" --> "Export". The export wizard will pop up. Select "EcoSpold2" to export processes as EcoSpold2 files, or "Impact methods" or "Processes" to export these as EcoSpold1 files. In the next window select the directory and processes/LCIA methods to be exported and click "Finish".

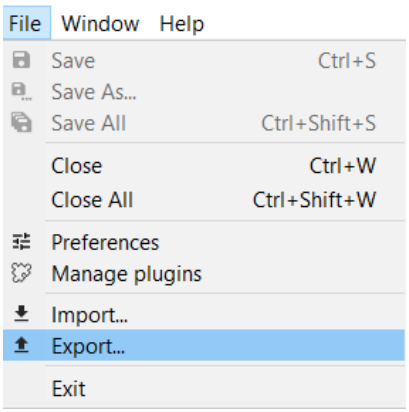


Figure 29: Exporting as EcoSpold1 and EcoSpold2, step 1

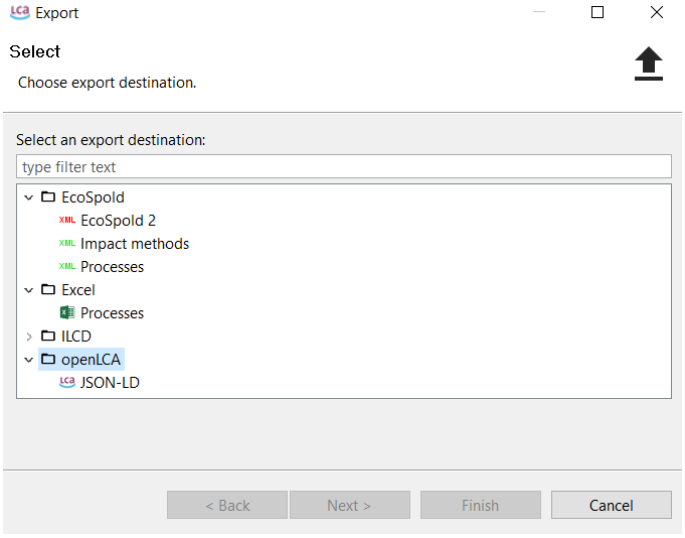


Figure 30: Exporting as EcoSpold1 and EcoSpold2, step 2

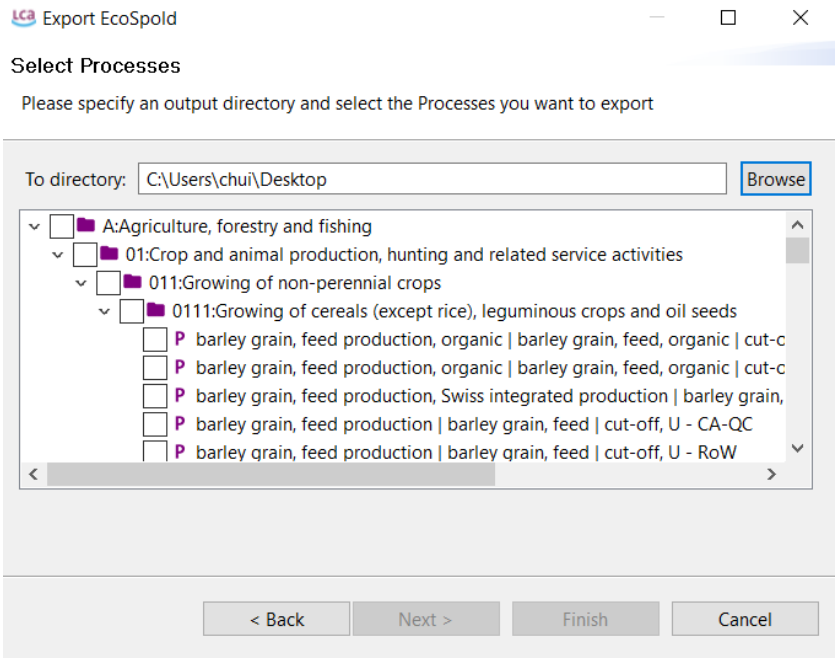


Figure 31: Exporting as EcoSpold1 and EcoSpold2, step 3

4.7.2 Exporting data in Excel format

OpenLCA can export processes, quick results, analysis results, Monte Carlo simulation results, product systems (elementary flows and product flows) and LCIA factors in Excel format. It is also possible to copy any tables from openLCA and paste them elsewhere.

- To export processes as excel files begin by activating the database from which data should be exported. Then select "File" → "Export" as illustrated in Figure 29. The export wizard will pop up. Select "Processes" in the Excel folder and click "Next". Then select the location where the data should be saved as well as the processes to be saved and click "Finish". Each process will be saved as an individual excel file.
- To export quick results, analysis results, and Monte Carlo simulation results you can either click on the excel export icon in the top left-hand corner or select the "Export to Excel" button under the "General information" tab of the results editor (Figure 32).

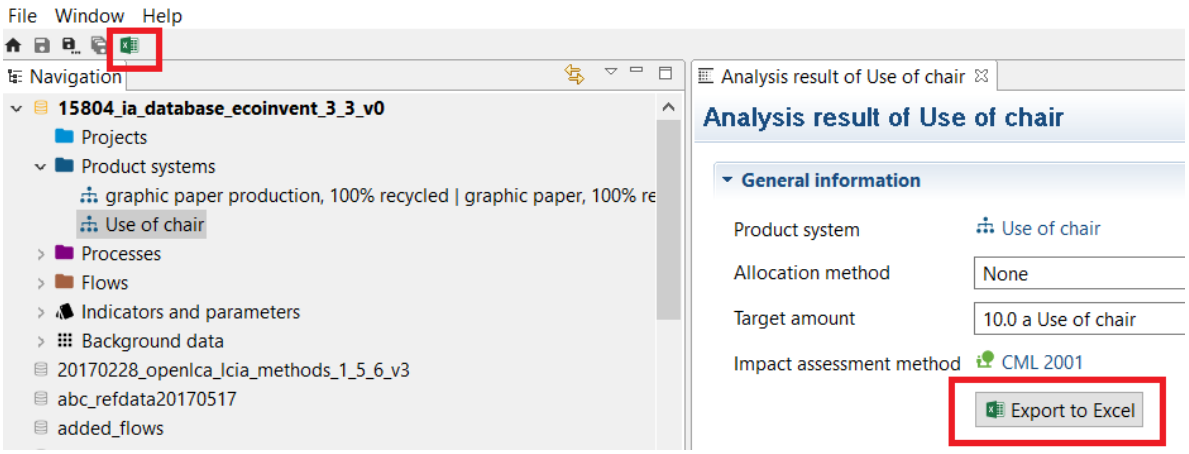


Figure 32: Exporting results as excel file

- To export product systems as excel first select the product system in the Navigator to open it up in the Editor. You can then select the excel export icon in the top left-hand corner (Figure 33). The product system excel export wizard will open. The "Methods" section can be filled out or left blank. Select an export directory and click "Finish" (Figure 34). A folder with one excel file for Elementary Flows and one for Product Flows will be saved.

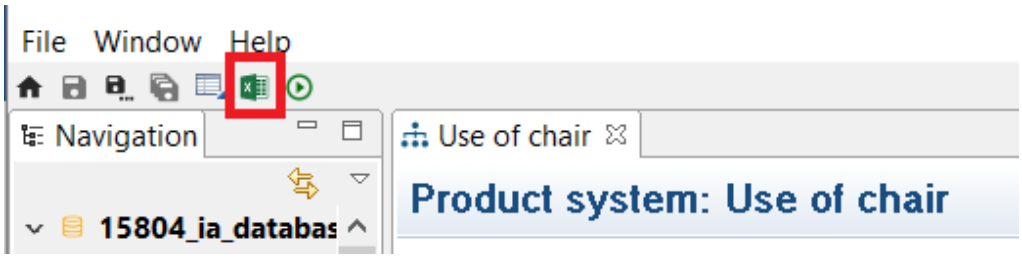


Figure 33: Exporting product system as excel file, step 1

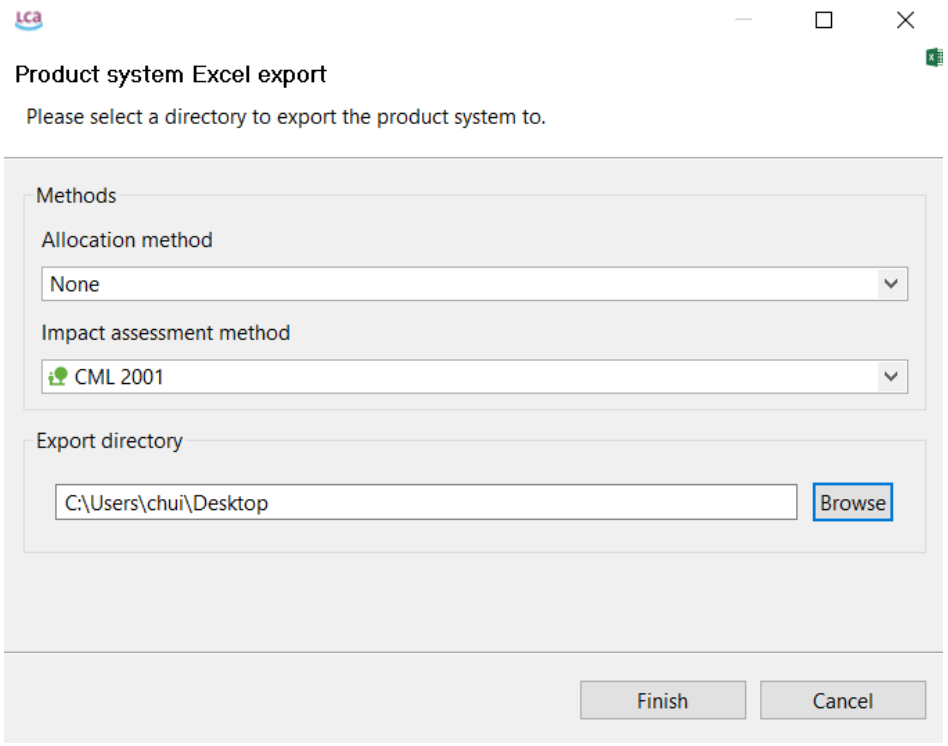


Figure 34: Exporting product system as excel file, step 2

4.7.3 Exporting data in ILCD ZIP-file format

openLCA can export the actors, flow properties, flows, LCIA methods, processes, product systems, sources and Unit groups in ILCD format. To do so, activate the database from which you would like to export. Then click on "File" → "Export". The export wizard will pop up. Select what database elements you would like to export in ILCD format. Then select a directory and the processes, flows, etc. to be exported and select "Finish".

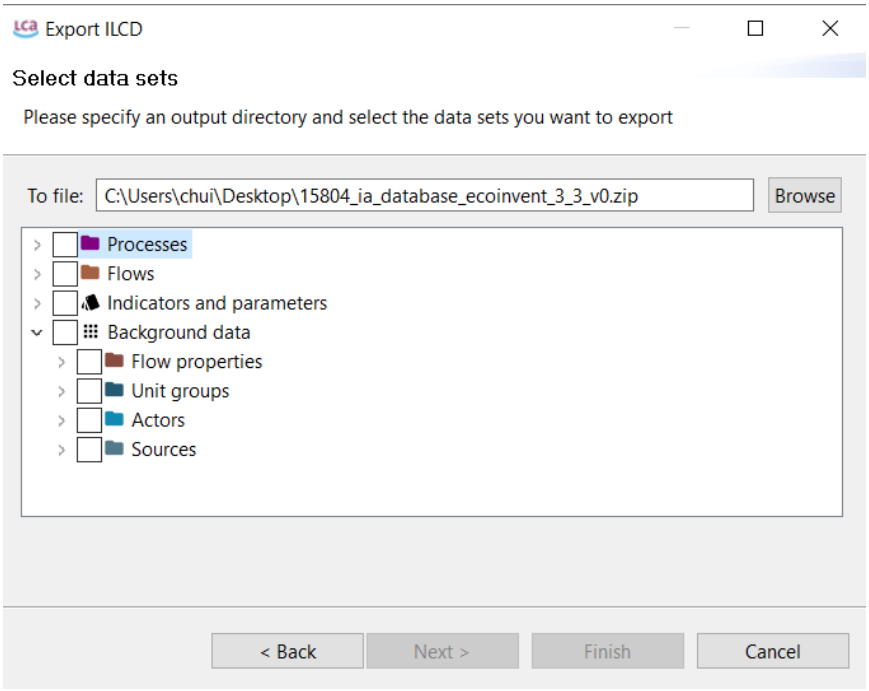


Figure 35: Exporting as ILCD, step 1

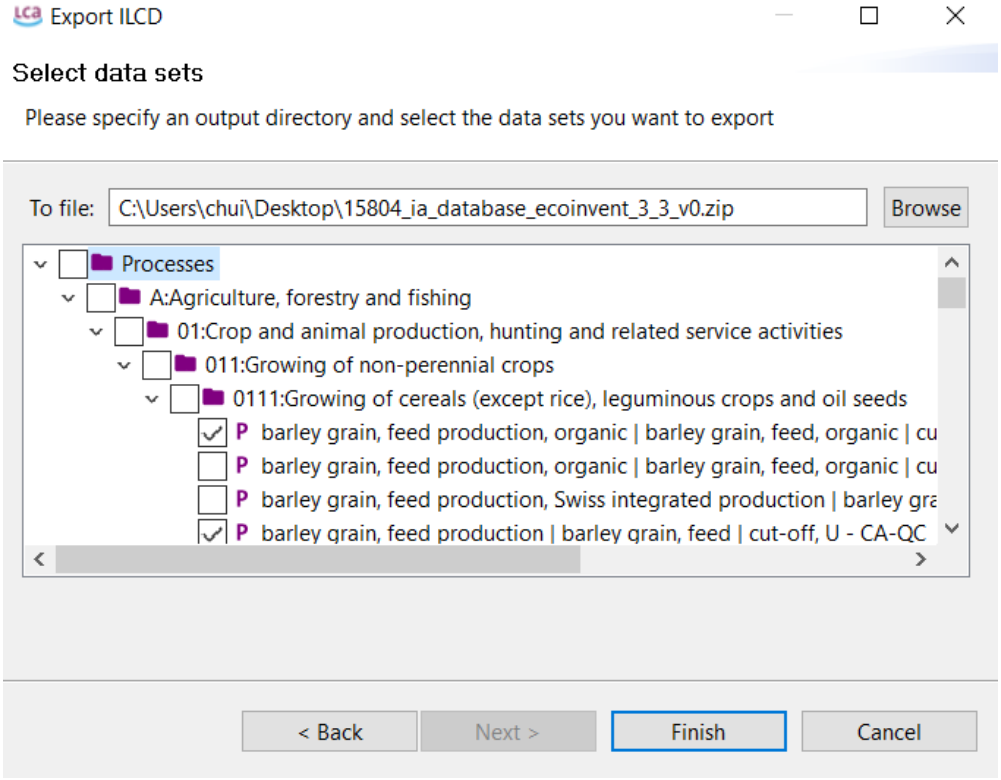


Figure 36: Exporting as ILCD, step 2

4.7.4 Exporting data as html file

openLCA can export Project results as in html format. To do so, calculate a project (see section 10 to find out how), then select the "Export report" icon when the Report Viewer is open in the Editor (Figure 37). Then simply select a directory and click "Okay".

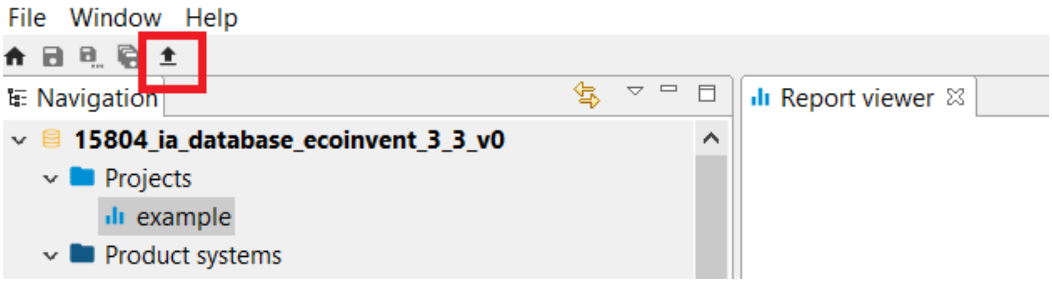


Figure 37: Exporting Project results as html file

4.7.5 Exporting data as CSV-Matrix

openLCA can export the Graph of a product system in CSV-Matrix format. To do so, open the product system you would like to export, then click on the "Matrix export" icon on the top left-hand side (Figure 38). The export wizard will appear. Here you can select the desired decimal and column separators as well as the file destinations for the technology and intervention matrix files (Figure 39). The files will be saved in excel format.

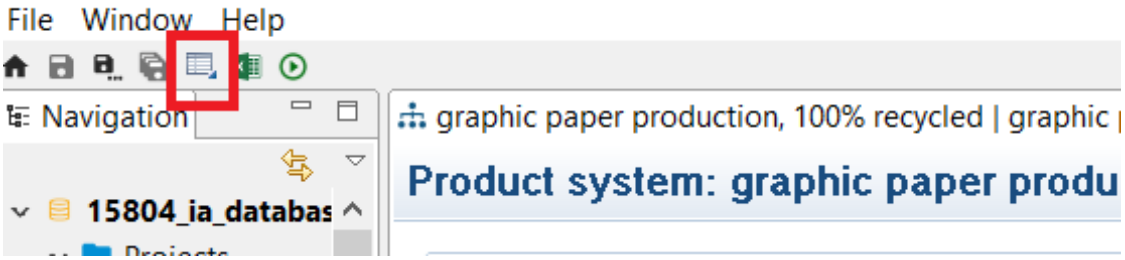


Figure 38: Exporting project results in SimaPro CSV format, step 1

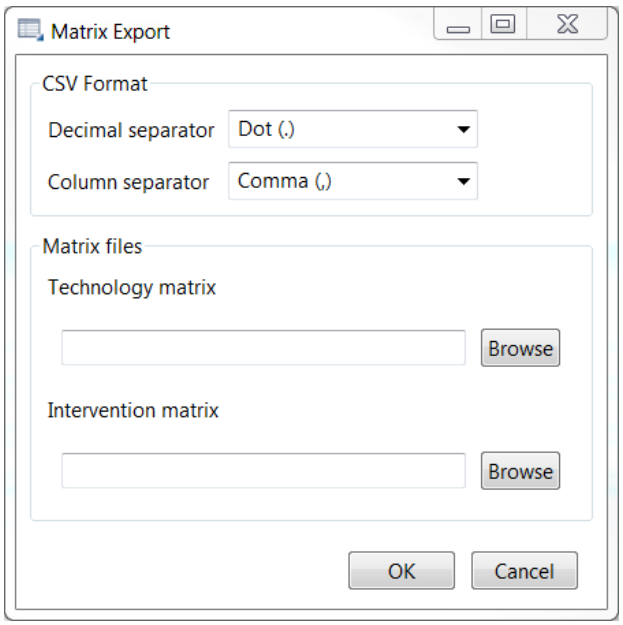


Figure 39: Exporting project results in SimaPro CSV format, step 2

4.7.6 Export data as JSON-LD file

Databases can be exported in JSON-LD format. To export these file types, activate the database from

which you would like to export processes/impact assessment methods. Then click on "File" --> "Export"(Figure 40). The export wizard will pop up. Select "JSON-LD" to export processes as JSON-LD files (Figure 41). In the next window select the elements to be exported and click "Finish" (Figure 42).

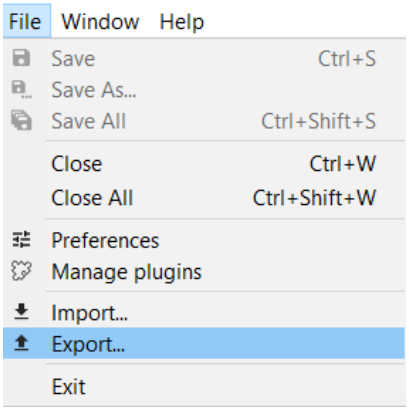


Figure 40: Exporting as JSON-LD file, step 1

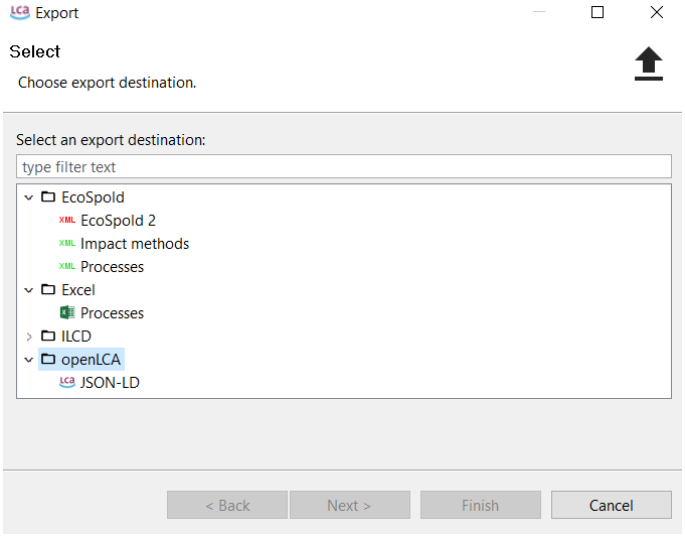


Figure 41: Exporting as JSON-LD file, step 2

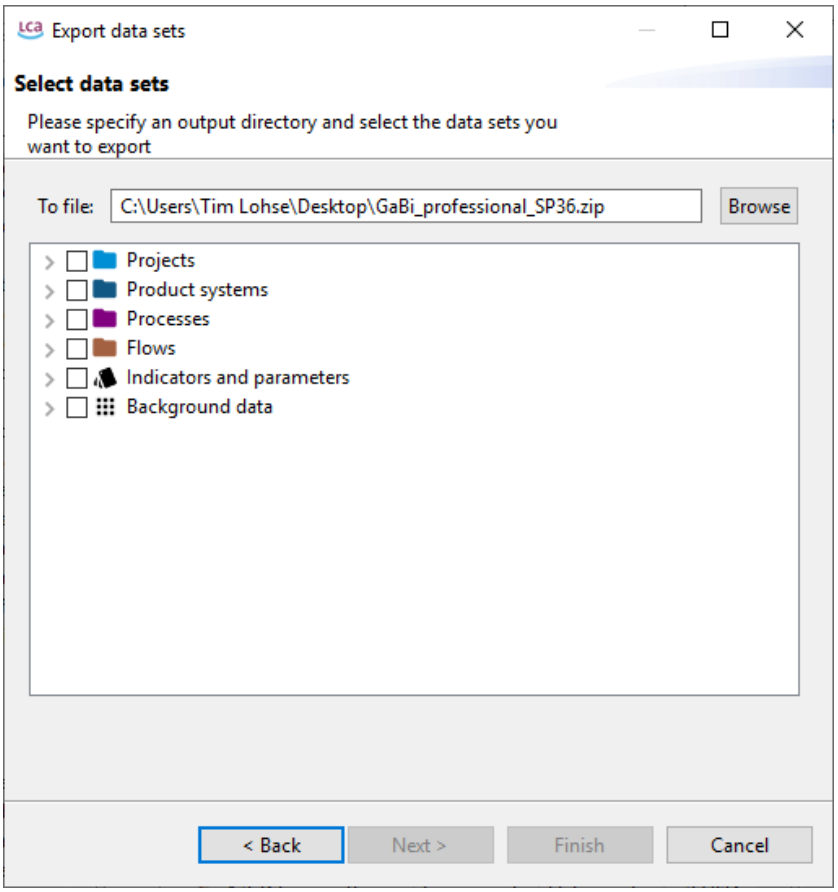


Figure 42: Exporting as JSON-LD file, step 3

4.7.7 "Copy" function for all openLCA tables

The information presented in all openLCA tables can be copied from the openLCA editor and pasted elsewhere (e.g. in excel, notepad, etc.). To select the columns and rows you would like to copy, select one cell, hold the "Shift" button on your keyboard and then select another cell in the table. All rows and columns in between will be marked. Then you can either right-click and select "Copy" or use the "ctrl/c" function on your keyboard to copy the table. Figure 43 shows an example from a process in the ELCD database (available for free at openLCA Nexus).

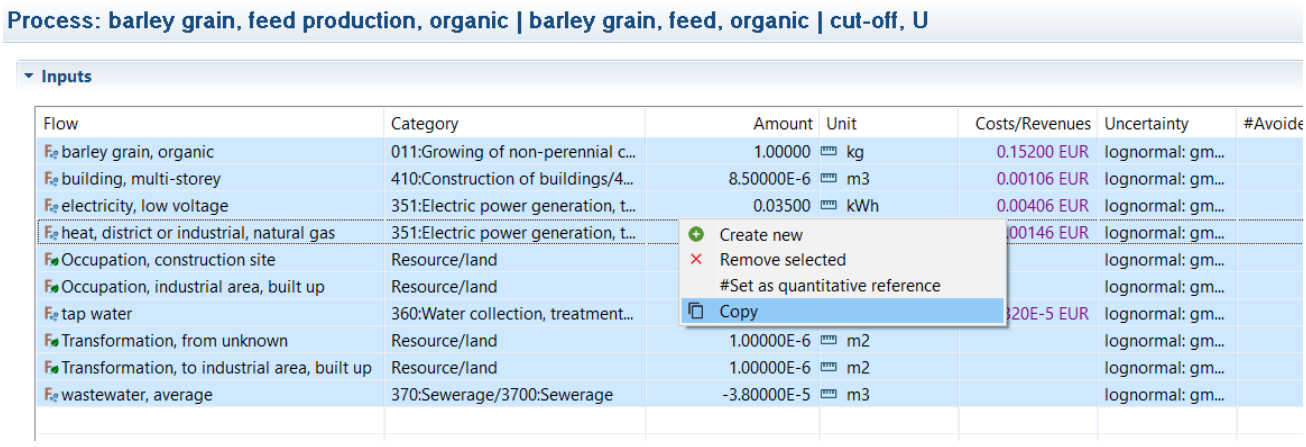


Figure 43: Copying data from openLCA tables

4.8 Link with Collaboration Server

A link between openLCA and a Collaboration Server is established to facilitate group work and establish a multi-user environment, which means that different users should be able to work with the same database and that quality assurance (e.g. tracking of changes) is needed. The Collaboration Server has the following main functions:

- User rights management system
- All changes are documented
- Versioning allows roll-backs
- A diff tool shows where data differs before accepting changes

A comprehensive manual for the Collaboration Server can be found here. An exemplary case (Figure 44) is presented to show how this Collaboration Server works.

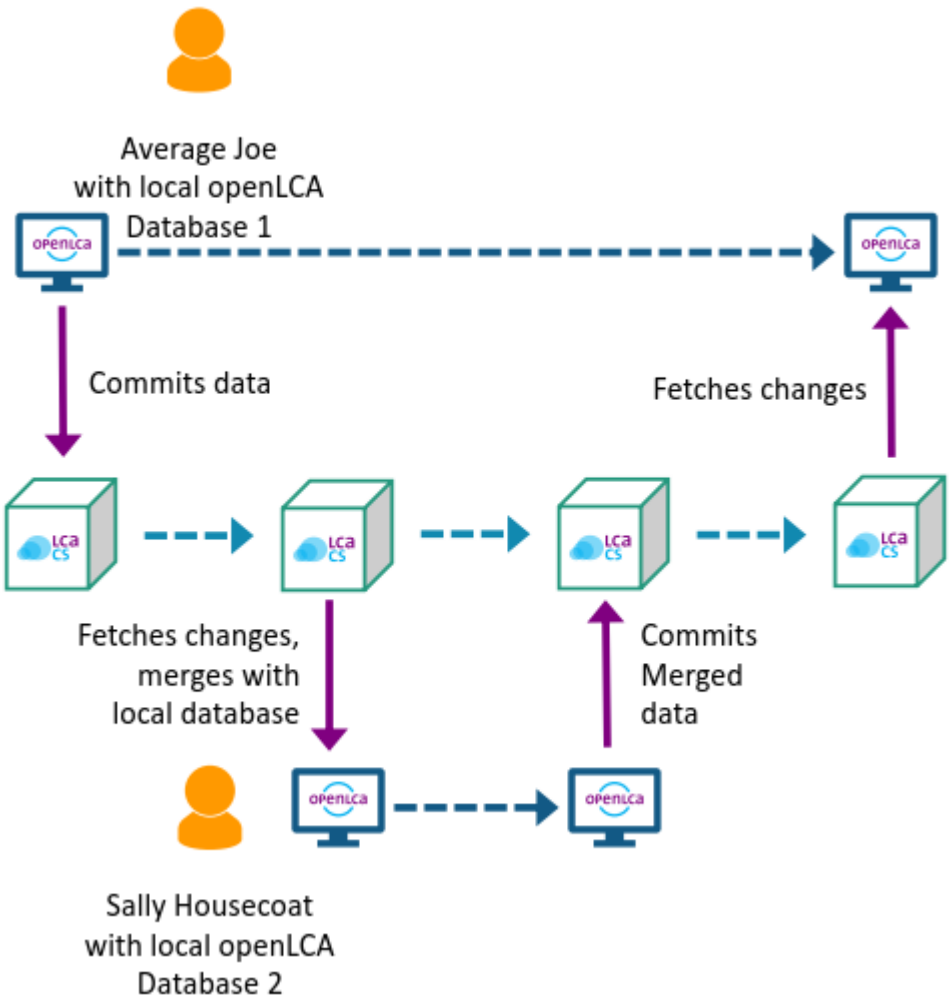


Figure 44: Collaboration Server, exemplary use case

By right-clicking on the active database it is possible to connect to a repository where User 1 and User 2

are members, see Figure 45 .

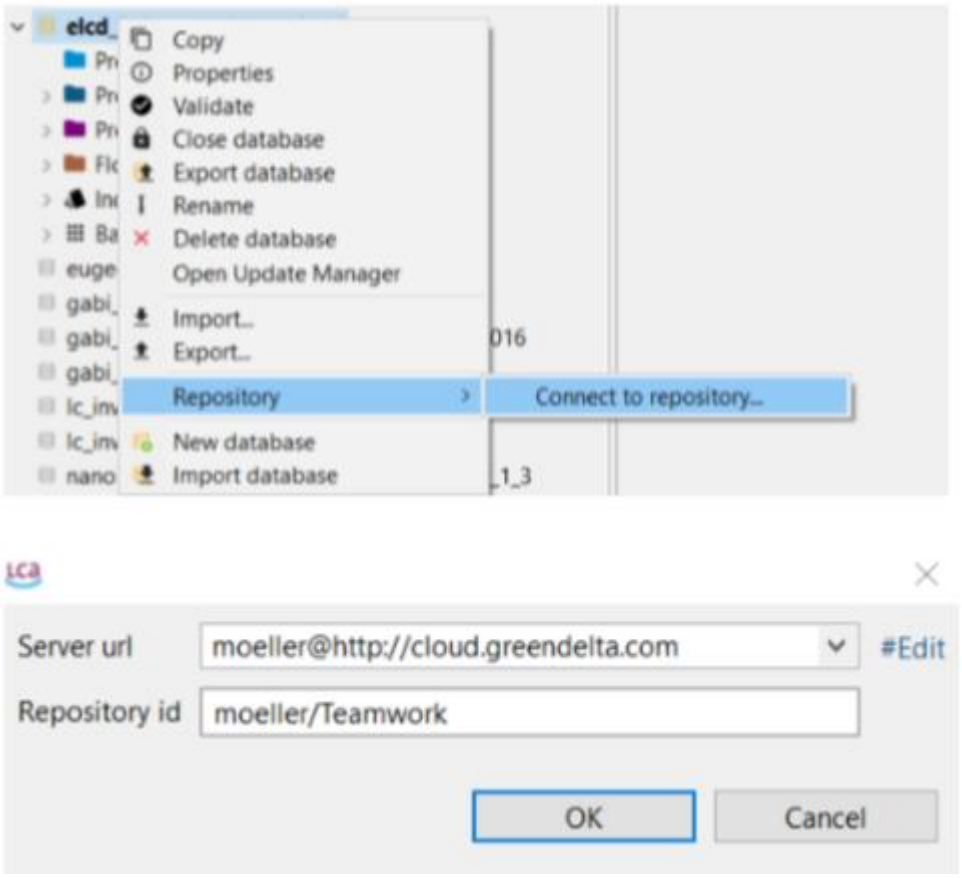


Figure 45: Collaboration server – connecting local database to repository

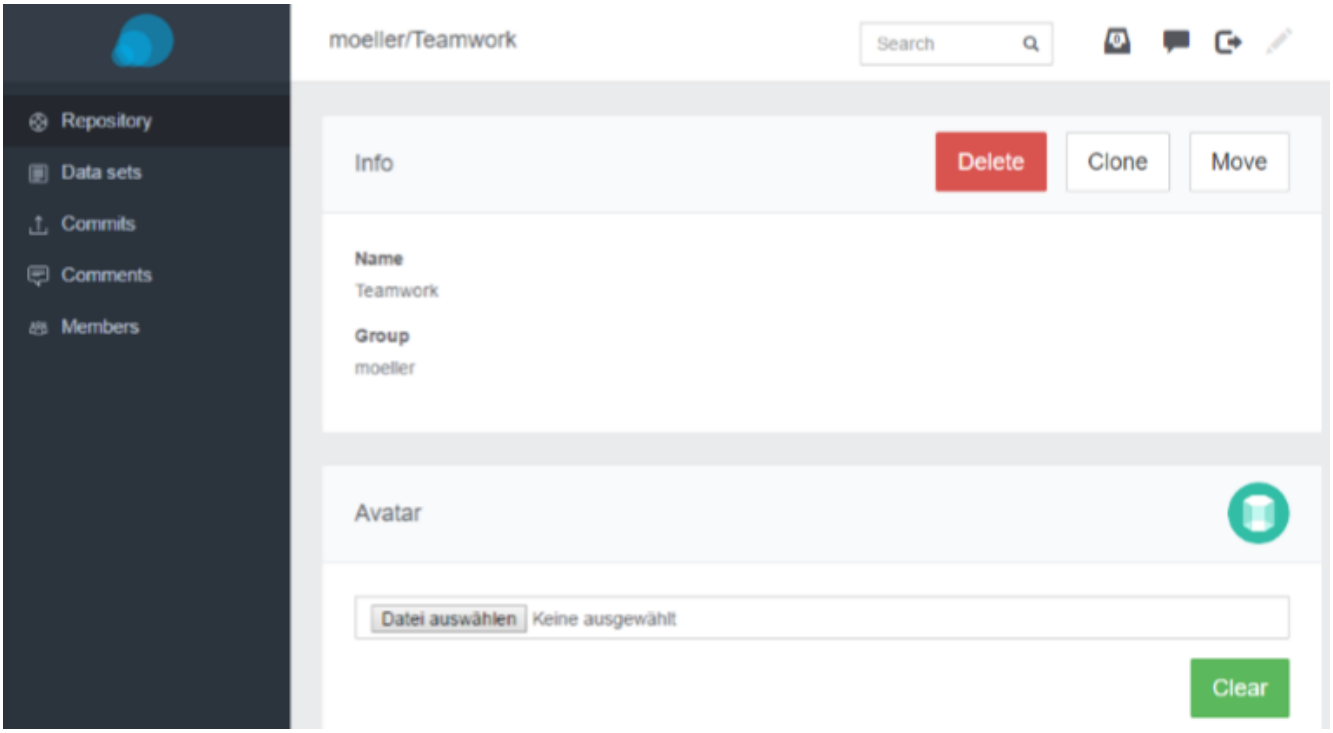


Figure 46: Collaboration Server - User 1 connects locale database to repository

At this point User 1 can commit data changes to repository and the data appear on the Collaboration

Server.

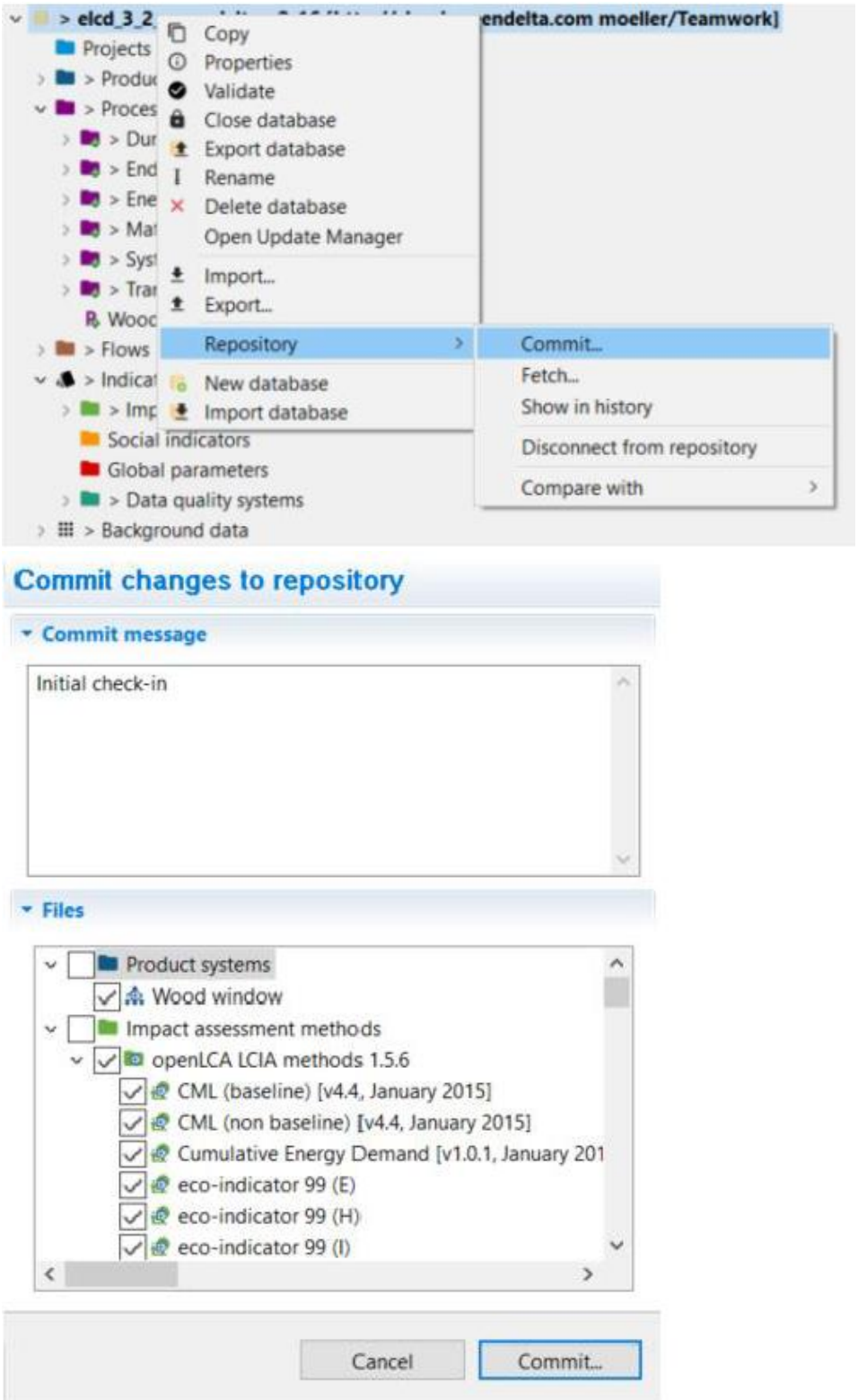


Figure 47: Collaboration Server - User 1 commits changes to repository

Now User 2 can connect to the same repository and fetch data. A summary of the differences to the local database appears.

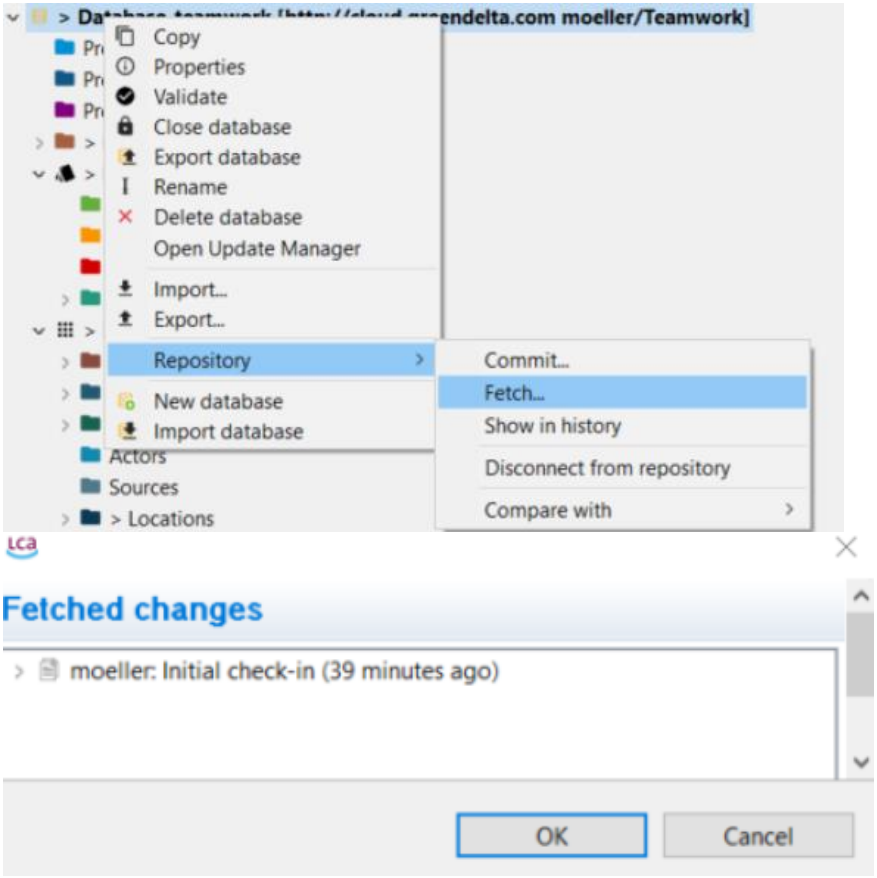


Figure 48: Collaboration server - User 2 connects to the same repository and fetches data

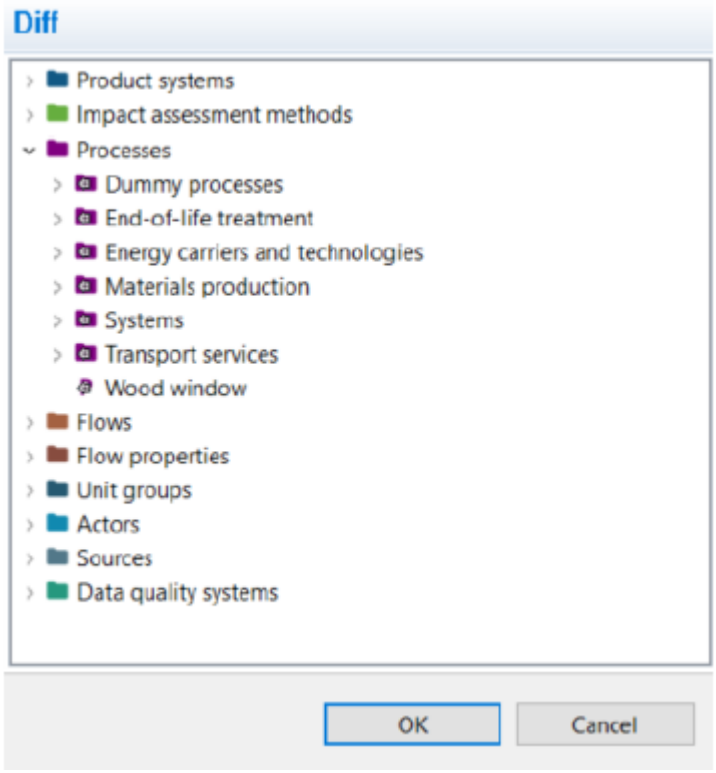


Figure 49: Collaboration server - summary of differences to local database

User 2 can make changes in local data and commit again to repository and at this point User 1 must fetch

changes from User 2 before committing changes in local database. The diff tool shows the differences in the data between local and remote model.

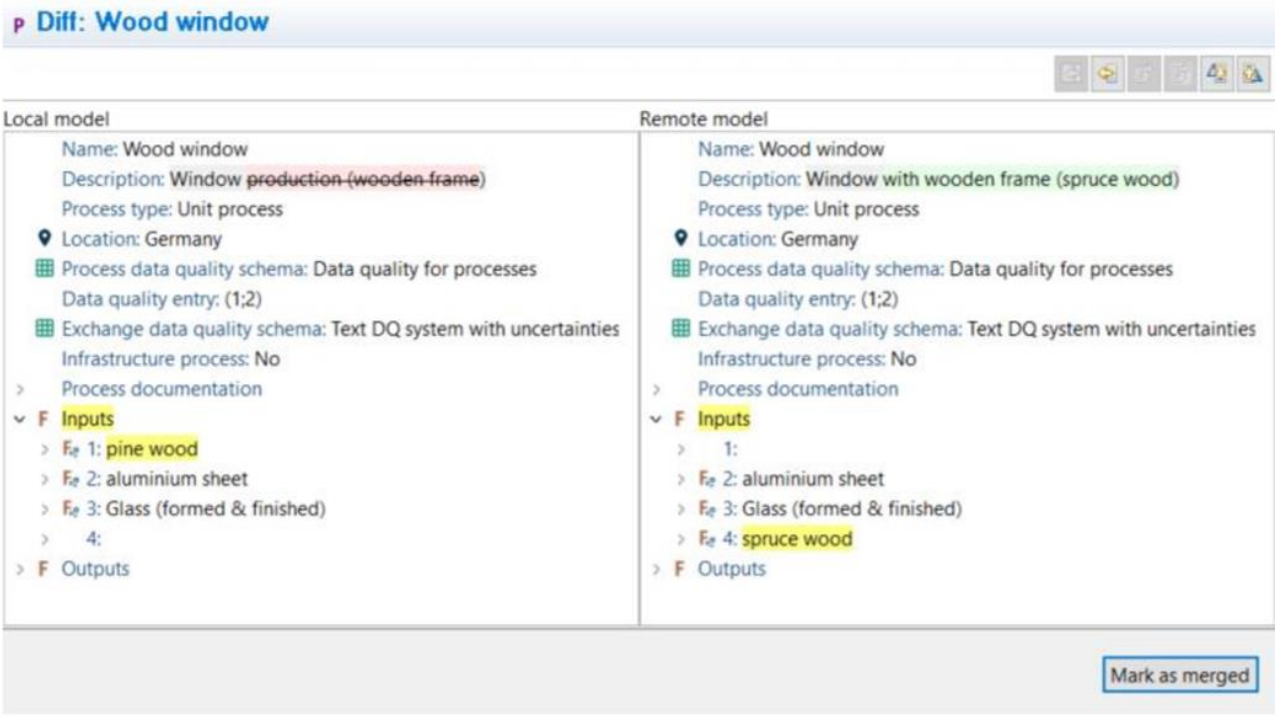


Figure 50: Collaboration server - Diff tool

It is finally possible to show the history of changes both in the web app (just click on “Commits”) and in openLCA (click on “Window”, “Show views”, “other” and then select “Commit history”), see Figure 51 and Figure 52.

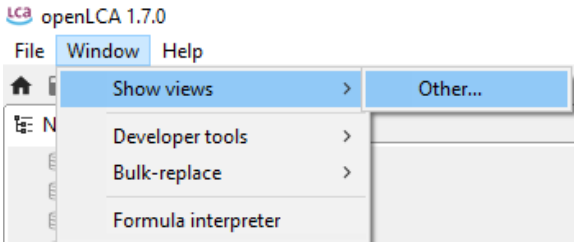


Figure 51: Collaboration server - Commit history in openLCA, step 1

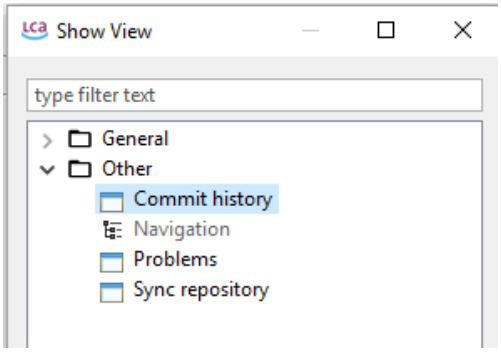


Figure 52: Collaboration server - Commit history in openLCA, step 2

5 Flows

5.1 Creating a new flow

To create a new flow, right-click on the “Flows” folder and select “New flow”. Name flow and define flow type and reference flow property, then click “Finish“. A new flow window will open in the editor (Figure 53).

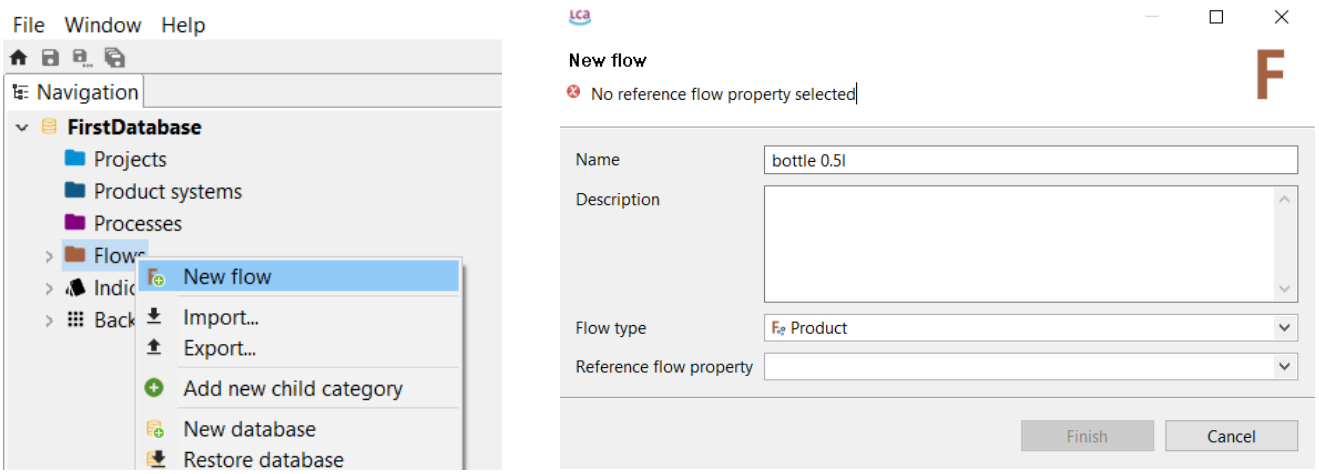


Figure 53: Creating a new flow

It is possible to define if the flow type is a product, an elementary flow or a waste flow. Furthermore, a reference flow property must be defined (e.g. duration, energy, volume, etc.)

5.2 Flow tab contents

5.2.1 General information

In the General information tab, you can see and change the name and add a description. It is also possible to add additional information such as a CAS number, formula and location. Under "Used in Processes", you can see which process consume the flow and which produce the flow. If you double-click on a process under "Used in processes", it will open in the editor.

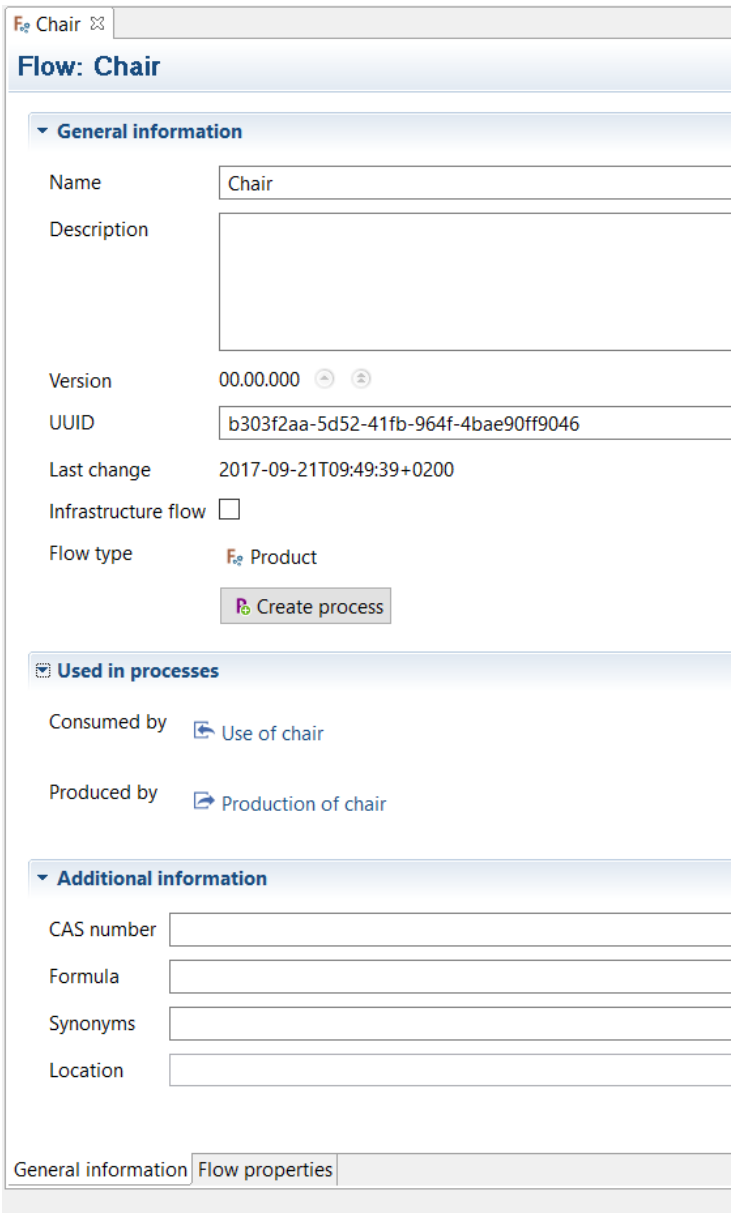


Figure 54: Flow editor - General information tab

5.2.2 Flow properties

Additional flow properties can be added in the “Flow properties” tab. Make sure to type in the correct conversion factor and to select the correct flow property as the reference when multiple flows are listed.

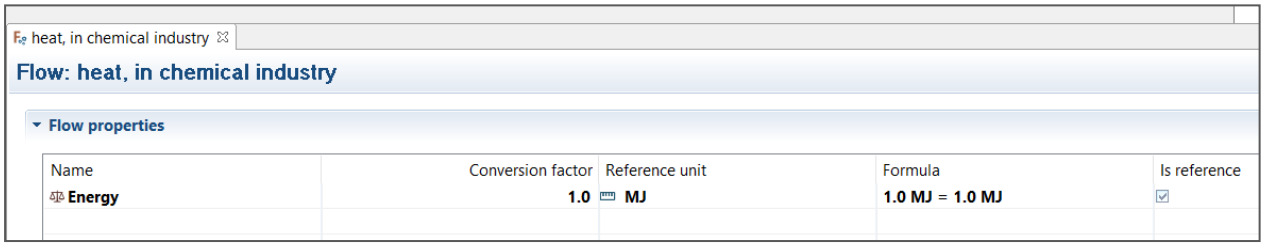


Figure 55: Flow editor - Flow properties tab

5.3 Waste modelling with the actual flow direction

One of the most important new features in OpenLCA since version 1.7.0 is the possibility to create a waste treatment process from a waste flow which results then as an input process.

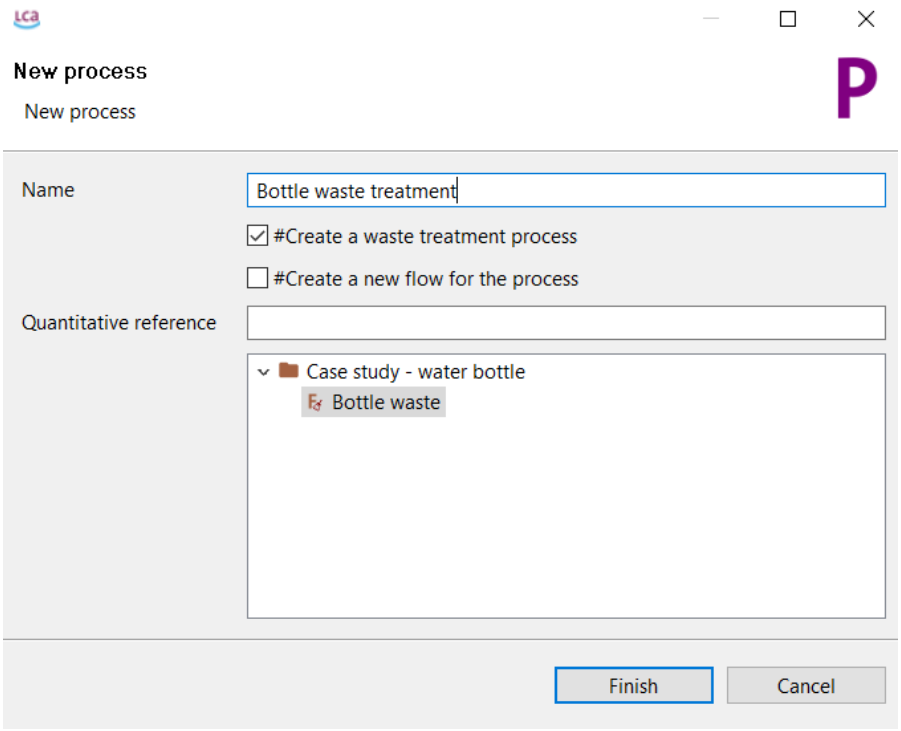


Figure 56: Creation of a waste treatment process

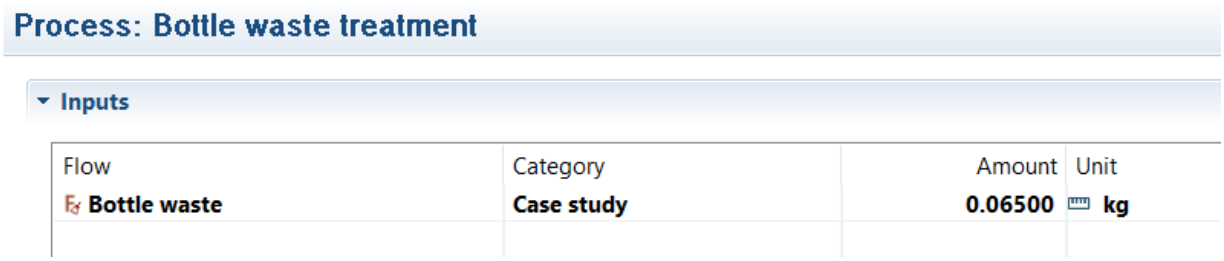


Figure 57: Waste flow as input in a waste treatment process

Furthermore, when applying system expansion, it is possible to mark an input waste flow as an avoided waste, Figure 58. This means an expense in the Cost column because the flow is considered as an output, as also shown in grey colour in the model graph of the product system (Figure 59).

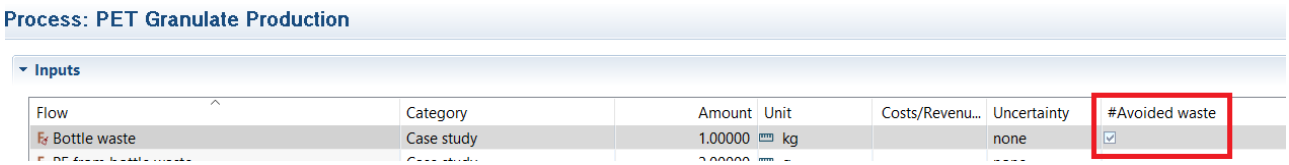


Figure 58: Waste input flow marked as avoided waste

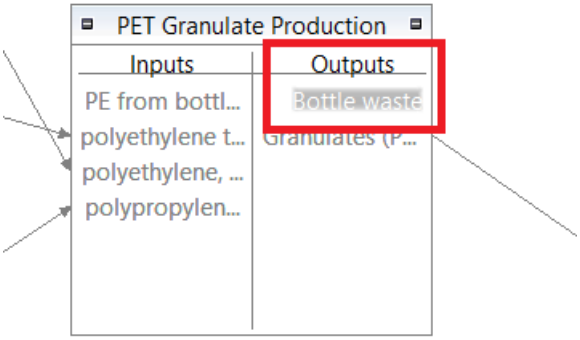


Figure 59: Product system, model graph – System expansion: waste flow as avoided waste on the output side of a process.

However, it is still possible to model waste flows as in the previous openLCA versions (below 1.7.0), which means considering the waste treatment as a "service" for the process to eliminate the product. In this case the waste flow is an output in the waste treatment process and the waste treatment process is an input in the process considered. Attention must be paid to the positive or negative values to be inserted in the Amount column.

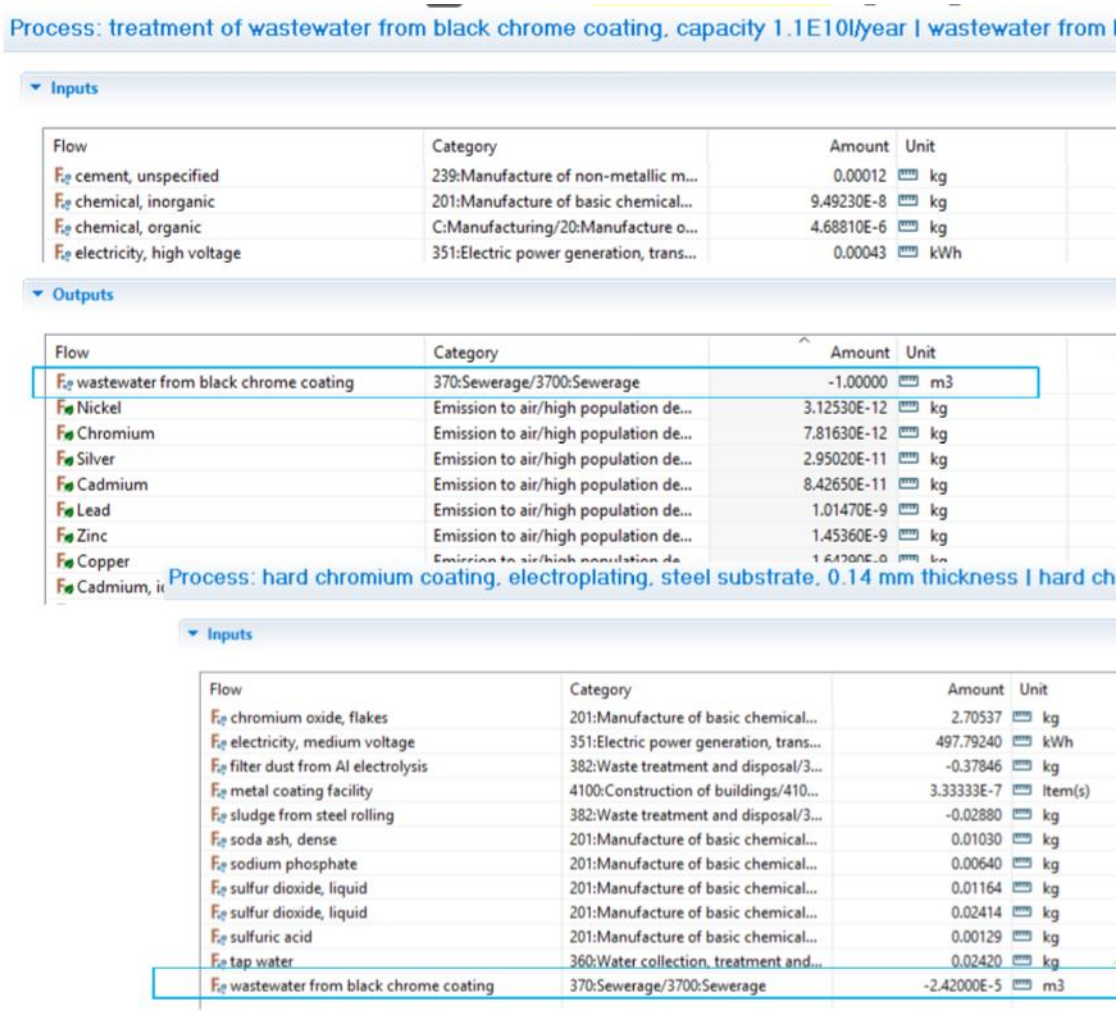


Figure 60: Waste modelling. Waste treatment as a service for the process

6 Processes

Processes can be either “unit processes” or “system processes”. A process is an activity that transforms an input into an output. The simplest form of a process in openLCA is a unit process. In the left half of Figure 61, each process from A to G is a unit process.

An aggregated process, as visible in the right half of Figure 61, is called a system process in openLCA. It is an aggregated life cycle result which is saved as a process for easy usage.

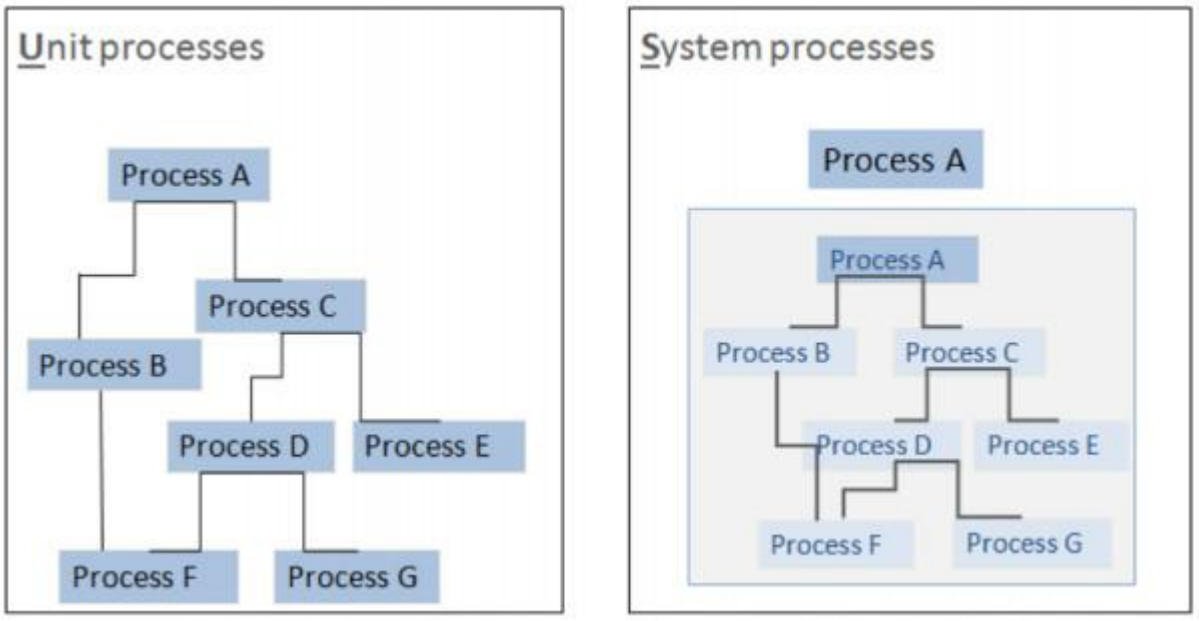


Figure 61: Difference between unit process and system process

Figure 62 shows how unit processes and system processes are displayed in the navigation window.

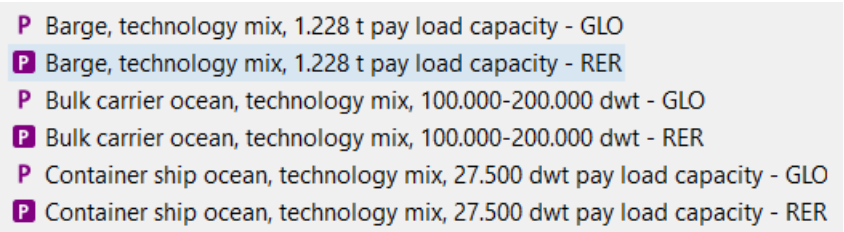


Figure 62: Unit processes (purple font colour) and system processes (white font colour on purple background)

6.1 Creating a new process

To create a new process, right-click on the “Processes” folder and select “New process”. Name the process and select a quantitative reference (the reference output of this process). It is also possible to create a new product flow for the process. The product flow will automatically be given the same name as the process. Once "Finish" is clicked, the new process will open in the editor.

An input flow or any elementary flow can also be used as a quantitative reference.

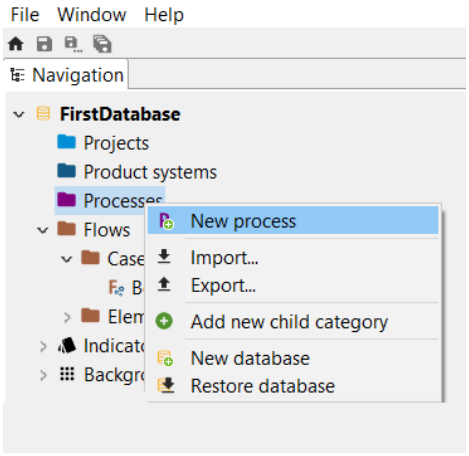


Figure 63: Creating a new process, step 1

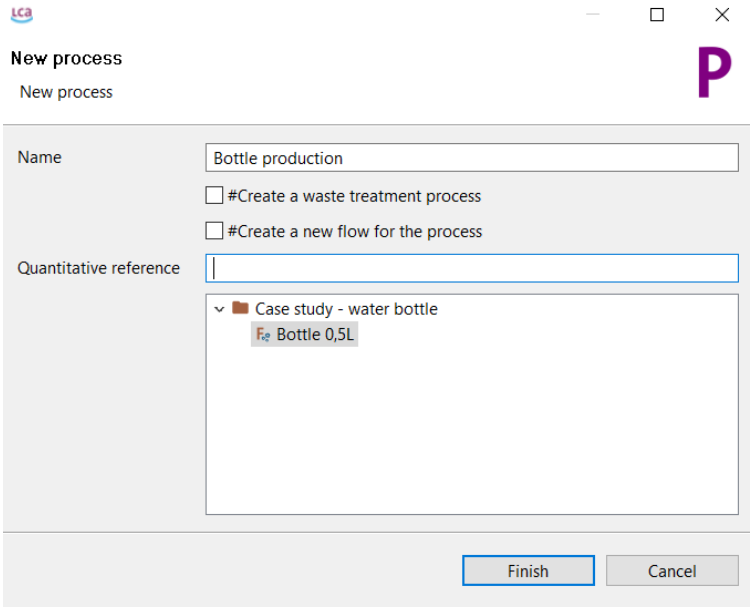


Figure 64: Creating a new process, step 2

Furthermore, it is possible to create a waste treatment process selecting a waste flow previously created as quantitative reference (the reference input of this process).

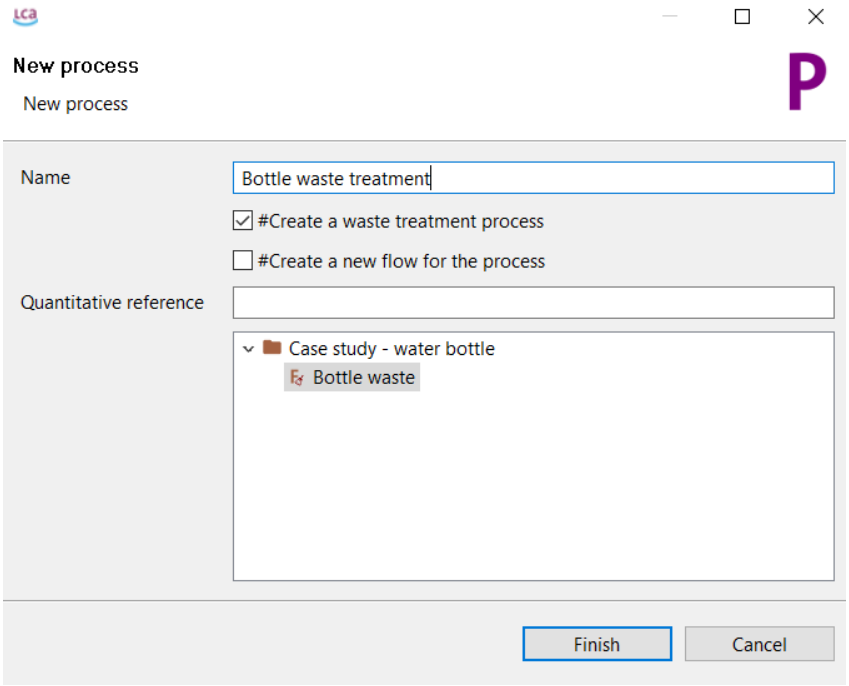


Figure 65: Creation of waste treatment process

6.2 Process tab contents

6.2.1 General information

In the General information tab of a process you can change the name, add description, set the quantitative reference, and add time, geography, technology and data quality information. If the database provider attached any images in their process description, the image will be visible here. It is also possible to create a product system from here (to find out how see section 8.1).

6.2.2 General information - Data quality

The last section in the General information tab is about data quality. It is possible to select a data quality system for the process, flows and social aspects. Data quality entry for the process must be defined in the General information section, Figure 66. On the other side, data quality information for flows (Figure 67) and social assessment needs to be selected in “Inputs/Outputs” (see section 6.2.3) and “Social aspects” (see section 6.2.8) respectively.

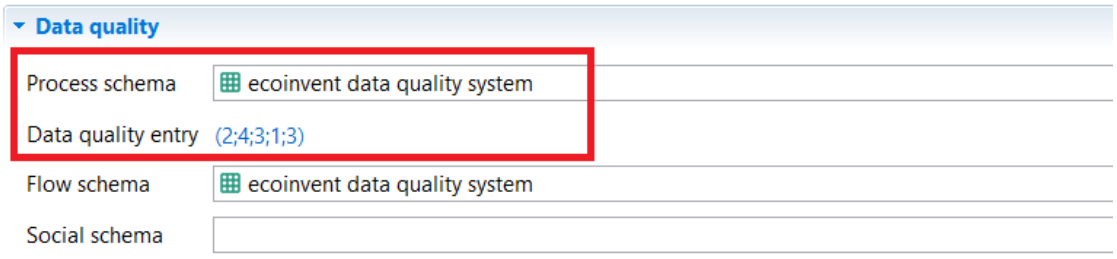


Figure 66: Process tab - General Information, data quality

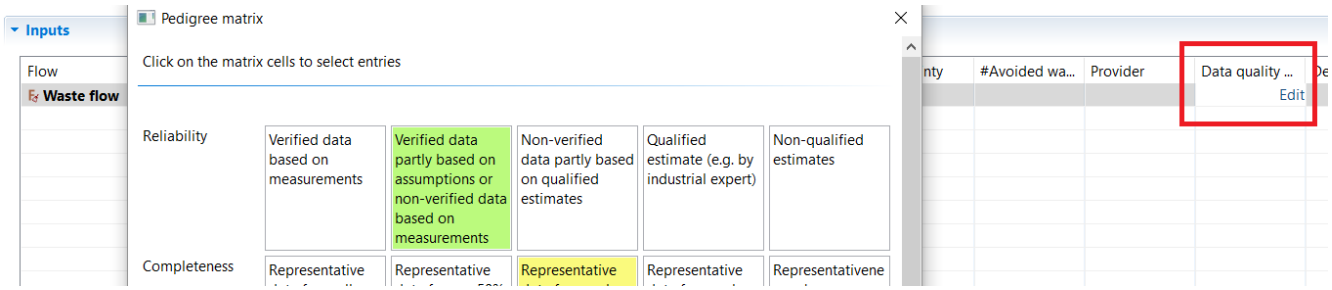


Figure 67: Process tab - Inputs/Outputs, Data quality for flows

Data quality system can be selected among the systems available in the "Data quality systems" directory in the "Indicators and parameters" section of the active database.

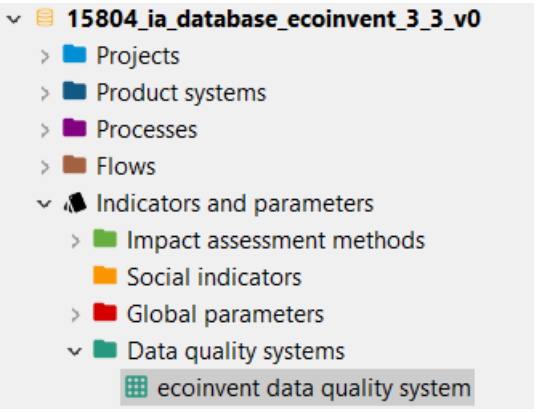


Figure 68: Data quality systems directory in an active database

If you double-click on an existing data quality system (e.g. ecoinvent data quality system) a "General information" tab is displayed. Scores for different indicators are shown as well as uncertainty values to be assigned to the previous indicators, see Figure 69.

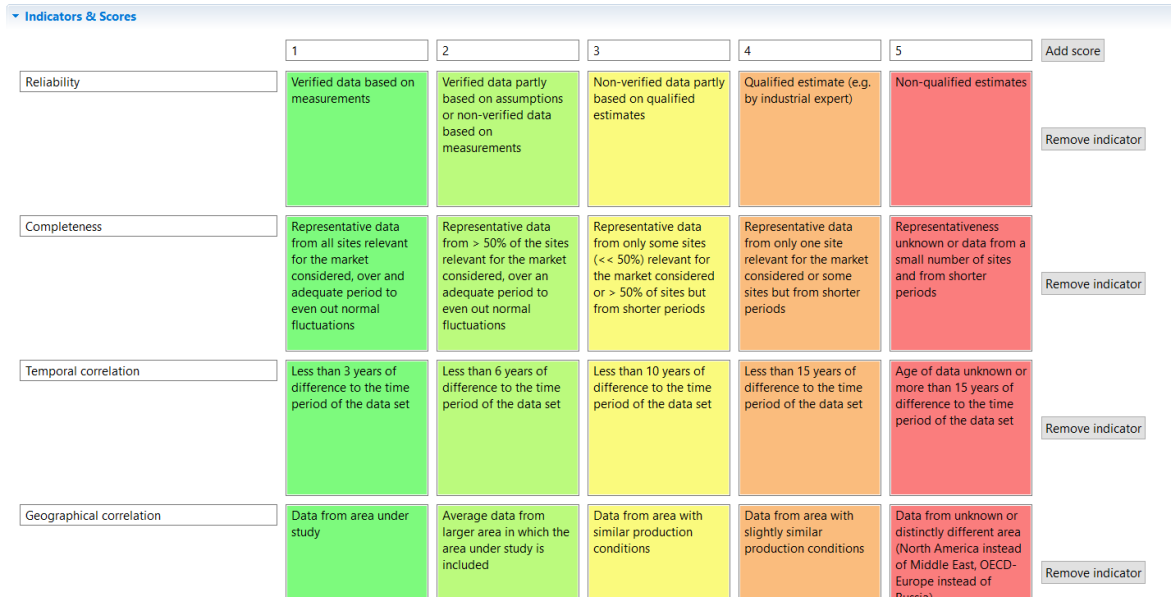


Figure 69: Data quality systems, indicators and scores

It is also possible to create a new database if you right-click on the "Data Quality Systems" directory and

select “New data quality system”. It is then possible to add indicators, scores and uncertainties.

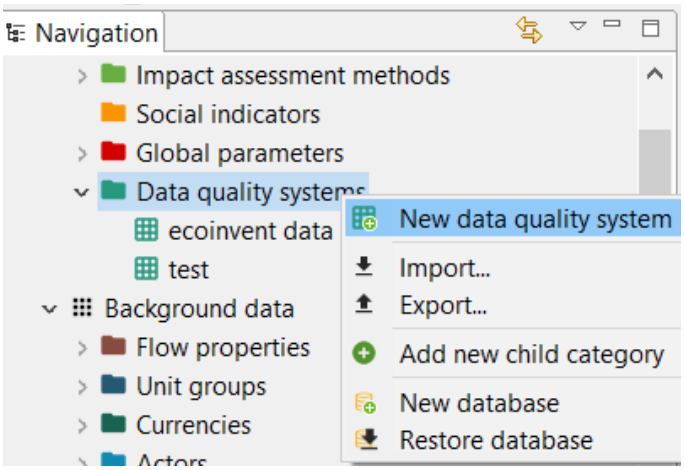


Figure 70: Create new data quality system, step 1

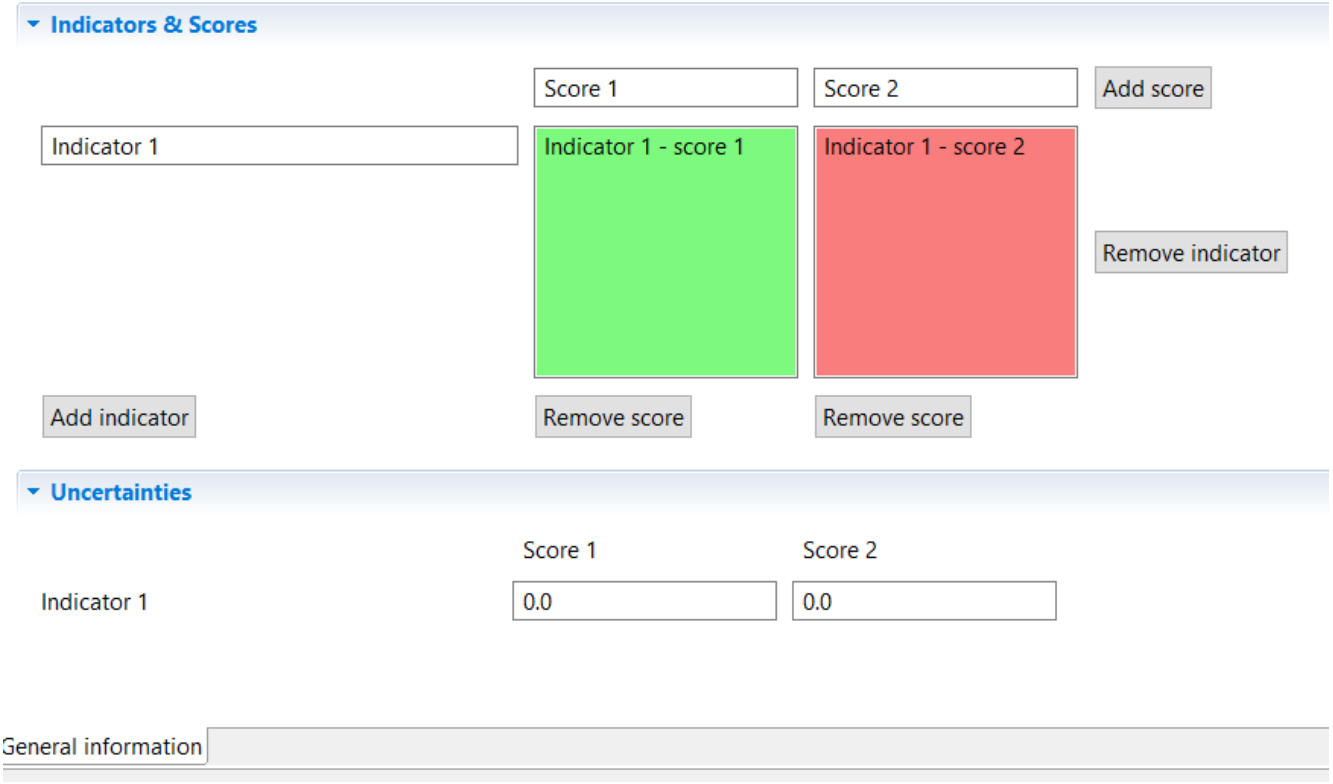


Figure 71: Create new data quality system, step 2

6.2.3 Inputs/Outputs

Here product, elementary and waste flows are listed as inputs/outputs. Information on respective categories, amounts, units, cost/revenues, uncertainty, avoided waste, provider, data quality and description are also contained in these tables. It is also possible to change units by clicking on the unit cell and selecting a new unit from the list.

Amounts can be typed in as values, formulas and/or parameters. Complex formulas require a certain

format (e.g. Tan(a), trunc(c), etc.). Use the formula interpreter to find errors (available under "Window" → "Formula interpreter"). When a formula and/or parameter is typed in the "Amount" field, the software will calculate the value for the amount automatically. To see the calculated value, click on this button: . To see the original formula/parameters, click on the button again. Under "Default provider" you can select a specific flow provider (when more than one provider for a specific product flow exists). Prices can be typed for each flow in input and output sections and they are displayed in green colour if they are revenues or in violet colour if they are expenses.

To navigate to the provider of a flow. Simply right-click on a flow and select Open provider.

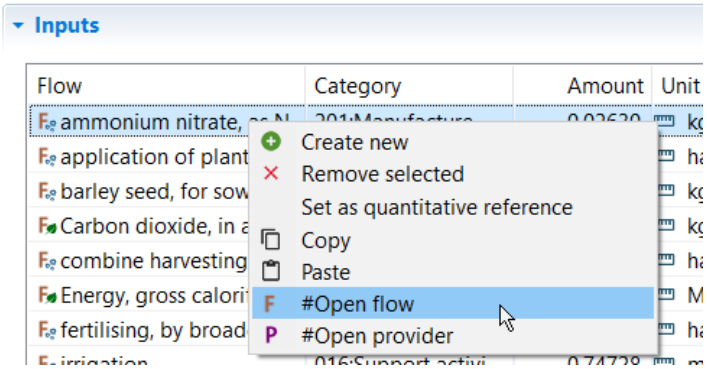


Figure 72: Flow provider

6.2.4 Administrative information

This section is quite self-explanatory. Entries to administrative information do not affect calculations.

6.2.5 Modelling and validation

Here it is possible to set the process type (as unit or system process) and add information about the dataset. Add reviewers by clicking the "Add actors" icon in the section "Process evaluation and validation". If the actor you would like to add is not included under "Actors", you can add an actor by right-clicking on the "Actor" folder in the Navigation and selecting "New actor". To add a source, click on the green "+" icon in the section "Sources". Once again, if the source you require is not listed, you can add a new one by right-clicking on the "Sources" folder in the Navigation and selecting "New source".

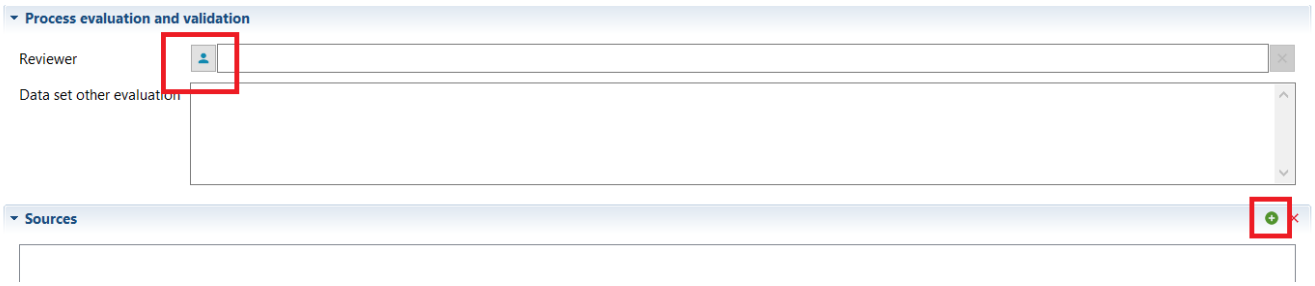


Figure 73: Adding a reviewer or source to processes

6.2.6 Parameters

Parameters can be used on the process, Impact assessment method, product system, project and database levels. Parameters can be used instead of concrete values for inputs/outputs. They can be defined as simple values, formulas or complex functions. Parameters can overwrite each other (e.g. the value set for a parameter in a process can be overwritten on the product system/project levels).

There are some rules when it comes to parameters. Parameter names cannot contain special characters or more than 1000 characters. Parameter formulas can contain single values, simple equations, or complex functions including logical expressions. Parameter formulas do not contain units, (so please add them in the description field) and cannot have more than 1000 characters. Theoretically, the number of parameters is unlimited. When you assign an amount to a parameter, use point (.) instead of comma (,) for the decimal numbers.

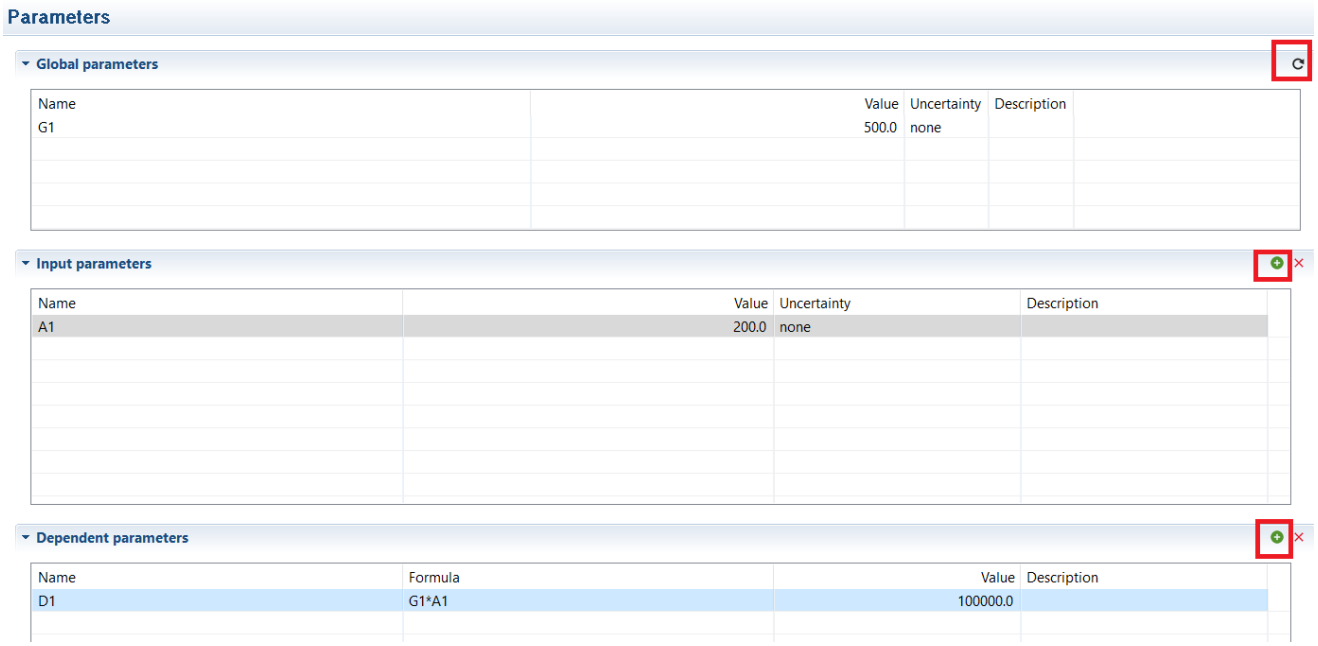


Figure 74: Global, Input and Dependent parameters

"Global" parameters can be found and are valid on all levels. "Input" parameters are parameters that are only valid for the process/LCIA method/Product system in which they are saved. "Dependent" parameters are parameters that include input or global parameters in their formula. See Figure 74 for an example.

New global, input and dependent parameters can be created within a process or impact assessment method. These are then also available in product systems and projects (note: it is not possible to create a new parameter on the product system or project levels). To create a global parameter, you can select “Global parameters” in the "Indicators and Parameters" section in Navigation and if you right click a tab for the creation of a new global parameter pops up. At this point you can enter name, description, type (if it is an input or a dependent parameter) and amount (see Figure 75). To load the global parameter just

created select the “reload” button in the “Global parameters” section in “Parameters” tab in a process or impact assessment method (see Figure 74).

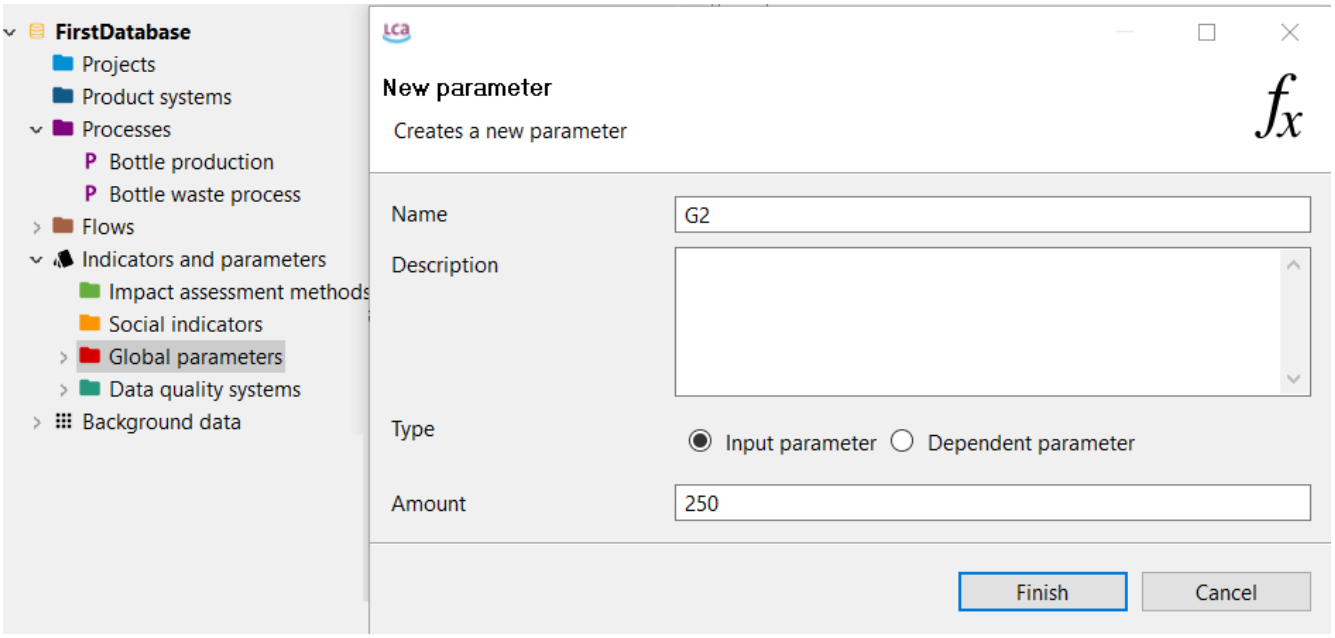


Figure 75: Creation of a global parameter

The use of parameters within a database can now be checked via the ‘usage view’ feature (right-click on a parameter and select “usage”).

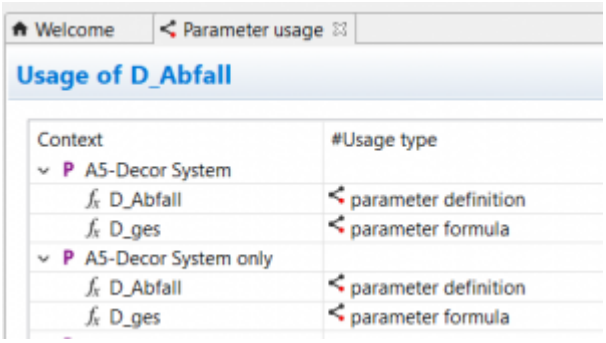


Figure 76: Parameter usage throughout the database

6.2.7 Allocation

Usually, life cycle assessment requires single-output processes, but this is not always the case. Multi-output processes (e.g. co-generation of heat and power, simultaneous milk, leather and meat production, etc.), however, occur frequently. These situations can be dealt with using two different strategies, namely through allocation (in which elementary flows and products from multi-output processes are mathematically divided into multiple processes) or system expansion (to avoid allocation).

There are 3 allocation methods in openLCA:

- physical allocation
- causal allocation
- economic allocation

The values (allocation factors) for physical, causal and economic allocation can be viewed/altered in the "Allocation" tab. For allocation to work, the main product and the co-products of the multi-output process need to have the same flow property. Select the "Calculate default values" button and the software will automatically calculate the values for all three allocation types. For physical allocation, the default (reference) flow property is used to calculate they physical ratio between the main product and the co-product on the bases of their amounts, e.g. in units of mass or energy. For economic allocation, the allocation factors are based on the economic value of product flows. Thus, to apply economic allocation, you first need to add an economic flow property to each product flow. Causal allocation can be applied by manually inserting the desired allocation factors in the causal allocation section.

Figure 77 illustrates allocation for wood production. The physical allocation factors are automatically calculated based on the ratio between the product outputs of the process as expressed in the reference flow property (e.g. kg). For this example, the output flows for bark and wood are 0.30 kg and 1kg, respectively, thus resulting in the physical allocation factors 0.23and 0.77. Economic allocation considers the market values of the output product flows. Prices are set at 0.40 $/kg and 1 $/kg for bark and wood. This results in the economic allocation factors you can see below. The causal allocation factors need to be inserted manually based on assumptions/prior scientific research on the relative impacts arising from each output, otherwise they are assumed to be those for physical allocation. In this case they are set to 0.4 and 0.6 for bark and wood respectively.

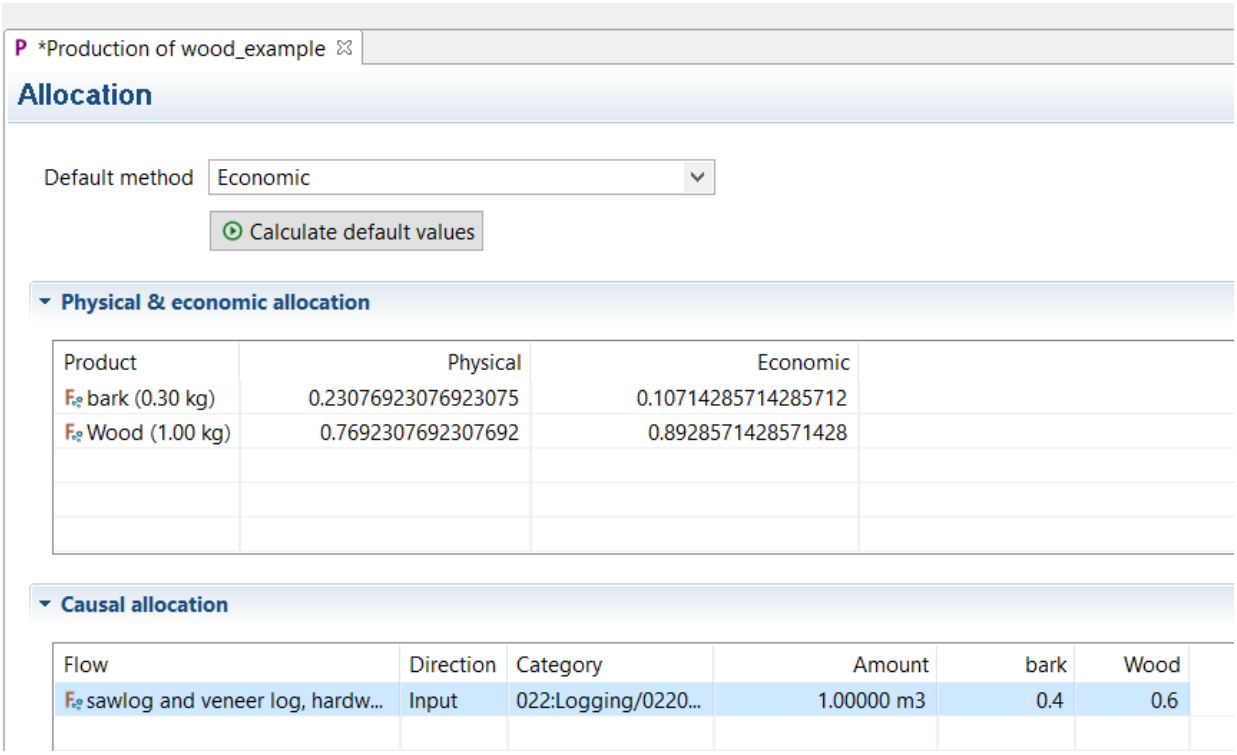


Figure 77: Physical, causal and economic allocation

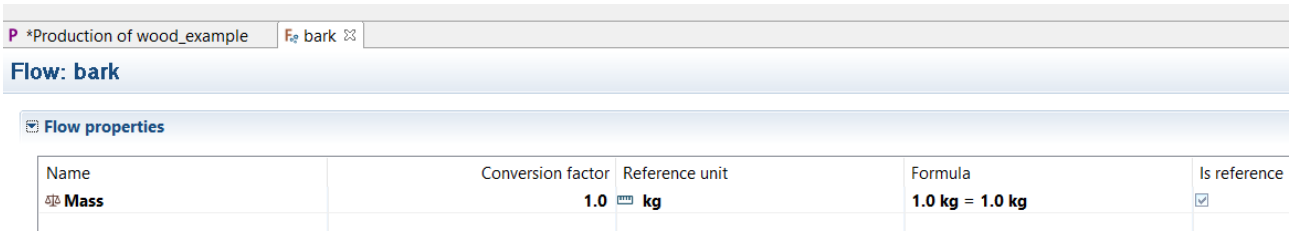


Figure 78: Flow property mass

To avoid allocation, system expansion can be performed by checking the box of "Avoided product" for the by-product, see Figure 79. It is important that a process providing the avoided product flow exists. The avoided product is then shown in grey in the model graph when creating the product system, see Figure 80 .

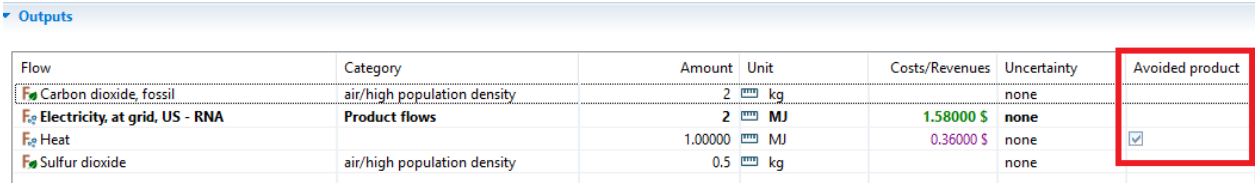


Figure 79: System expansion, marking an avoided product in a process

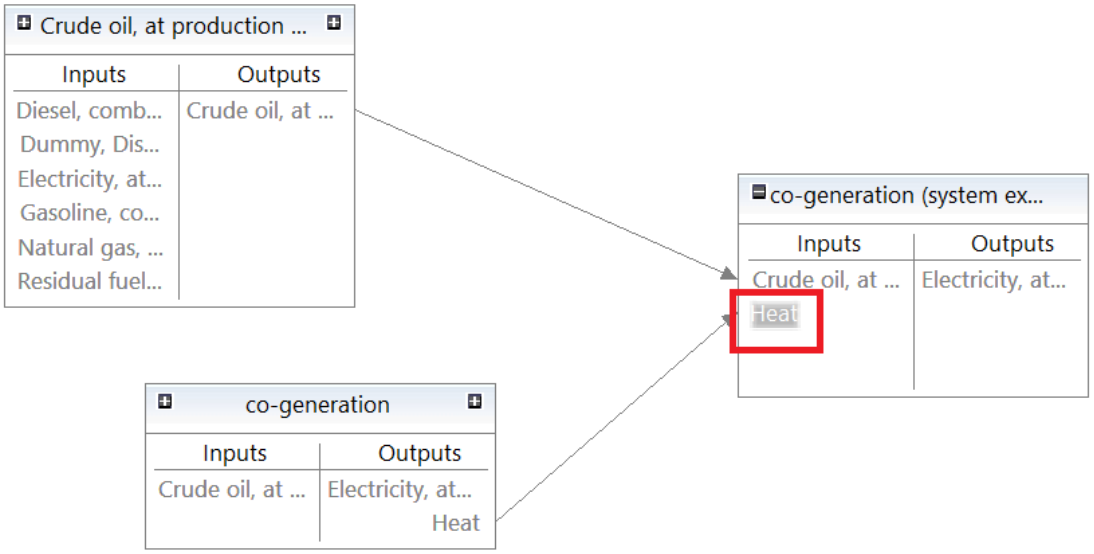


Figure 80: System expansion - avoided product in model graph

6.2.8 Social aspects

This section shows contents only if a database for the performance of Social LCA is active. Databases for Social LCA (e.g. Soca or Psilca) are available in OpenLCA Nexus website (https://nexus.openlca.org/), in the Database section. Social indicators are available in the “Social Indicators” directory in the “Indicators and parameters” section in the active database.

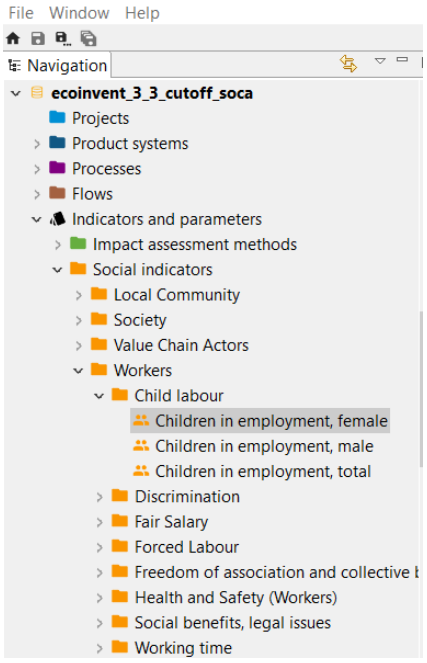


Figure 81: Social indicators in "Indicators and parameters" directory

To open one social indicator, you can double-click on it. General information, additional information about unit of measurement and evaluation schema and activity variable (name, quantity and unit) are displayed.

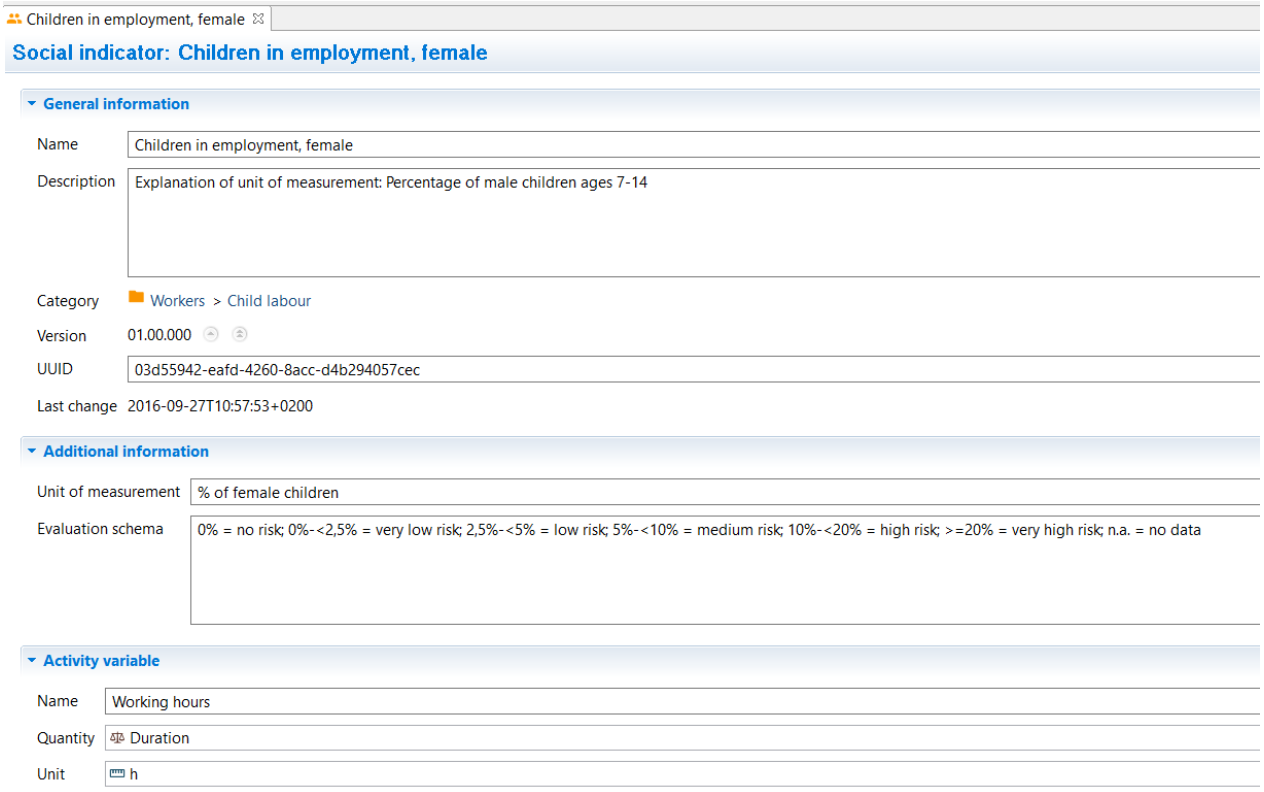
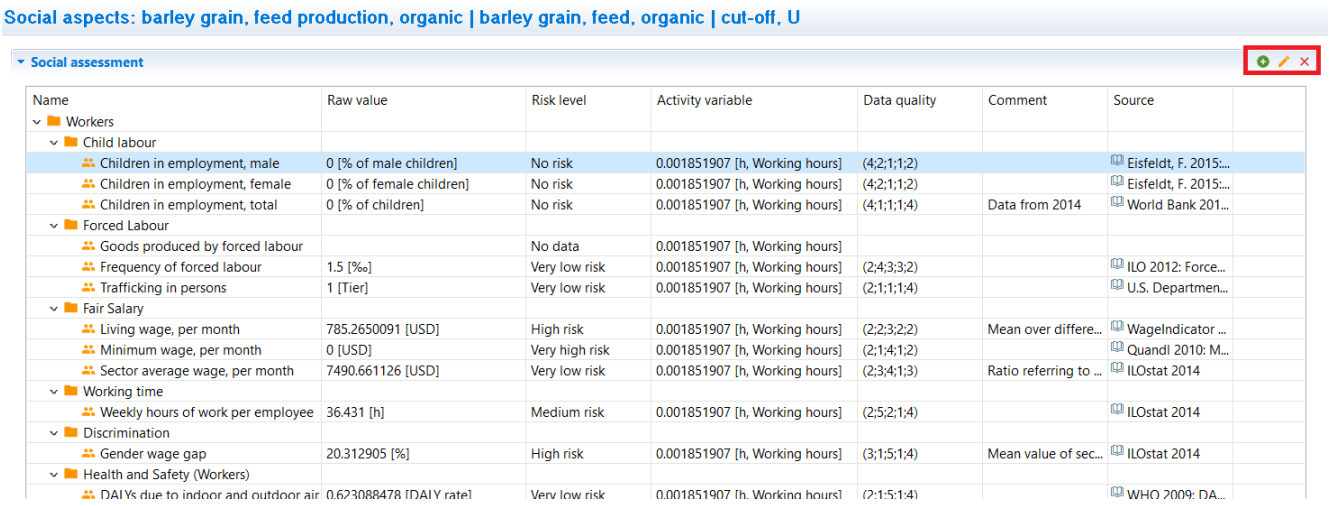


Figure 82: Social indicators, General information tab

For each process, the "Social Aspects" tab shows social indicators for different categories with information on raw value, risk level (evaluated according to the amount of the "raw value"), activity variable, data quality, comment and source. The risk-assessed indicators are characterised with the activity variable. For instance, for the time being, all indicators use working hours as activity variable.



Name	Raw value	Risk level	Activity variable	Data quality	Comment	Source
Workers						
Child labour						
Children in employment, male	0 [% of male children]	No risk	0.001851907 [h, Working hours]	(4;2;1;1;2)		Eisfeldt, F. 2015:...
Children in employment, female	0 [% of female children]	No risk	0.001851907 [h, Working hours]	(4;2;1;1;2)		Eisfeldt, F. 2015:...
Children in employment, total	0 [% of children]	No risk	0.001851907 [h, Working hours]	(4;1;1;1;4)	Data from 2014	World Bank 201...
Forced Labour						
Goods produced by forced labour		No data	0.001851907 [h, Working hours]			
Frequency of forced labour	1.5 [‰]	Very low risk	0.001851907 [h, Working hours]	(2;4;3;3;2)		ILO 2012: Force...
Trafficking in persons	1 [Tier]	Very low risk	0.001851907 [h, Working hours]	(2;1;1;1;4)		U.S. Departmen...
Fair Salary						
Living wage, per month	785.2650091 [USD]	High risk	0.001851907 [h, Working hours]	(2;2;3;2;2)	Mean over differe...	WageIndicator ...
Minimum wage, per month	0 [USD]	Very high risk	0.001851907 [h, Working hours]	(2;1;4;1;2)		Quandl 2010: M...
Sector average wage, per month	7490.661126 [USD]	Very low risk	0.001851907 [h, Working hours]	(2;3;4;1;3)	Ratio referring to ...	ILOstat 2014
Working time						
Weekly hours of work per employee	36.431 [h]	Medium risk	0.001851907 [h, Working hours]	(2;5;2;1;4)		ILOstat 2014
Discrimination						
Gender wage gap	20.312905 [%]	High risk	0.001851907 [h, Working hours]	(3;1;5;1;4)	Mean value of sec...	ILOstat 2014
Health and Safety (Workers)						
DALYs due to indoor and outdoor air	0.623088478 [DALY rate]	Very low risk	0.001851907 [h, Working hours]	(2;1;5;1;4)		WHO 2009: DA...

Figure 83: Process, social aspects

It is possible to modify a social indicator by double-clicking on it or by clicking on the pencil icon on the top-right corner. A window will pop-up with the possibility to modify the different categories.

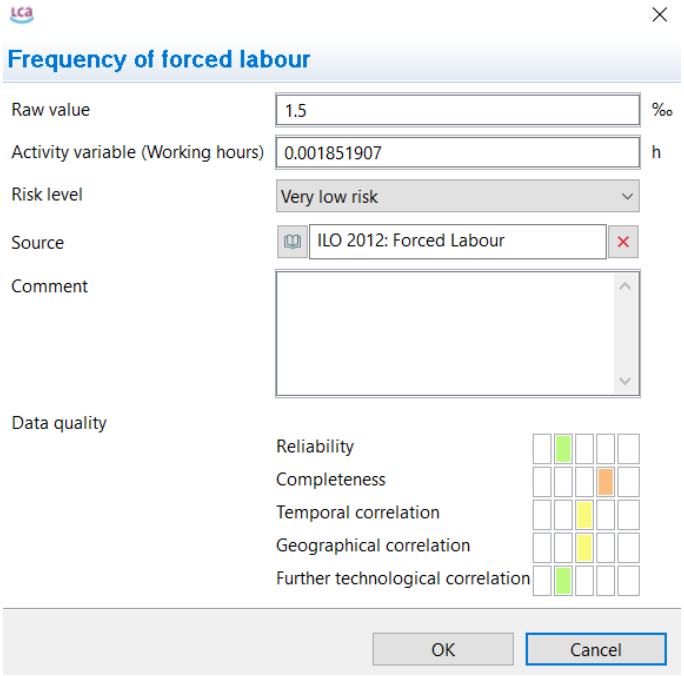


Figure 84: Process - Social aspects, modify social indicators

It is also possible to add social indicators by clicking on the green button on the top-right corner and selecting then an indicator from the available directories.

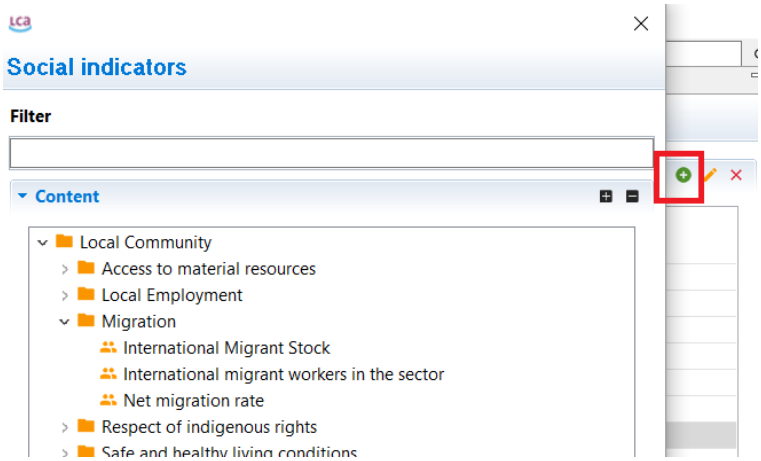


Figure 85: Process - Social aspects, add a social indicator

7 LCIA methods

The databases in openLCA Nexus do not contain LCIA methods (termed "impact assessment methods" in the software). LCIA methods need to be imported/created manually in each database in openLCA to carry out life cycle impact assessment.

7.1 Importing LCIA methods into openLCA

A LCIA method pack openLCA is available at www.openlca.org/downloads. This comprehensive package of environmental impact assessment methods is formatted for use with all the databases available at

openLCA Nexus, including, for example, ecoinvent 3, GaBi and ELCD. This pack includes normalisation and weighting as far as this is foreseen by the method. A pack containing a social LCIA method for use with the Social Hotspots Database also available at www.openlca.org/downloads. Ecoinvent LCIA methods are also available for openLCA. You can download these from openLCA Nexus (https://nexus.openlca.org/database/ecoinvent). A LCIA method developed especially for Ökobaudat is contained in the database file when downloaded from openLCA Nexus for openLCA.

Once you have downloaded one or more of these method packs to your computer, you can then import them into an openLCA database. To do so, begin by activating the database in which the method pack should be imported. Then go to "File" --> "Import" and select "Import entire database" from the "Other" folder of the import wizard. In the next window select "From exported zolca file", browse for the file and finally select "Finish". The import will then begin automatically.

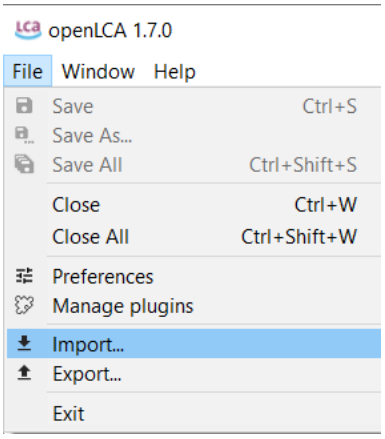


Figure 86: Importing LCIA methods, step 1

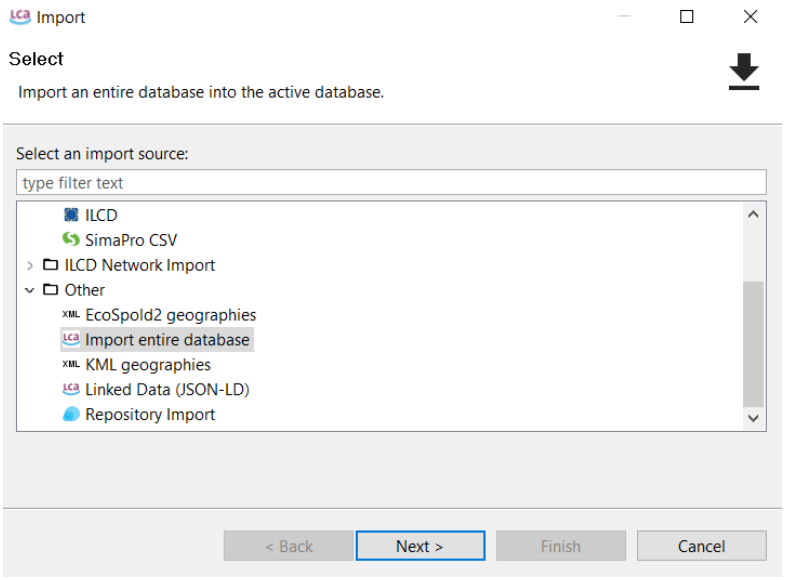


Figure 87: Importing LCIA methods, step 2

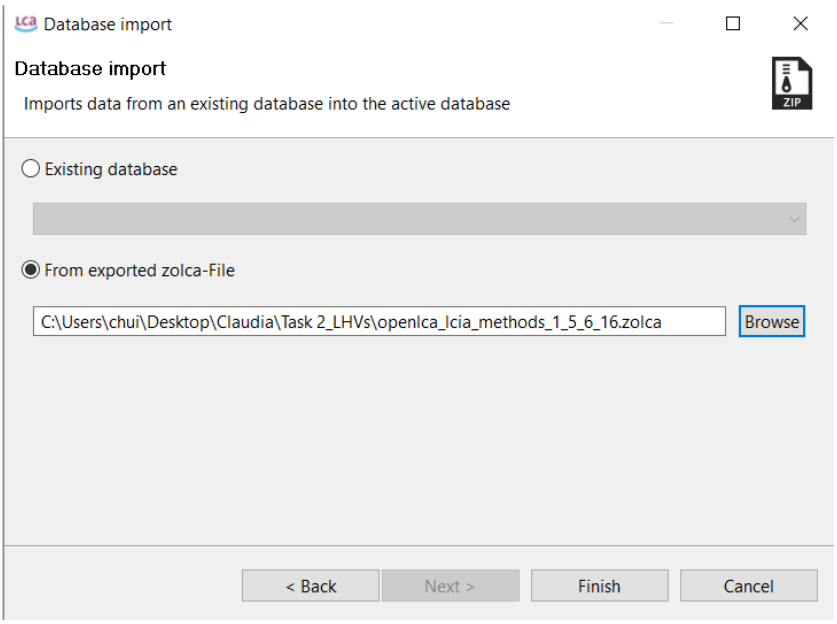


Figure 88: Importing LCIA methods, step 3

This import may take a few minutes. When it is finished, the LCIA methods will be available in the database, as shown in Figure 89.

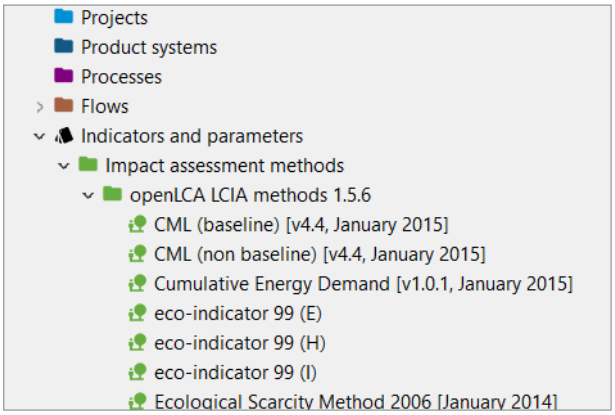


Figure 89: LCIA methods in openLCA

7.2 Creating a new impact assessment method

To create a new life cycle impact assessment method, right-click on the “Impact assessment methods” folder and select “New LCIA method”. Name the new method and add a description (optional). Once "Finish" is clicked, the new impact assessment method will open in the editor. To see how to add impact categories, characterization factors, etc, see the next section.

7.3 Impact methods tab contents

7.3.1 General information

Here you can change the name and description of the method as well as add/remove impact categories. To add a new impact category, click on the green "+" button on the right-hand side. Then simply name the category and type in a reference unit and description (optional). Figure 90 shows an example of the

General information tab for the CED method from ecoinvent.

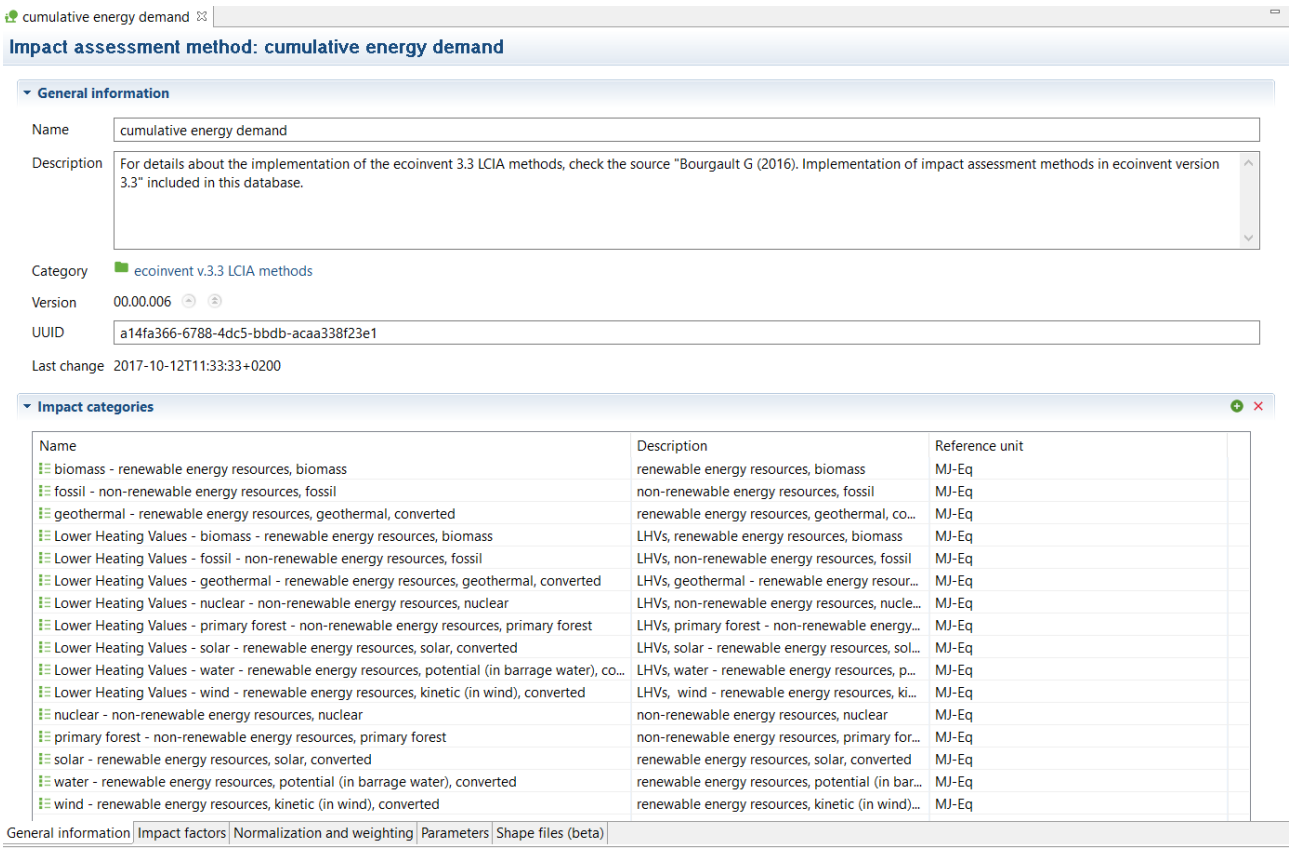


Figure 90: LCIA methods - General information tab

7.3.2 Impact factors

Flows as well as their corresponding characterization factors, categories, flow properties, units and uncertainty data can be added/removed/edited in this tab. Begin by selecting the desired impact category. Then add or remove flows using the "+" and "X" buttons on the top right-hand corner of the editor. Then type in the characterization factor for each flow as well as the uncertainty distribution, if applicable.

7.3.3 Normalization/Weighting

To add normalization and weighting factors to the impact categories of a LCIA method, begin in the "Normalization and weighting" tab by clicking on the green "+" tab to add a new set. The impact categories saved in the method will automatically appear in the window on the right, where you can then manually type in normalization and weighting factors.

7.3.4 Parameters

Parameters can be used in the same way for LCIA methods as for processes, as described in section 6.2.6.

7.3.5 Shape file parameters

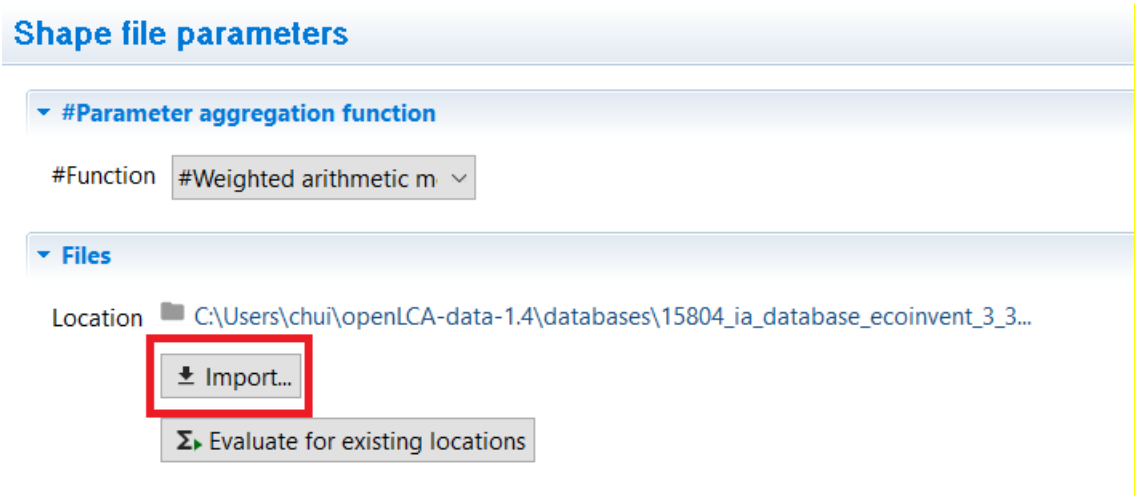


Figure 91: Shape files tab, import files

This tab is connected to Regionalized LCA, see section 9.3. The data for the regional characteristics is imported to openLCA in shape files (i.e. GIS vector data), being each characteristic an attribute of the existing geometries in the file. The imported attributes can then be added as new parameters in the LCIA method editor. This way, site-specific data is only needed for the regional characteristics included in the LCIA model, instead of having to import GIS files with thousands of attributes representing the impact factors of each elementary flow per impact category.

Shape file parameters are created in openLCA by importing GIS files of extension *.shp which contain the parameters as numerical attributes of each feature included in the file. 1. For importing the GIS file, click "Import" in the "Shape files" tab of the LCIA method editor and select the file to import. Once the shape file has been imported, the numerical attributes included in it will be added as parameters in openLCA. The name of the parameter will be the name of the attribute in the shape file and its minimum and maximum values will be displayed in the table. The different values per location and parameter can also be checked in a map. For this, right-click on the parameter name and select "Show in map".

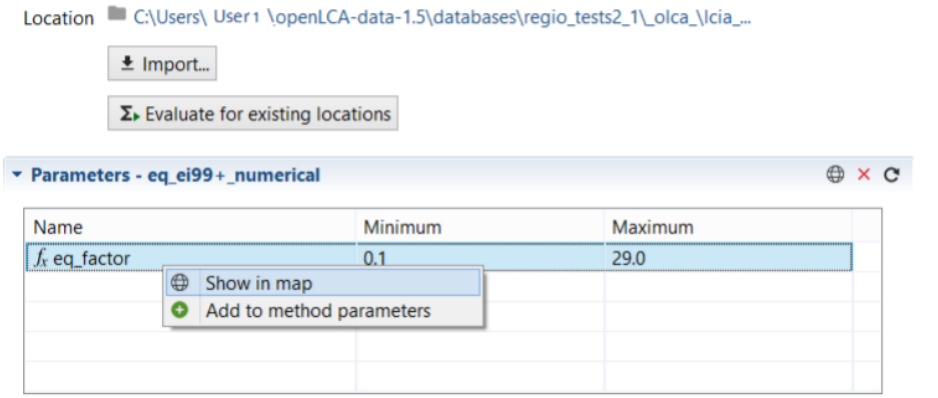


Figure 92: Shape files tab, import files, show in map

The value of each feature included in the shape file can be checked selecting the function "i" available in

top left corner of the pop-up window, and then clicking on the point of interest in the map. An additional pop-up window will show the different information available in the shape file for the feature that includes the selected point.

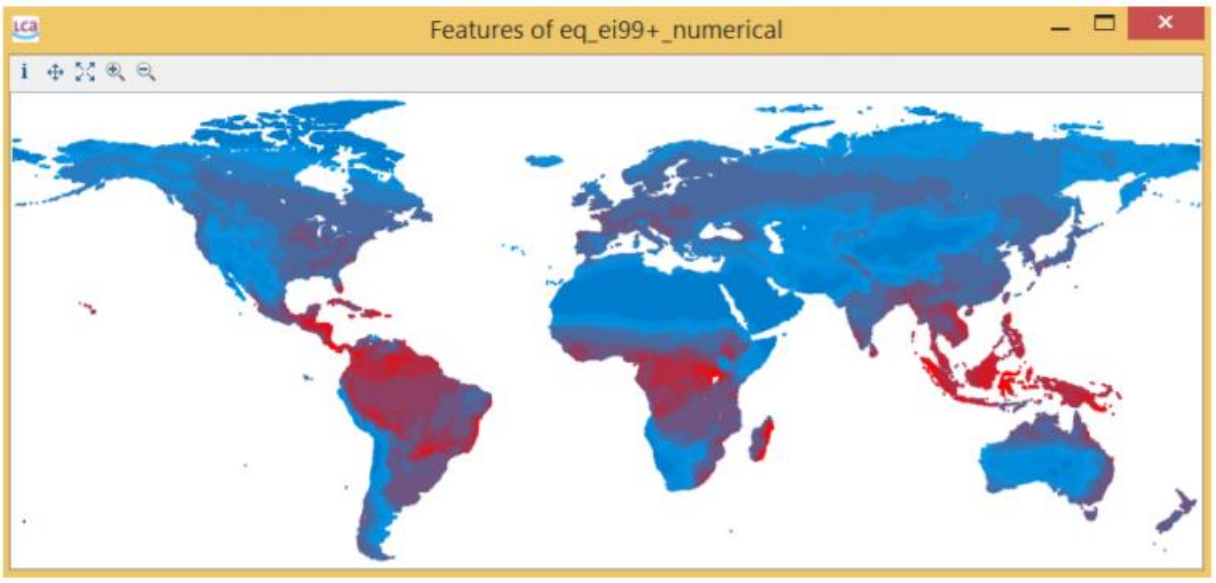


Figure 93: Map displaying the values of the selected parameter for all the features included in the imported shape file

The function "Evaluate for existing locations" pre-calculates the intersections with all the locations in the database. If the intersections are not pre-calculated and a regionalized LCIA calculation is performed, the intersections for the locations used by the product system will be determined during the calculation and saved for later use. Once the intersections are calculated, a weighted average value for each parameter is obtained and the formulas defined in the impact factors are evaluated with the correspondent parameters values. If no KML data is available for a location, the default value of the parameter is used in the calculation of the impact factor. The resultant impact factors are finally applied to the regionalized inventory and site-specific impact results are calculated.

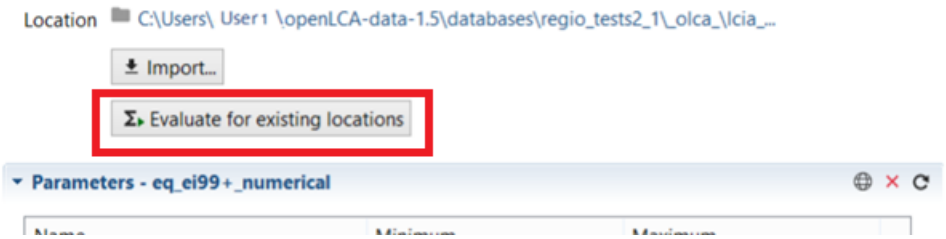


Figure 94: Shape files tab, Evaluate for existing locations selection

8 Product systems

As in ISO 14040, the life cycle model of a product is called a product system. There are different ways to create, edit and complete product systems, depending on the database and user preferences,

which will be explained in the following section.

8.1 Creating a new product system

There are two ways to create a new product system. For option one, begin by right-clicking on the "Product systems" folder and select "New product system". The second option is to create a product system directly from a process. To do this, go to the "General information" tab of the process and select the button "Create product system" (Figure 95).

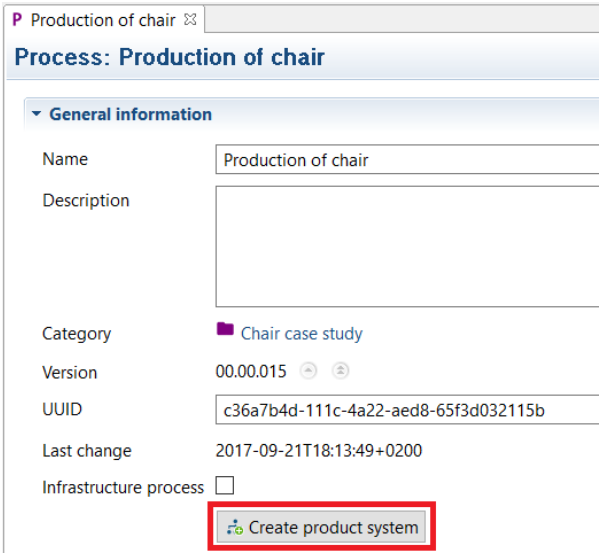


Figure 95: Creating a product system, step 1

The next steps are then the same for both cases. Namely, a pop-up window will appear (Figure 96) in which you can name the product system, add a description (optional) and select a reference process.

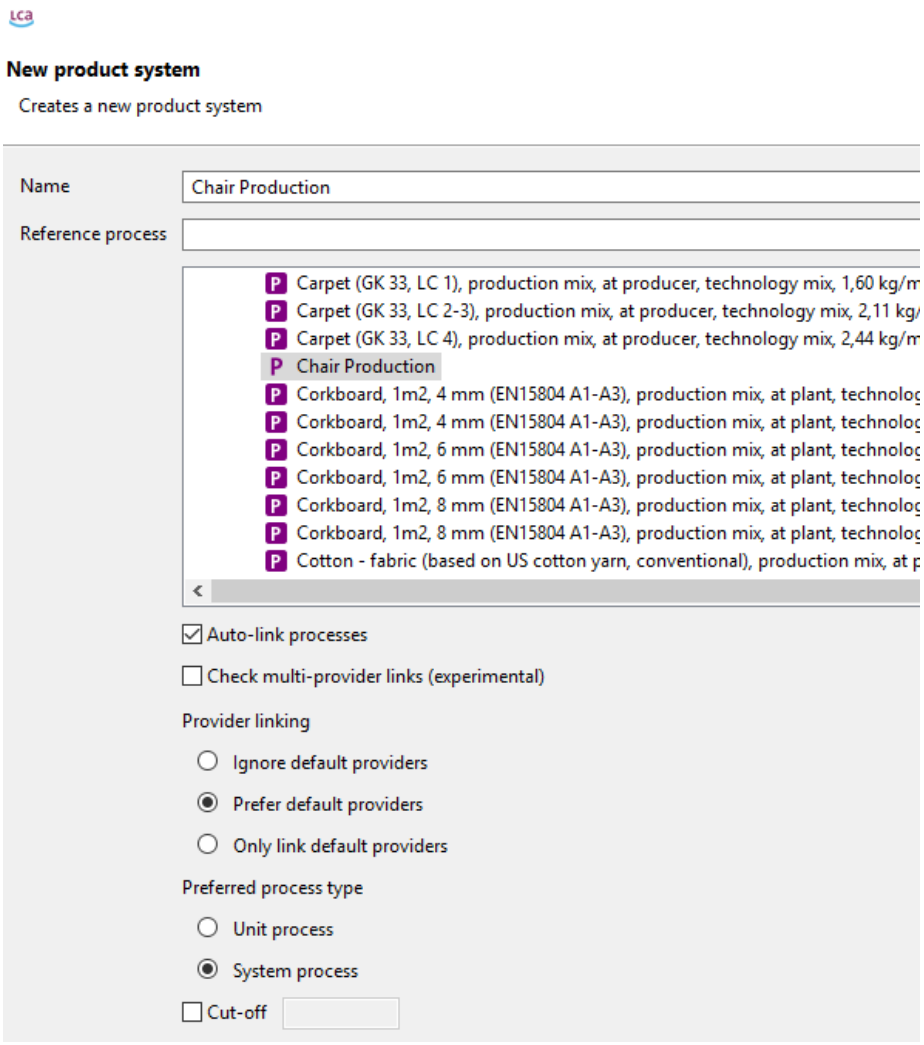


Figure 96: Creating a product system, step 2

A reference process is the process at the very end of the chain. If you create a product system directly from a process, that process will automatically be selected as the reference process. Otherwise, by entering the text into the "Reference Process" field, you can narrow down the selection which helps to identify the desired process.

When creating a product system, openLCA can automatically check for flows with multiple providers. If flows with multiple providers are detected in a product system, a dialogue pops up which lets you select the default provider for the respective flows.

To automatically have all upstream processes linked to the reference process, select "auto-link processes". This enables three options for provider linking.

- “Only link default providers”: Many processes from databases such as ecoinvent have preselected providers. Enabling this option will make sure that exclusively these providers are linked with the product system, and none else.

- “Prefer default providers”: Enabling this option will make sure the mentioned default providers are used when they are available. When they are not available, another provider with a fitting reference flow will be used.
- “Ignore default providers”: Enabling this option will make sure that default providers are not used whenever they are available.

The next step is to choose whether to save the process as a unit process or system process. The difference between the two is describe in section 6.

Finally, it is possible to set a cut-off threshold. Processes that contribute to less than the chosen percentage will be disregarded as providers.

To create the product system, click “Finish”

After creating a product system, it is possible to add and delete connections using the "Model graph". This will be described in section 8.2.3.

8.2 Product system tabs contents

8.2.1 General information

Here you can change the name of the product system as well as add a description (optional). In the "Reference" section you can see the reference process and make changes to the reference product, flow property, unit and the target amount. The target amount should be selected in accordance with your functional unit. It is also possible to calculate a product system from the general information tab. For more information, please see section 8.4.

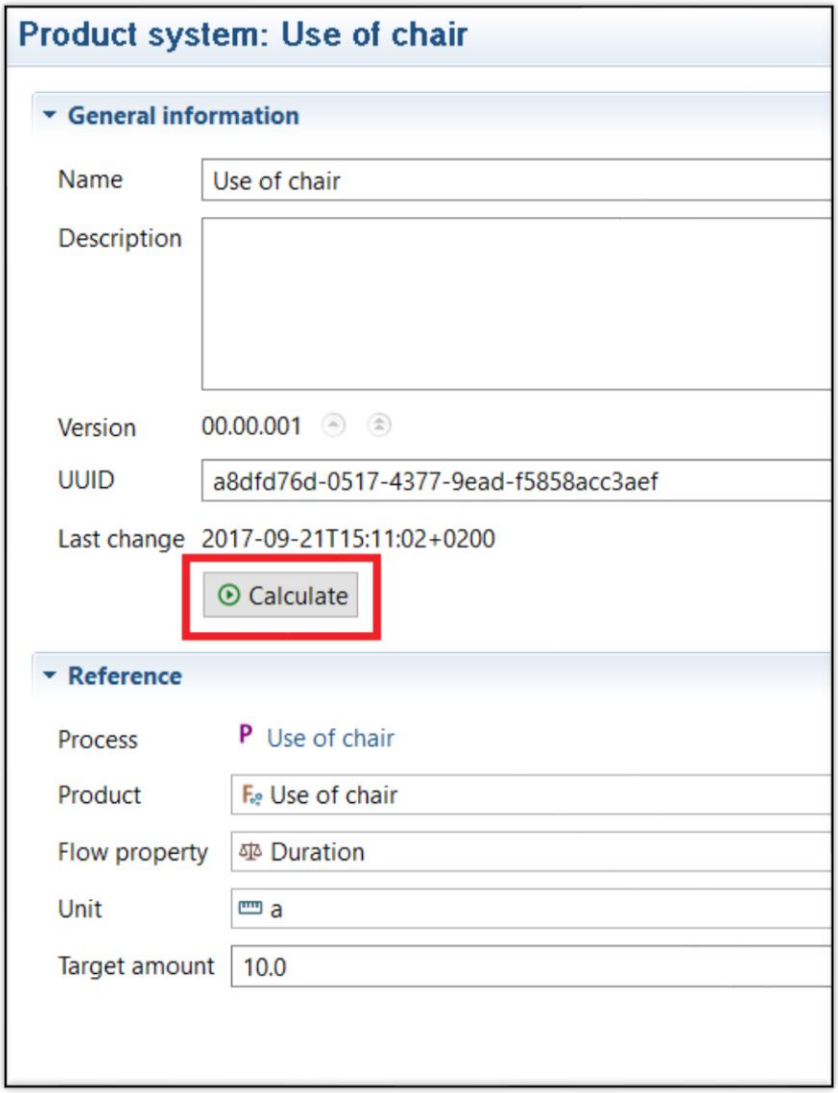


Figure 97: Product system - General information tab

8.2.2 Parameters

In the product system level, it is possible to change the amounts of parameters defined in the processes included in that product system. To do so, add the parameter for which you would like to change the amount by selecting the green "+" button in the top right-hand corner of the Editor and selecting one. To select multiple parameters at once use your keyboard's "Shift" button. The amounts saved in a product system will override those saved in a process. However, the values saved in the process will not change. It is not possible to create new parameters on the product system level.

8.2.3 Model graph

The model graph in the product system shows the connected processes. These connections can be edited (added/deleted) and processes can be entirely deleted from the product system if they have no connections in the life cycle any more. Only connected processes will contribute to the product system's calculation. To find out more, see the next sections on making alterations to the product

system within the model graph and on calculating product systems.

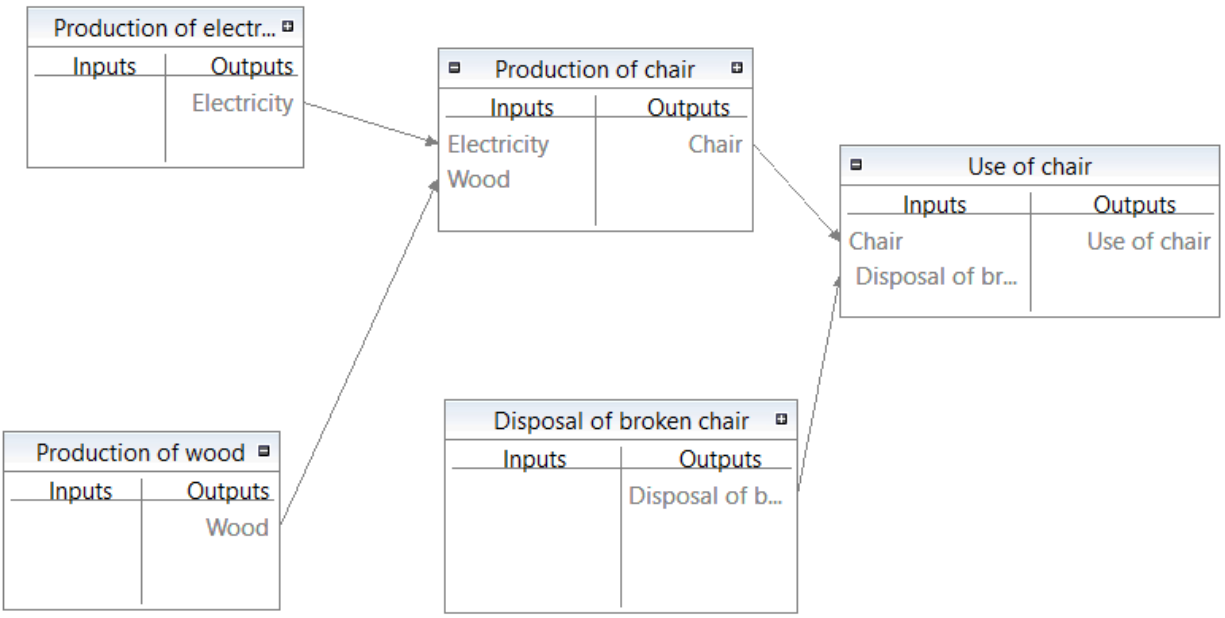


Figure 98: Product system model graph - example

By right-clicking on the background of the model graph, the following options will appear:

- Save as image (save image of model graph as .png file)
- Expand all (expand model graph to show all connected processes)
- Collapse all (minimize connected processes to show only first and second tier)
- Maximize all (expand processes' view to show inputs/outputs)
- Minimize all (show only process names, no inputs/outputs)
- Layout (choose between "Layout as tree", "Layout as minimal tree" to organize the model graph; select "Route" to have the model graph displayed with connections made of right angles)
- Show outline (the "Outline" lists all processes in the product system alphabetically. To find a process in the model, right-click on the flow in the Outline and select "Show" and it will appear in the model graph)
- Open miniature view (useful for navigating your way around complex model graphs by giving a miniature overview of the model graph. The blue area represents the current view (see Figure 99).

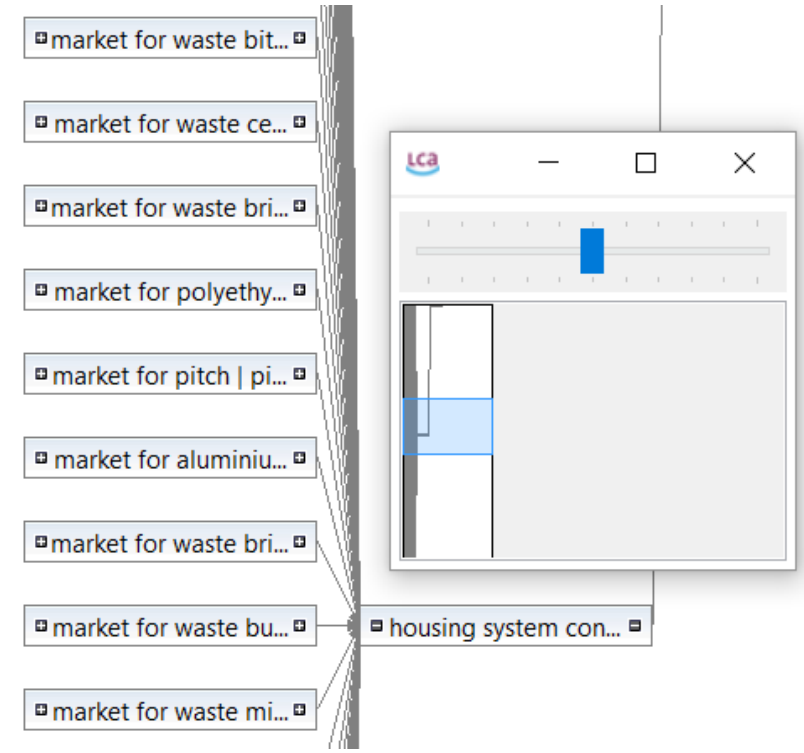


Figure 99: Model graph - miniature view

As an alternative to the automatic connection of processes, you can also manually connect processes in the model graph of a product system. By right-clicking on the processes, under "Build supply chain" you can then select whether to build the entire supply chain for the process, or just the next tier.

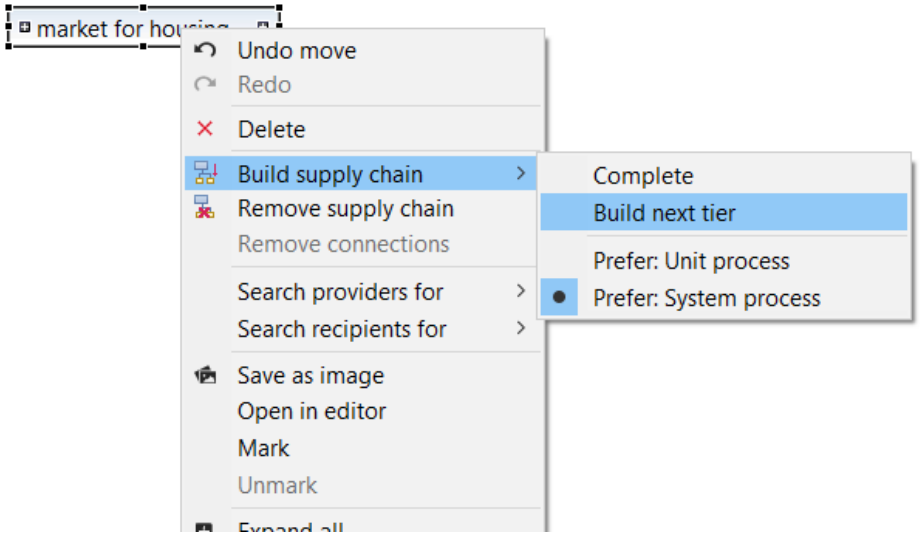


Figure 100: Model graph - build next tier

Selecting "Search providers for" allows you to select the providers for each individual product of the process. To do so, right-click on a process, select "Search providers for" and then select the respective product. A pop-up window will appear with a list of all possible providers for that product. If the listed providers are not already present in the model graph, you can select in the table which provider you would like to add to the model graph and check off "Connect" to automatically connect the process to the

product. Likewise, it is possible to search for recipients for specific outputs.

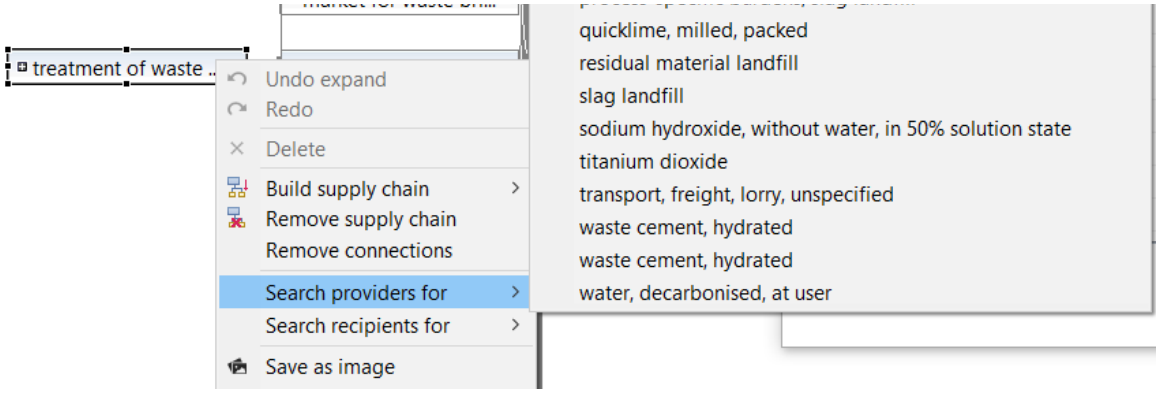


Figure 101: Model graph - search providers for, step 1

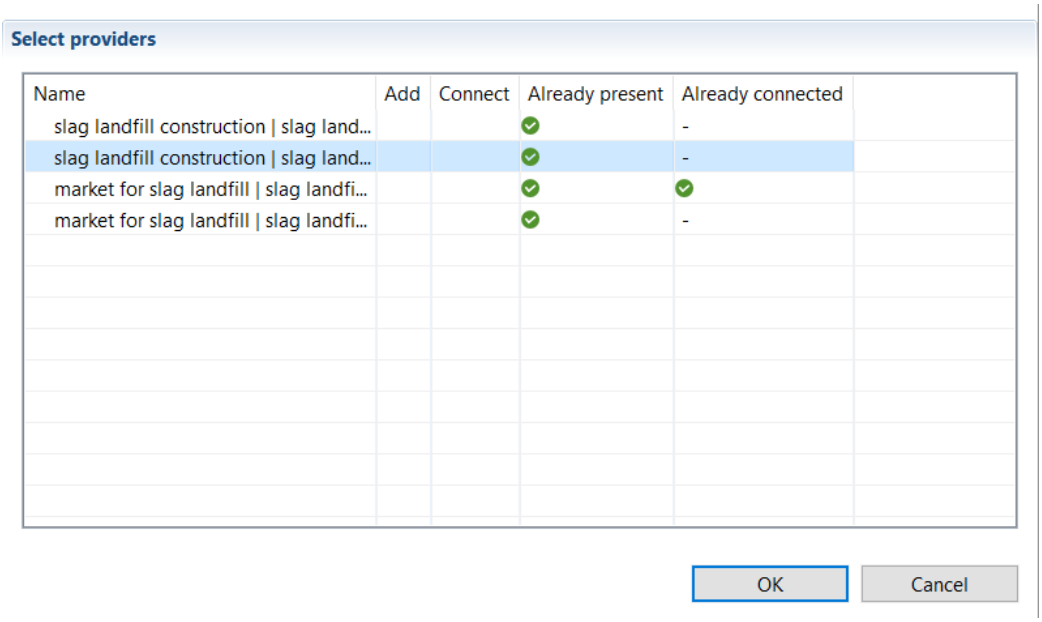


Figure 102: Model graph - search providers for, step 2

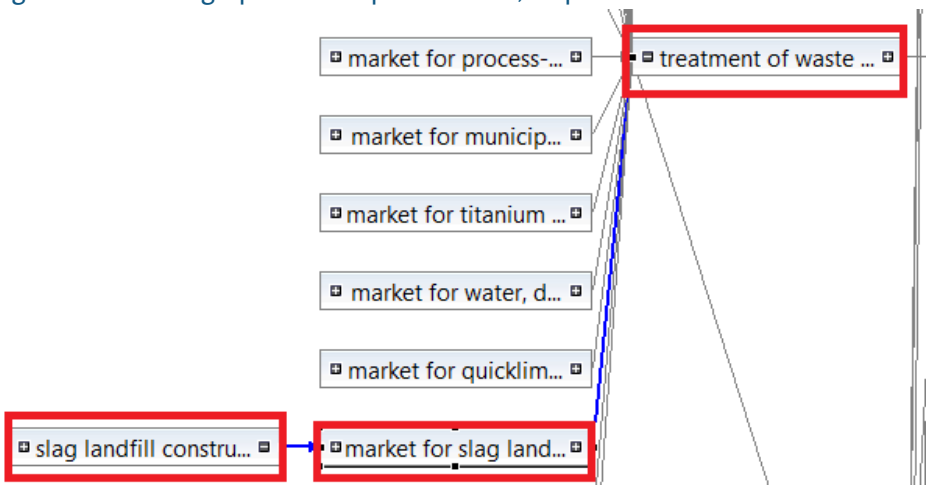


Figure 103: Model graph- search providers for, connected process

Another way to connect inputs/outputs to their respective providers/recipients is to 'draw' the connection manually in the model graph. Click on a not-connected input/output and guide the mouse to the

respective provider/recipient. If the symbol ⊘ appears next to the mouse, this indicates that no valid provider/recipient has been selected.

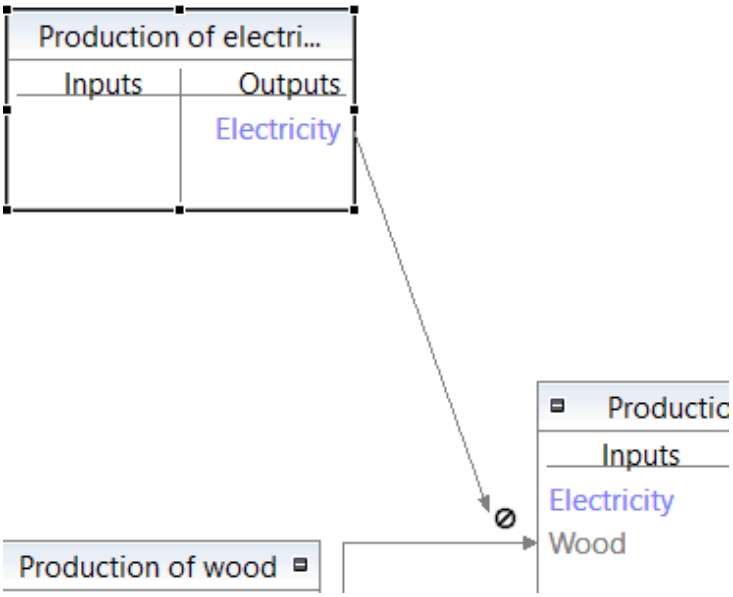


Figure 104: Model graph - drawing connections, step 1

Release the mouse when over valid provider/recipient and the connection will be made:

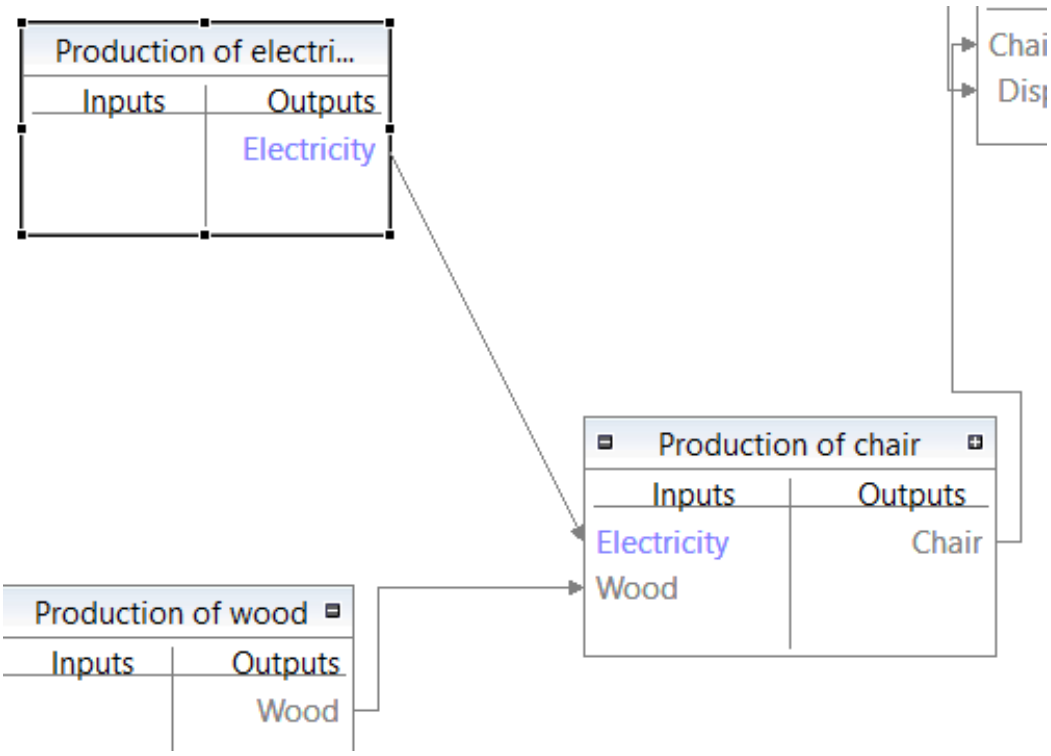


Figure 105: Model graph - drawing connections, step 2

It is also possible to delete connections in the graph. To do so, simply right-click on the connection arrow and select "Delete":

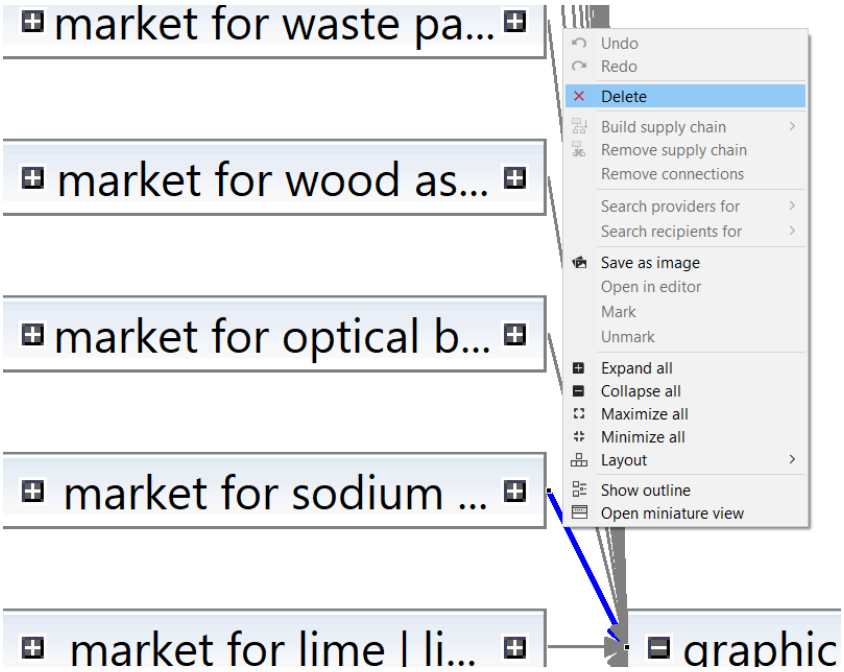


Figure 106: Model graph - deleting connections

Once a connection has been deleted, the upstream process will still be shown in the model graph; however, as it is not connected, it will not contribute to the calculation. Delete an unconnected process by right-clicking on it and selecting "Delete".

8.2.4 Statistics

This section gives information about general statistics, including number of processes, links, if the graph is connected, and the name of the reference process. Additional information about provider linking and processes with the highest in-degree and out-degree is provided. The in-degree counts how many times a process is mentioned in the input side of the product system. The out-degree shows how many times a process is used on the output side of the product system. The process with the highest out-degree is thus likely to have a large impact on the product system, but the extend of its impact depends on the quantity.

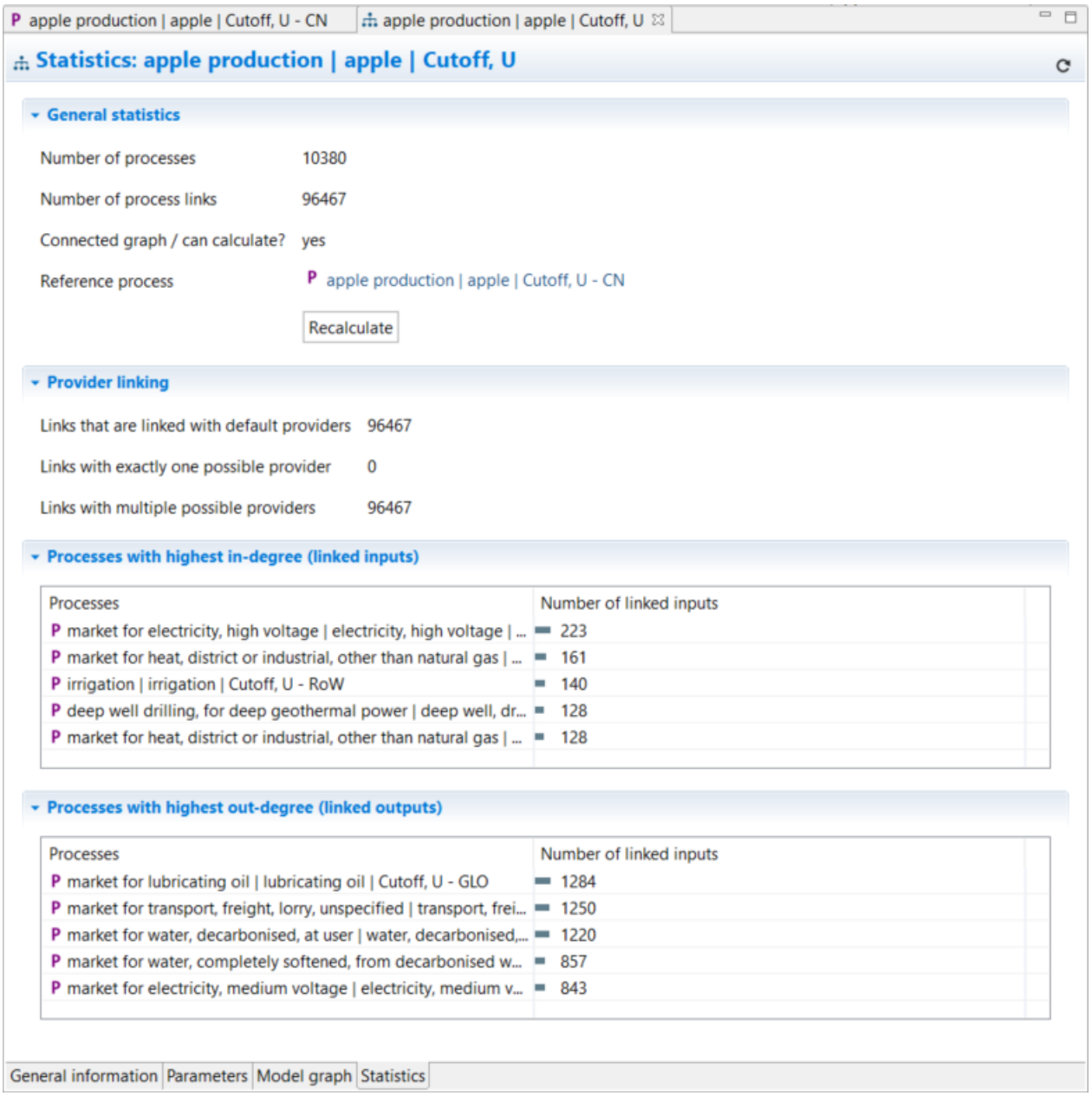


Figure 107: Product system statistics

8.3 Nested product systems

A product system can serve as the provider to another product system. Drag and drop a product system into the model graph of a different product system (Figure 108) and connect it to one of the input flows via “search recipients for”.

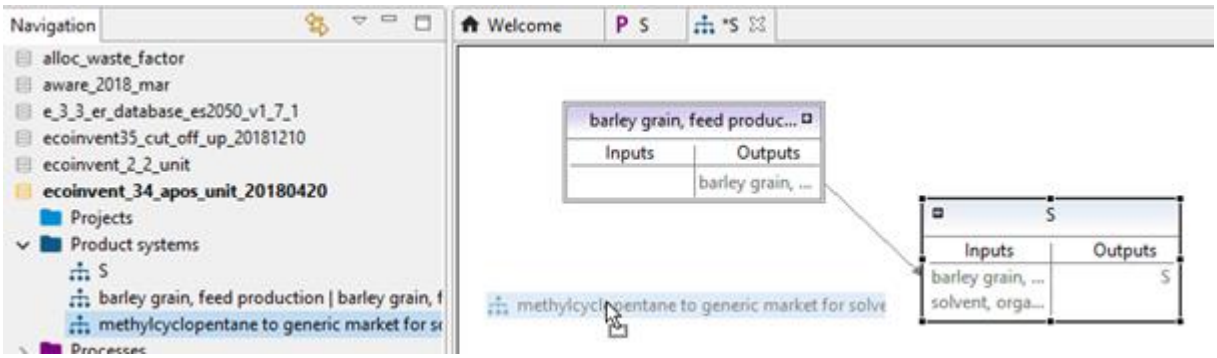


Figure 108: Drag-and-drop of a product system into another product system

The contributions of the sub product system to the overall results are displayed e.g. in the impact analysis and contribution tree results.

A product system can also be used as an input flow for a process (Figure 109). The flow of the quantitative reference of the product system is then added to the process.

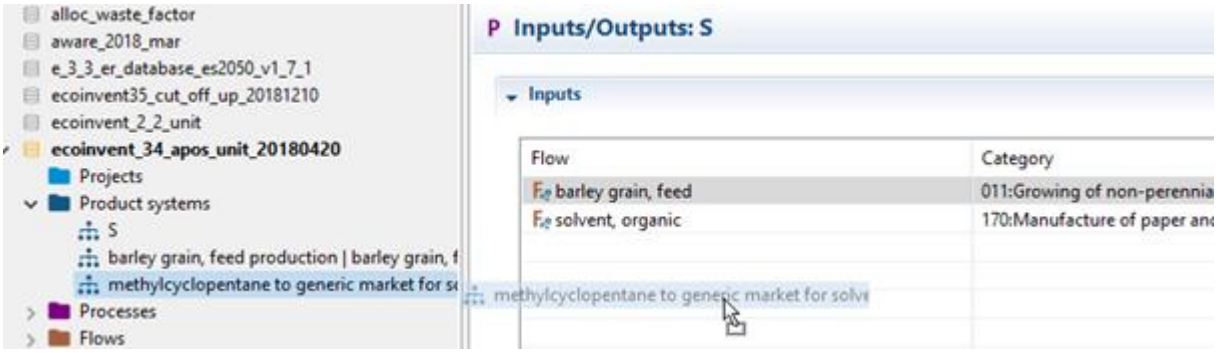


Figure 109: Drag-and-drop of a product system into the input flows of a process

Details about nested product systems are available here.

8.4 Calculating a product system

When you have a product system open in the Editor, there are two ways to initiate calculation. Either select the "Calculate" button from the General information tab of the product system or the "Calculate results" icon in the top left-hand corner:

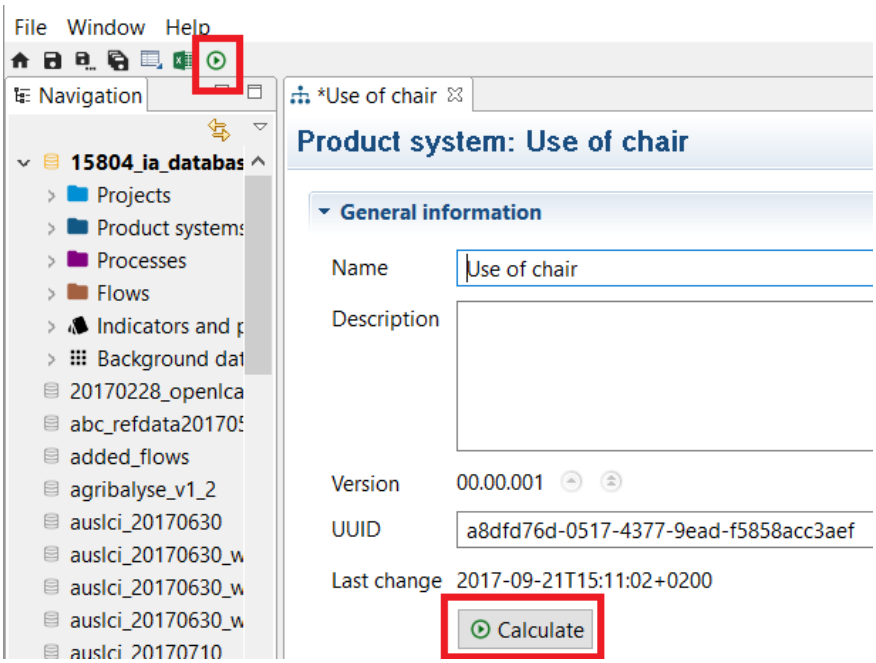


Figure 110: Calculating a product system, step 1

You can then select the calculation properties in the wizard. For allocation, "None" is the default setting. Otherwise, you can choose between the options "Causal", "Economic", "Physical" or "As defined in processes". Then, select the desired impact assessment method from the list of methods available in your activated database (if no methods are listed you need to first import one or more methods into the database or create a new method). If applicable, select a normalization and weighting set. The calculation types "Quick results", "Analysis", “Regionalized LCA” and "Monte Carlo Simulation" will be explained in more detail in section 9. It is possible to include cost calculation and assess data quality when related boxes are checked. Finally, click "Finish" to begin calculation. Calculation time can vary, depending on the size of the product system and database, and the type of calculation (e.g. Monte Carlo Simulations take considerably longer than other calculations).

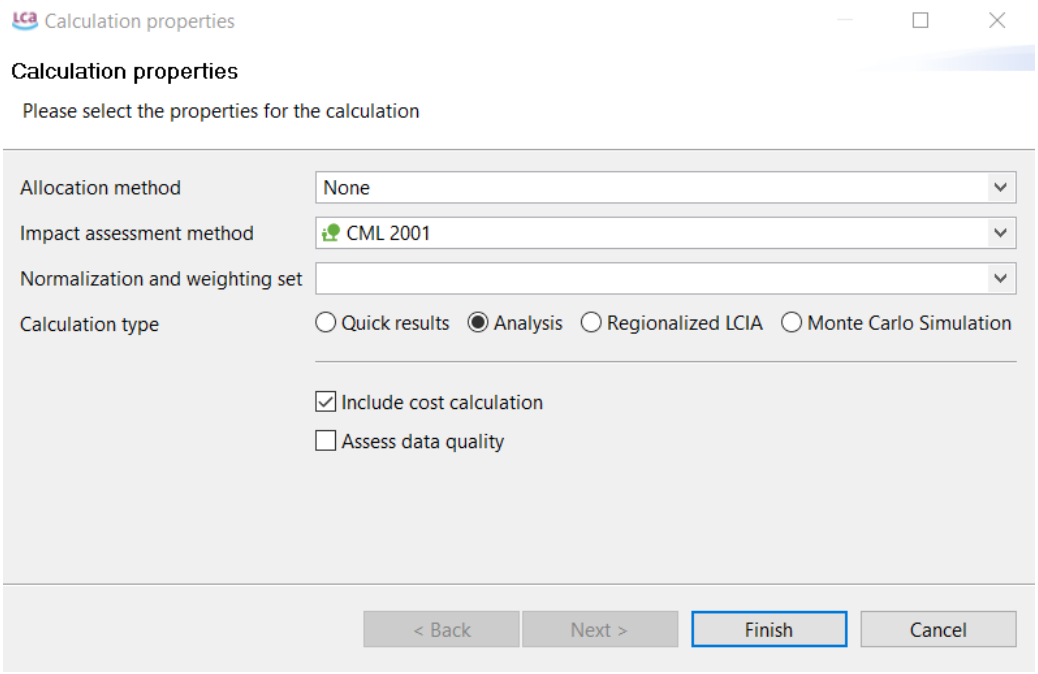


Figure 111: Calculating a product system, step 2

For the calculation of the data quality of a product system an aggregation needs to be done. After checking the box, you click on “Next” and you are free to select an aggregation method, a rounding method and what to do with exchanges that do not have a data quality value.

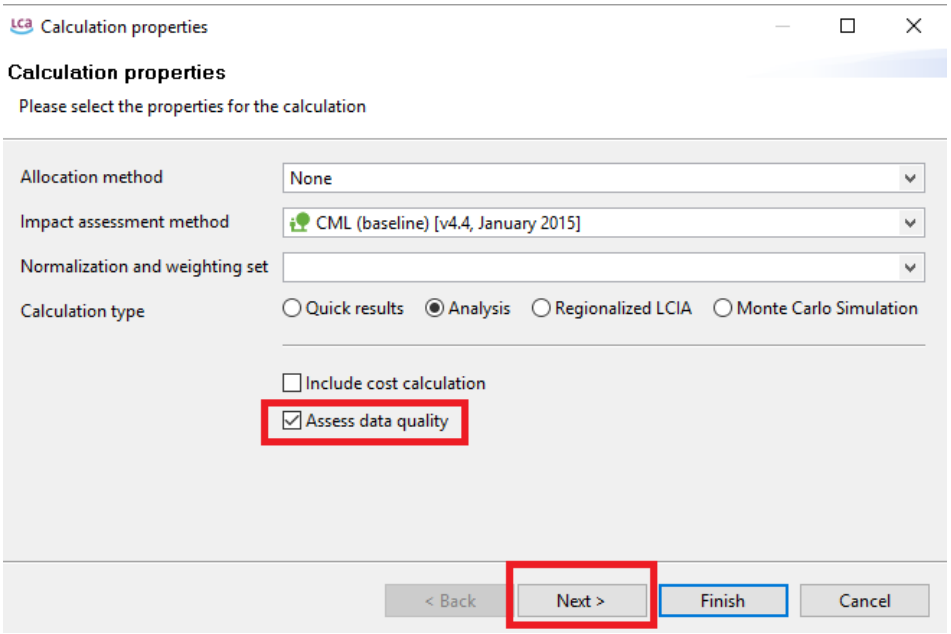


Figure 112: Calculating a product system, assessing data quality, step 1

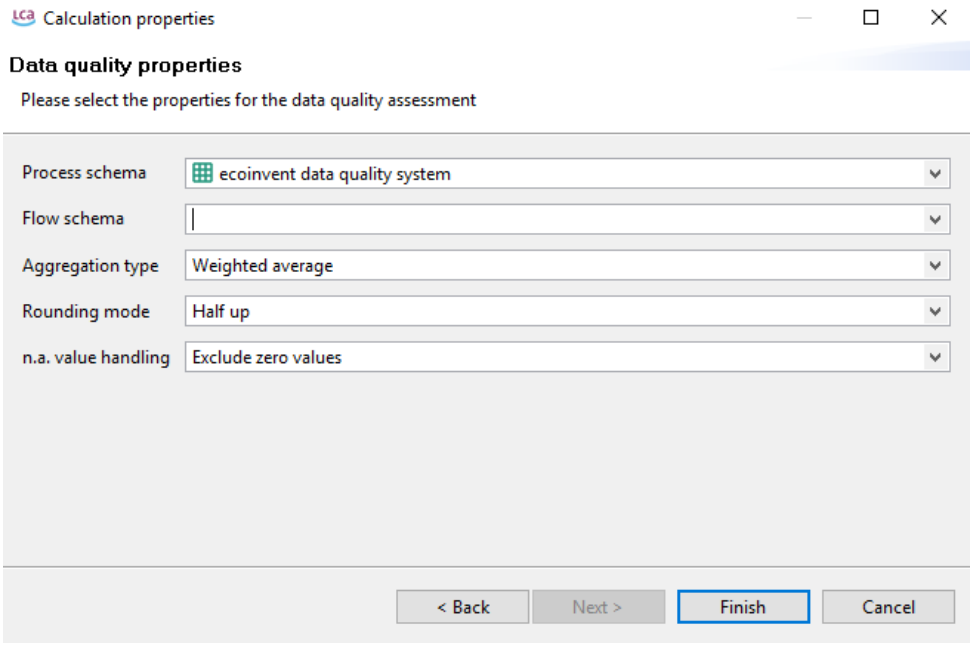


Figure 113: Calculating a product system - assessing data quality, step 2

9 Result Analysis

This section will cover the four different calculation types "Quick results", "Analysis"," Regionalized LCA" and "Monte Carlo Simulation" and describe the information contained in each of the result edit tabs. Life Cycle Costing in OpenLCA is also presented in the last section.

9.1 Quick results tab contents

The option "Quick results" provides information on direct impacts. Upstream impacts are not represented here (they are included in the Analysis results).

9.1.1 General information

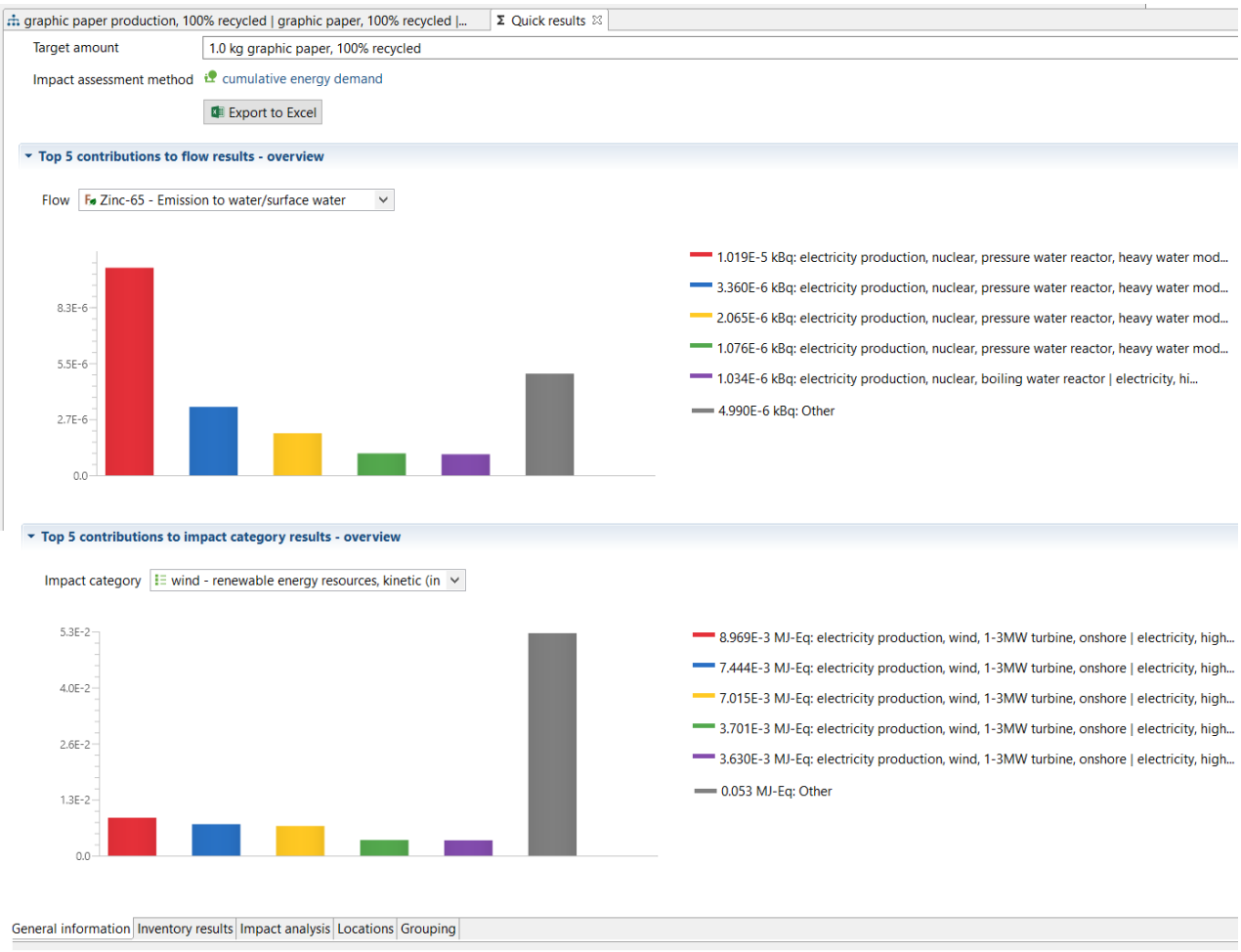


Figure 114: Quick results - General information tab

The General information tab displays which product system was calculated, including information on the allocation method, target amount and LCIA method used for the calculation. The "Top 5 contributions to flow results - overview" section shows a histogram chart illustrating the five processes with the highest direct contributions to the selected flow. Change the flow information displayed by selecting the desired flow from the list. Likewise, the " Top 5 contributions to impact category results - overview " section displays the five processes with the highest direct contributions to the selected impact assessment category.

If the box “Assess data quality” is checked when setting the calculation properties (see section 8.4), information and statistics about data quality is displayed in the Data Quality Section according to the data quality schema previously defined for the processes, see section 6.2.2.

9.1.2 Inventory results

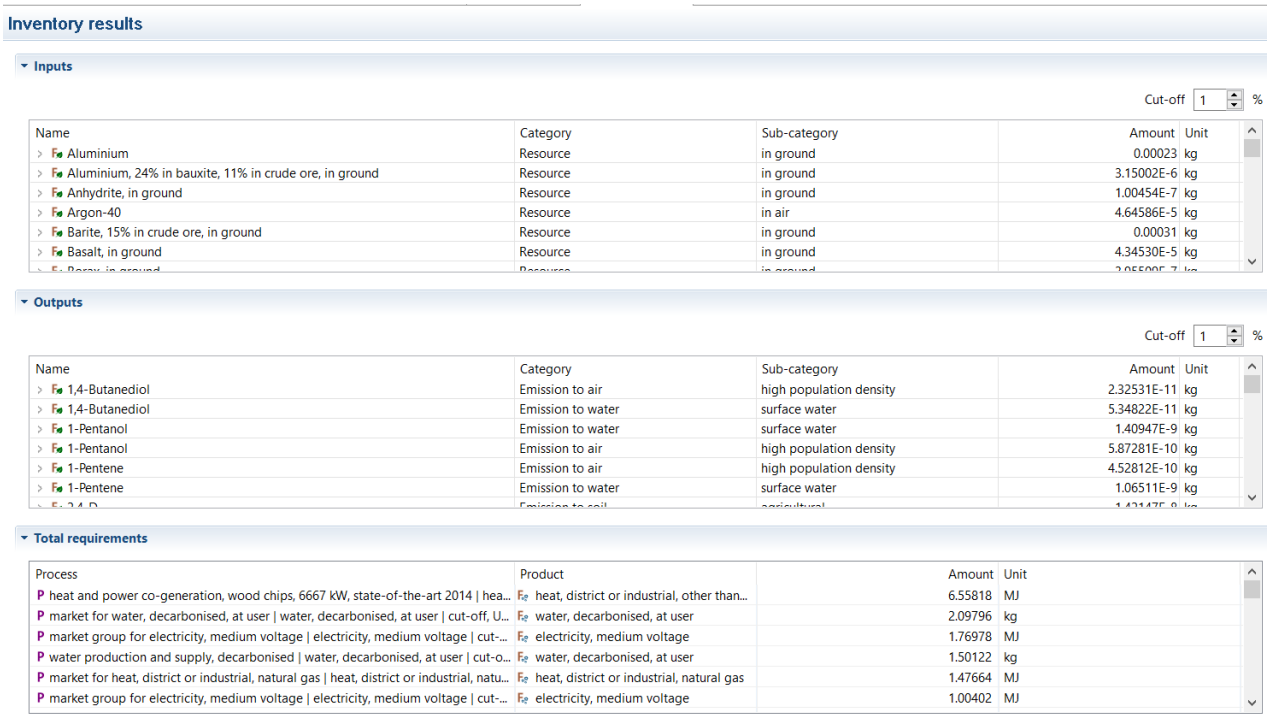


Figure 115: Quick results - Inventory results tab

The inventory results tab contains a table of all the product system's input and output flows, displaying the amounts and units for each entry (in random order). To see the list of flows in alphabetical order, simply click on "Flow". Likewise, to have the table organized according to category, sub-category, unit or amount, click on those respective cells at the top of the table. If you click on the arrow symbol before the name of the flow in Inputs or Outputs section, all processes in the product system where the flow is used will appear (see Figure 116).

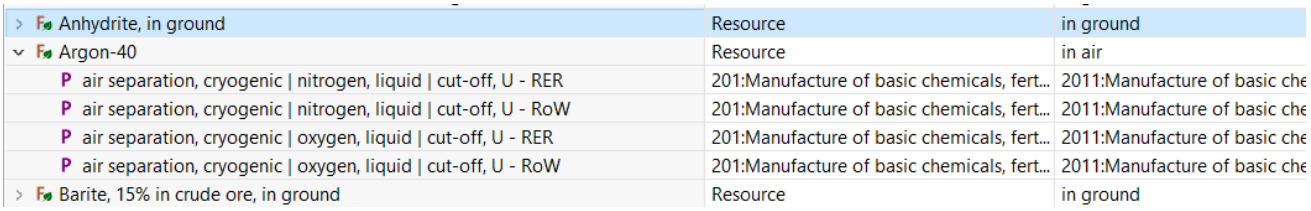


Figure 116: Quick results - Inventory results, inputs section

If the box "Assess data Quality" is checked when setting the calculation properties (see section 8.3), information about data quality is displayed in the Inputs and Outputs Section according to the data quality schema previously defined for the processes, see section 6.2.2.

Sub-category	Amount	Unit	R	C	T	G	F
in ground	0.23498	kg	3	3	5	5	3
in ground	8.40286E-5	kg	1	3	5	5	1
0810:Quarrying of stone, sand and clay	7.79835E-5	kg	1	3	5	5	1
0810:Quarrying of stone, sand and clay	5.02312E-6	kg	1	3	5	1	1
in ground	0.11672	kg	1	1	5	5	1
0520:Mining of lignite	0.11219	kg	1	1	5	5	1
0520:Mining of lignite	0.00274	kg	1	1	5	1	1
in ground	2.68976E-5	kg	2	4	4	5	1

Figure 117: Quick results - Inventory analysis - data quality information

The information presented in all openLCA tables can be copied from the openLCA editor and pasted elsewhere (e.g. in excel, notepad, etc.).

The last table contained in this tab is named "Total requirements" and it shows in the first column all the processes that are included in this product system. The second column shows the output product of the related process in the first column and its unit and amount (Figure 115).

If the box "Include Cost Calculation" is checked when setting the calculation properties (see section 8.4), the calculation of the added value for each process and the total result are displayed.

Total requirements

Process	Product	Amount	Unit	Added value
heat and power co-generation, wood chips, 6667 kW, state-of-the-art 2014 | hea...	heat, district or industrial, other than...	6.55818	MJ	0.04 USD
market for water, decarbonised, at user | water, decarbonised, at user | cut-off, U...	water, decarbonised, at user	2.09796	kg	0.00 USD
market group for electricity, medium voltage | electricity, medium voltage | cut-...	electricity, medium voltage	1.76978	MJ	0.00 USD
water production and supply, decarbonised | water, decarbonised, at user | cut-o...	water, decarbonised, at user	1.50122	kg	0.00 USD
market for heat, district or industrial, natural gas | heat, district or industrial, natu...	heat, district or industrial, natural gas	1.47664	MJ	0.00 USD
market group for electricity, medium voltage | electricity, medium voltage | cut-...	electricity, medium voltage	1.00402	MJ	0.00 USD

Total added value: 0.11 USD

Figure 118: Quick results - inventory analysis, added value calculation

Previously calculated inventory results (*quick calculations*) are stored. If you work with the LCA Collaboration Server, calculated inventory results are shared along with product systems.

9.1.3 Impact analysis

Impact analysis

Subgroup by processes ☑ Cut-off 1 %

Name	Category	Inventory result	Impact factor	Impact result	Unit
solar - renewable energy resources, solar, converted				7.45079E-5	MJ-Eq
electricity production, photovoltaic, 570kWp open ground installa	3510:Electric power generation, transmi...			7.24460E-6	MJ-Eq
Energy, solar, converted	Resource / in air	7.24460E-6 MJ	1.00000 MJ-Eq/MJ	7.24460E-6	MJ-Eq
electricity production, photovoltaic, 3kWp slanted-roof installatio	3510:Electric power generation, transmi...			6.48802E-6	MJ-Eq
electricity production, photovoltaic, 3kWp slanted-roof installatio	3510:Electric power generation, transmi...			5.22206E-6	MJ-Eq
electricity production, photovoltaic, 3kWp slanted-roof installatio	3510:Electric power generation, transmi...			4.73412E-6	MJ-Eq

Figure 119: Quick results - LCIA Results tab

This tab will be included in the quick results in the case that an impact assessment method was selected in the calculation wizard. The table lists the results and reference units for the respective impact assessment method categories with related processes (and flows) that contribute to the impact categories.

If the box "Assess data Quality" is checked when setting the calculation properties (see section 8.4),

information about data quality is displayed in the Impact analysis according to the data quality schema previously defined for the processes, see section 6.2.2.

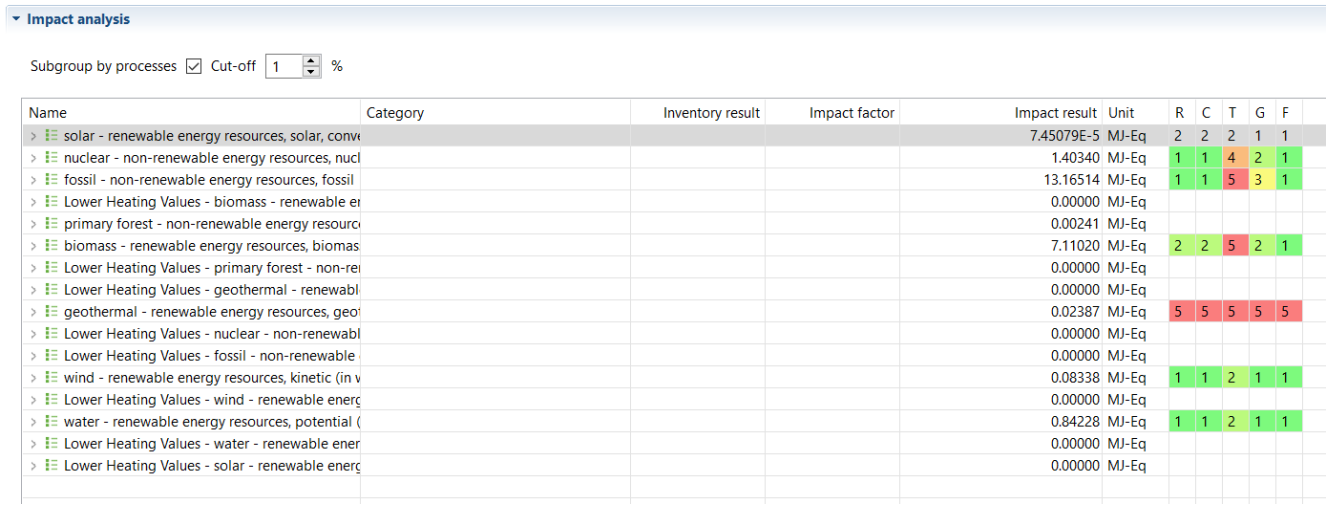


Figure 120: Quick analysis - Impact Analysis, data quality

9.1.4 LCIA checks

Upon calculating the Analysis results of a product system (or Quick results if an LCIA method has been selected), the tab LCIA Checks lists all flows of the Life Cycle Inventory (LCI) of the calculated product system that are not captured by the applied LCIA method. The flows can either be listed regardless of the LCIA category or grouped by LCIA category.

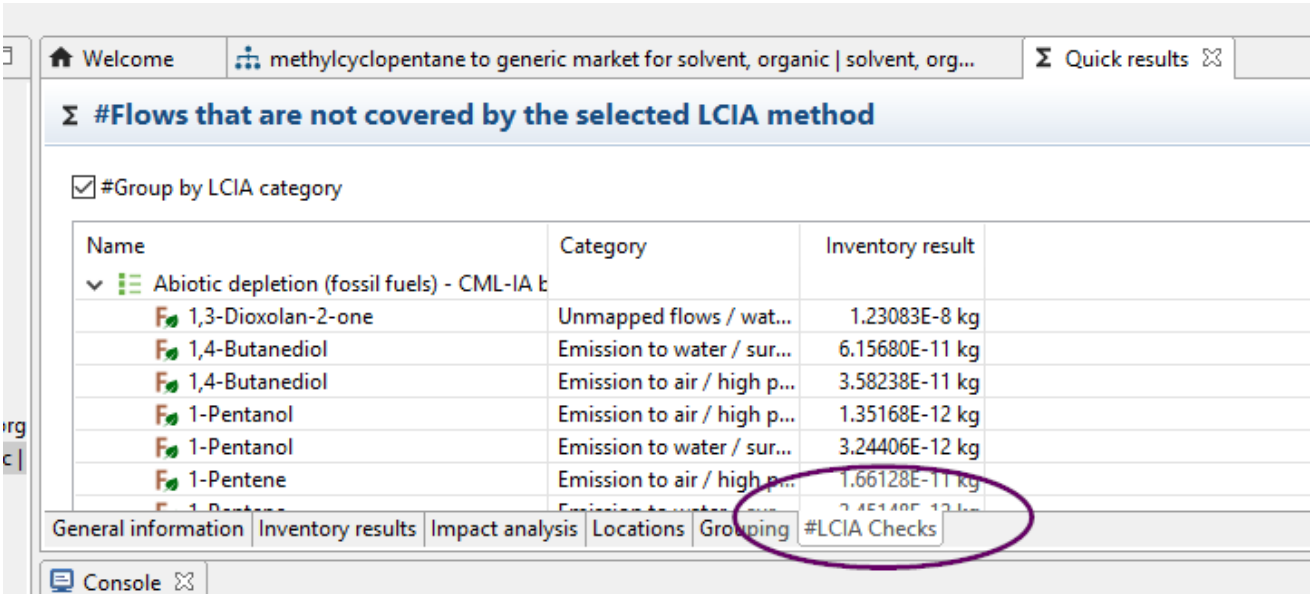


Figure 121: LCIA checks tab

In addition, when opening a process, the tab Impact analysis shows which LCIA categories of a selected LCIA method capture the flows of the inventory of the respective process. Optionally, zero values can be excluded.

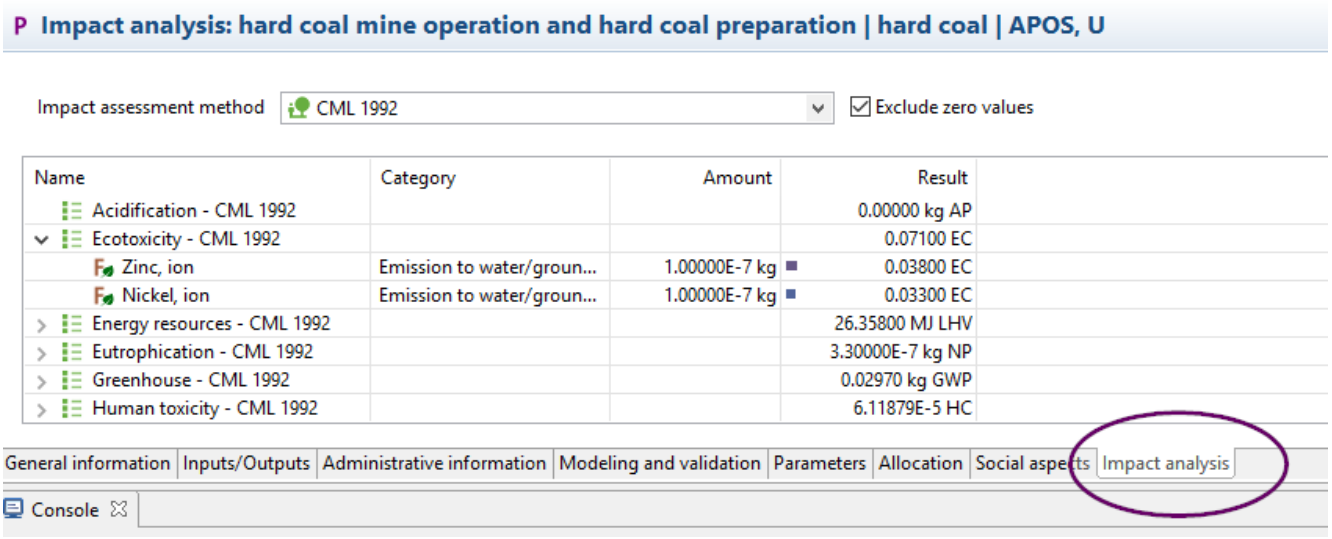


Figure 122: Impact analysis tab

Additionally, when opening a flow, the tab Impact factors lists the LCIA categories and respective LCIA methods which capture the flow as well as the corresponding impact/characterization factor.

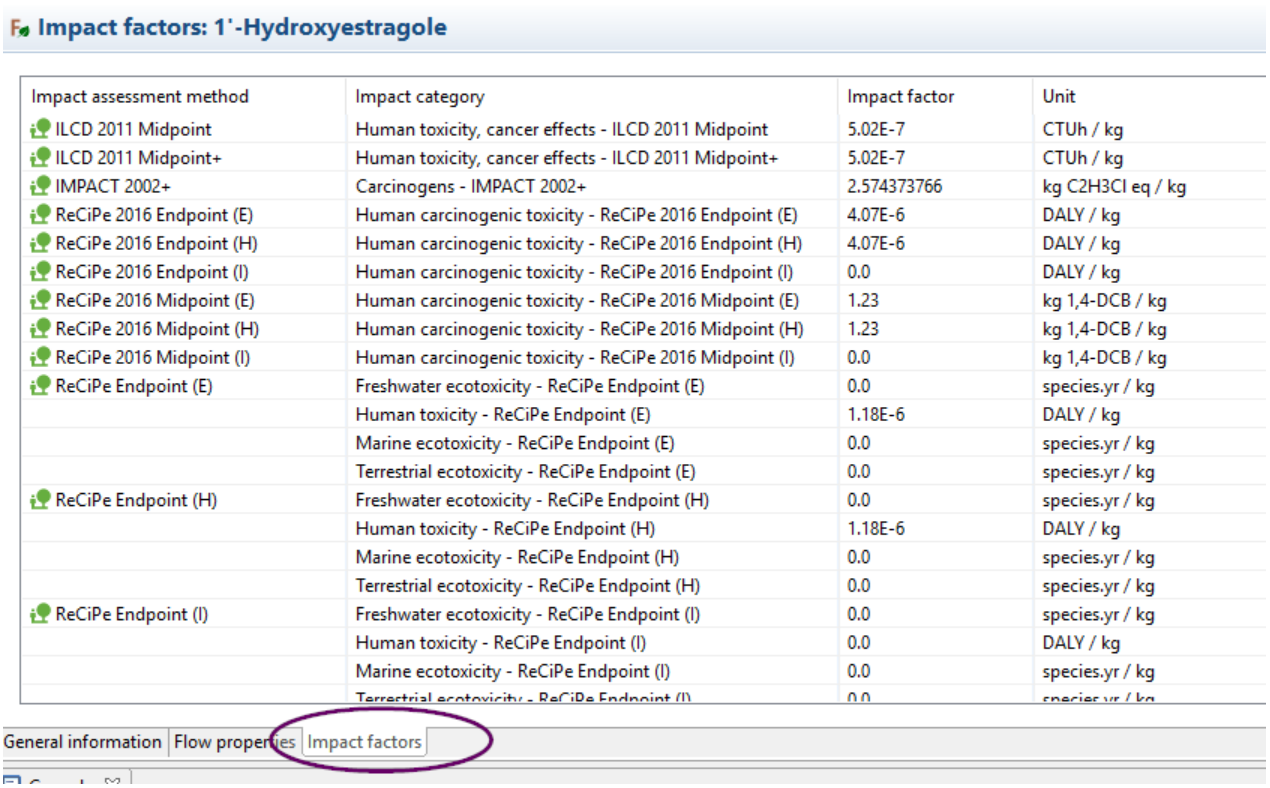


Figure 123: Impact factors tab

9.1.5 Locations

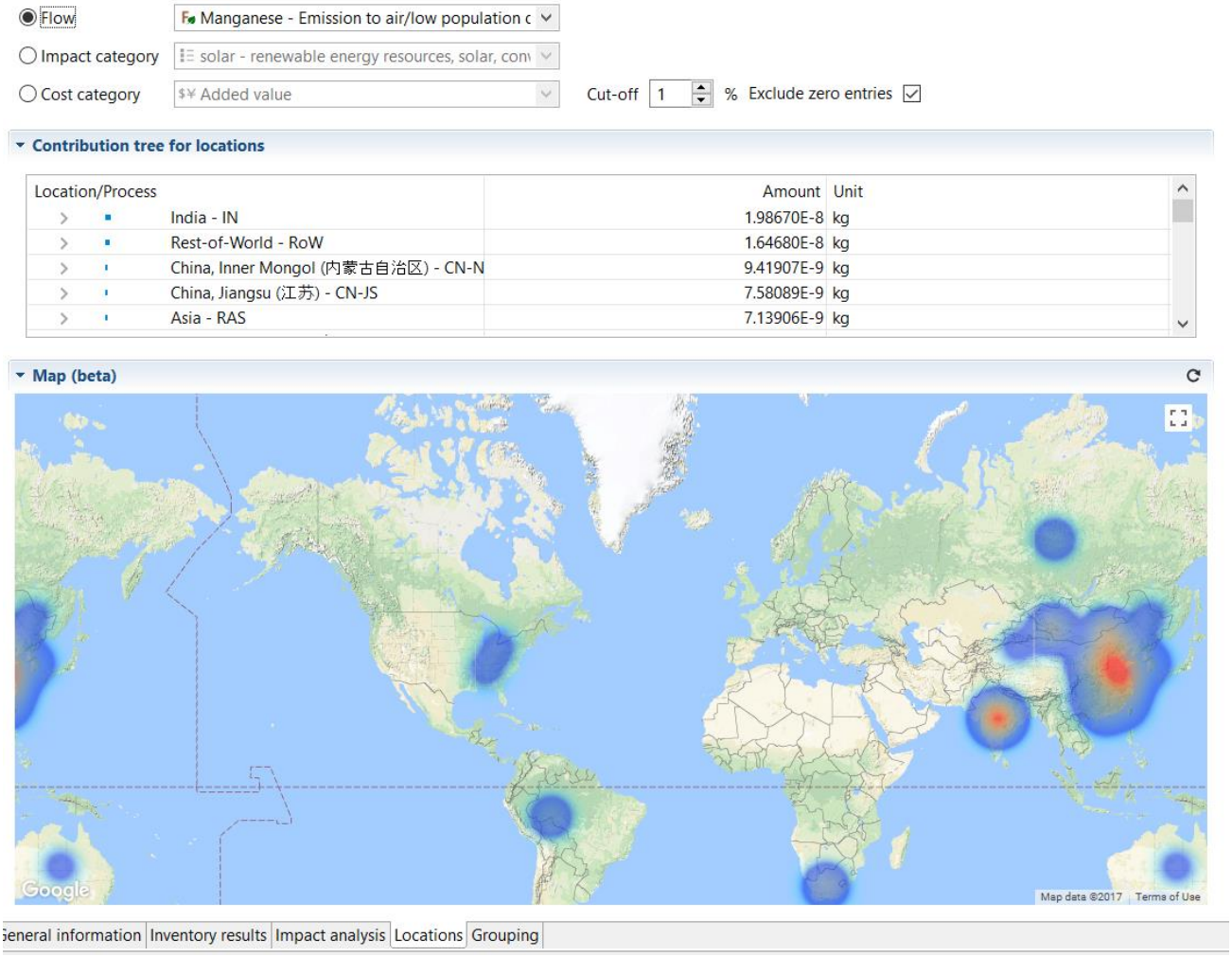


Figure 124: Quick results - Locations tab

The location tab illustrates specific information on localized flows and impact and cost categories (if the box “Include Cost Calculation” is checked when setting the calculation properties, see section 8.4). The locations are set in the flow level in openLCA.

9.1.6 Grouping

In openLCA, it is possible to group products to see the cumulative values for products. Please note, the values shown in the “Grouping” tab are the direct impacts (i.e. not including upstream values). To create a new group, select the green “+” icon in the right-hand corner of the editor. Then name the new group (Figure 125).

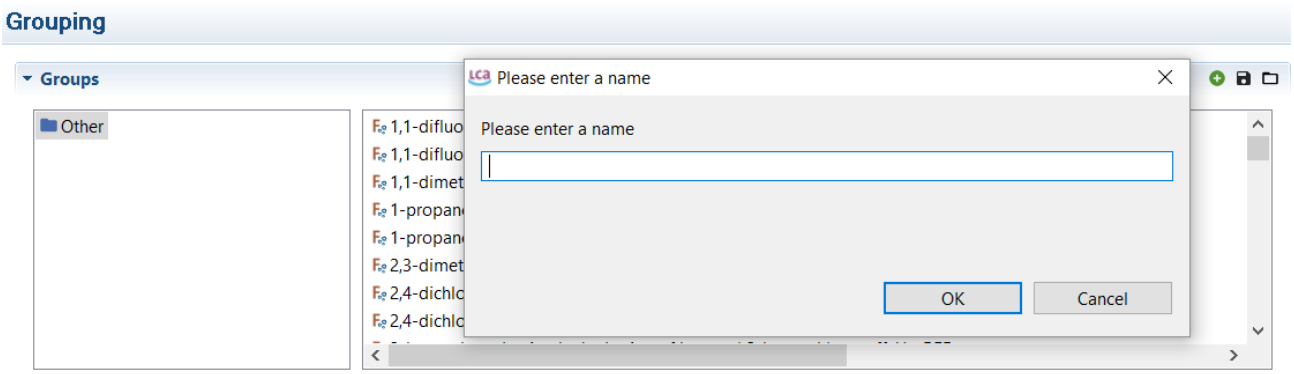


Figure 125: Creating a new group, step 1

To move a product to a group, right click on the flow and select “move” and then the desired group. To select more than one process at once click on one, hold the “shift” key on your keyboard, and then scroll and select another product. All products in between will also be selected.

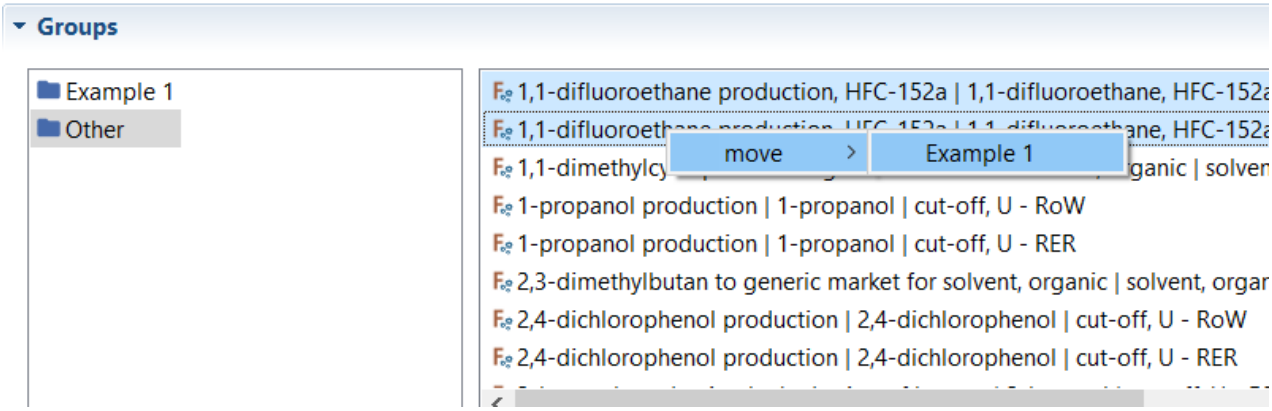


Figure 126: Creating a new group, step 2

Once you have created groups and added products to them, their contributions for specific flows and impact categories will be displayed in the table and as a histogram chart. Please note, the contributions displayed are direct, which means without upstream contributions. To take upstream contributions into account it is necessary to include all upstream processes in the group.

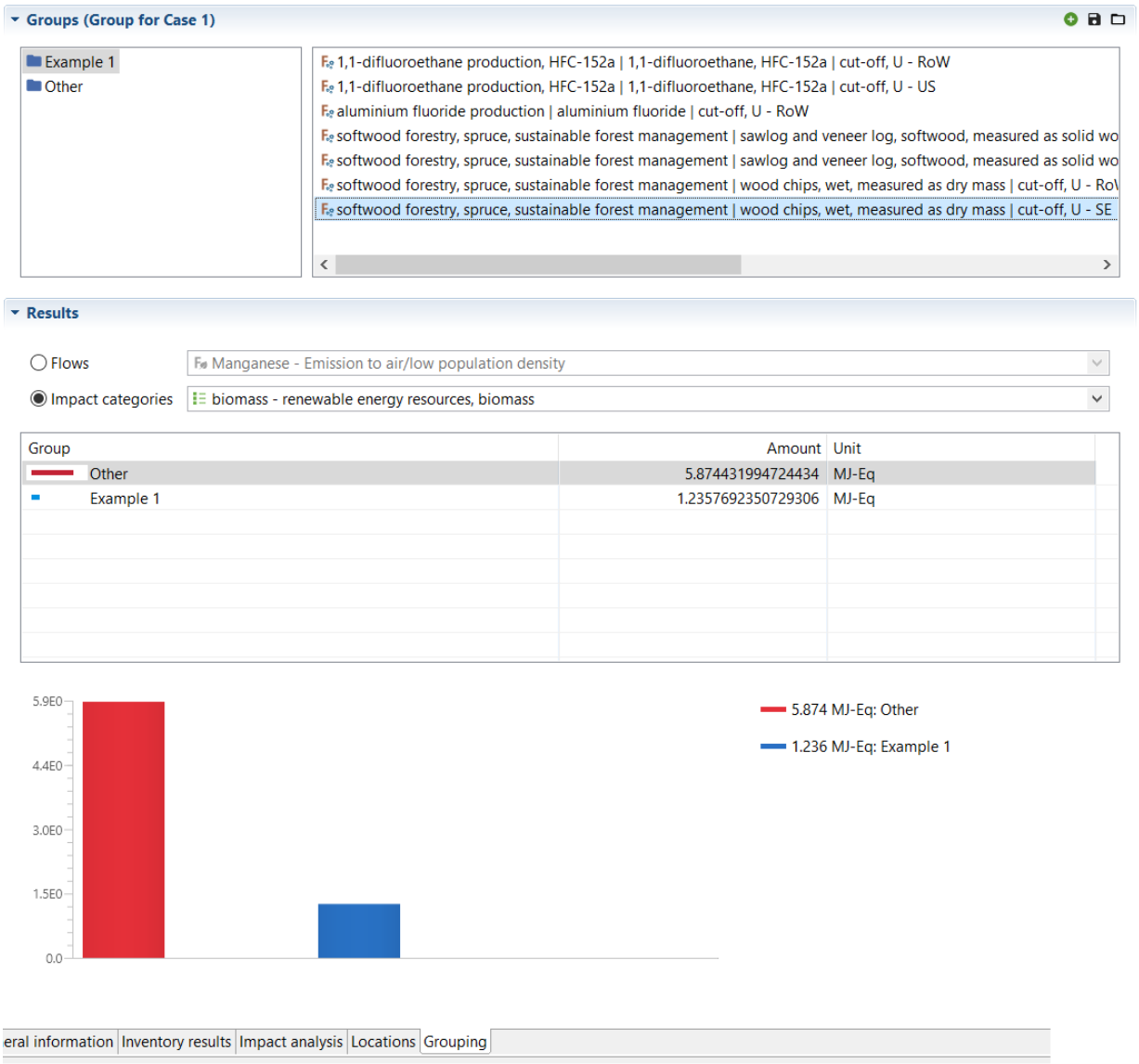


Figure 127: Quick results - Grouping tab

As it can be time-consuming to create new groups, you have the option to save groups in the Grouping tab. To do so, click on the “Save” icon in the top right-hand corner of the Grouping editor. Then give the group a name and press ok. These groups will be available in the Quick results editor each time you carry out a quick results or analysis calculation for any product system.

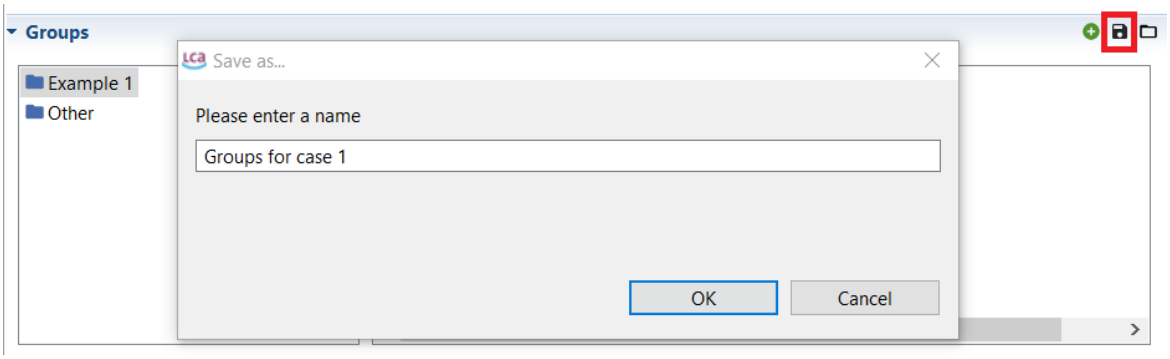


Figure 128: Saving groups

To open saved groups, click on the icon of the open folder in the top right-hand corner of the Grouping tab.

9.2 Analysis tab contents

The option “Analysis” provides information on direct as well as upstream impacts. All the tabs with information on direct impacts from the quick results are included in the analysis. Upstream impacts are shown in the tabs “Process results”, “Contribution tree”, “Sun Burst” and “Sankey diagram”.

9.2.1 General information

The General information tab of the Analysis contains the same information as for “Quick results”. For more information, see section 9.1.1.

9.2.2 Inventory results

The Inventory results tab of the Analysis contains the same information as for “Quick results”. For more information, see section 9.1.2.

9.2.3 Impact analysis

The LCIA Results tab of the Analysis contains the same information as for “Quick results”. For more information, see section 9.1.3.

9.2.4 Process results

The Process results tab is unique to the Analysis calculation. Here, both the direct and the total upstream contributions are displayed. In the section “Flow results”, select a process from the list and the input and output flows that contribute to that flow will be listed. Select a process from the list in the section “Impact assessment results”. All the impact categories of the chosen impact assessment method will be listed along with the values for which the selected process contributes to each impact category. Direct contributions/impacts are those resulting from that specific process only. Upstream total is the sum of direct and upstream contributions/impacts.

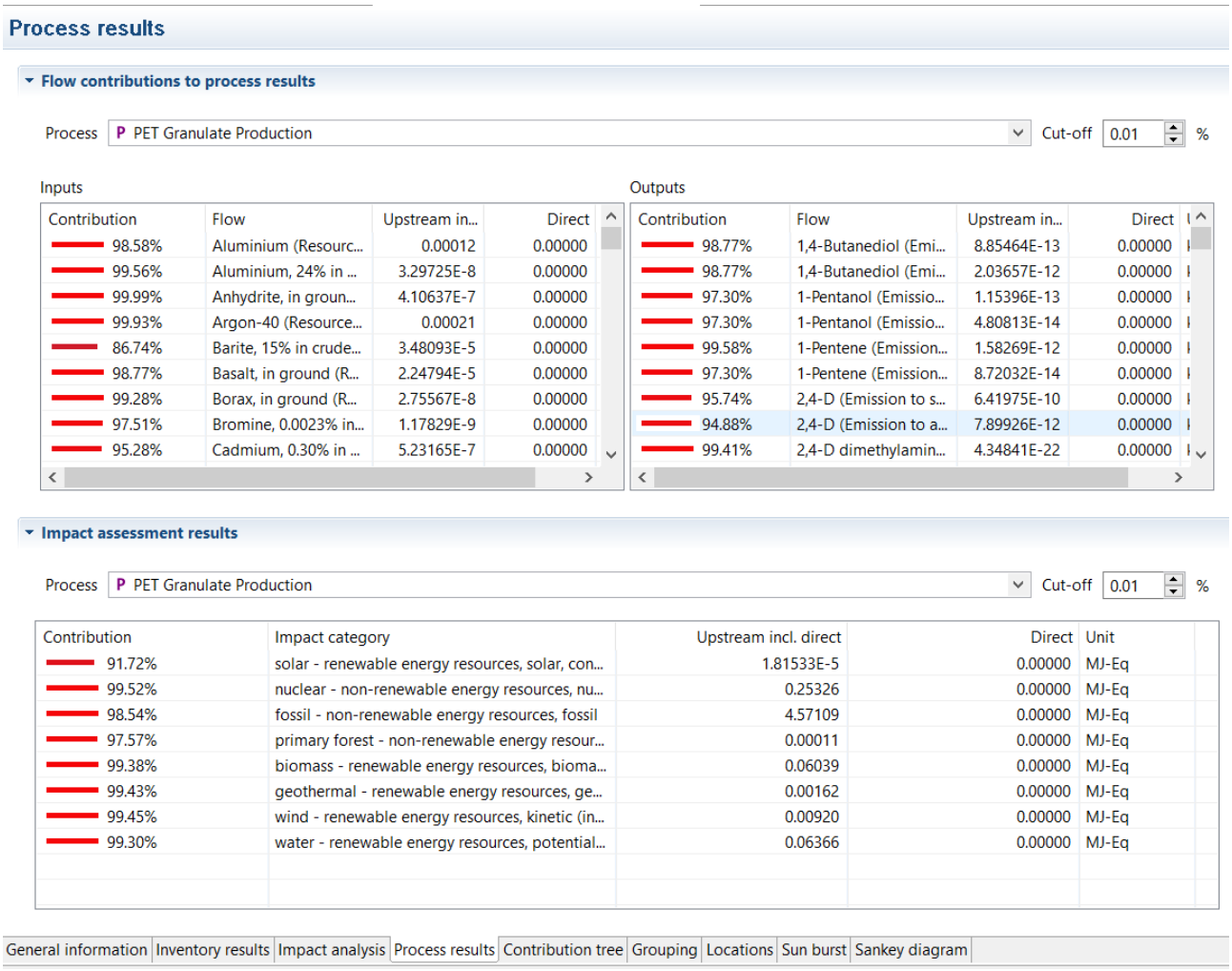


Figure 129: Analysis - Process results tab

9.2.5 Contribution tree

The contribution tree is unique to the Analysis calculation. It breaks down process contributions to flows and impact categories, displaying upstream totals.

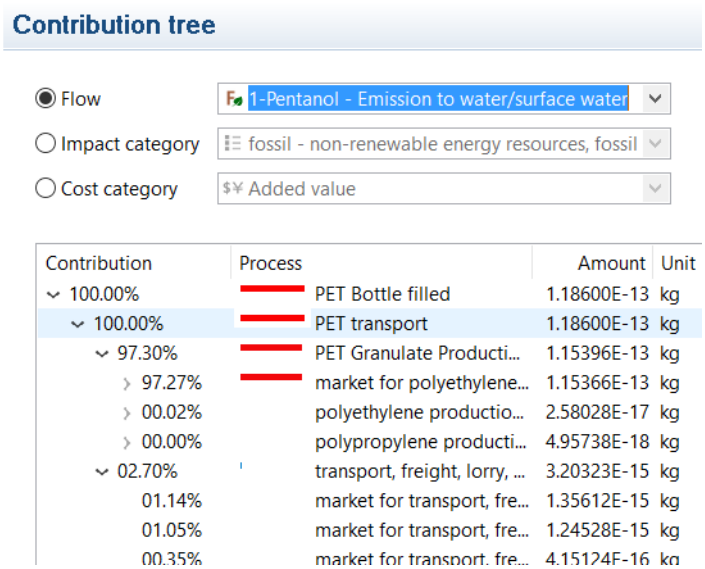


Figure 130: Analysis - Contribution tree tab

If the box “Include Cost Calculation” is checked when setting the calculation properties (see section 8.4), the contribution tree breaks down process contributions to cost categories (added value or net cost), displaying upstream totals.

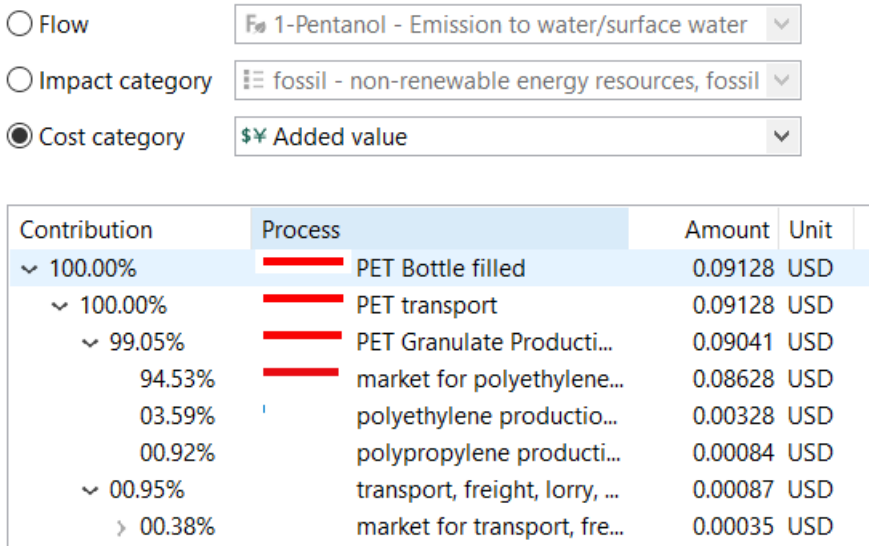


Figure 131: Analysis - Contribution tree tab, cost category

Also, the contribution tree can be copied and pasted to excel. To do so, select visible entries (ctrl+select to select several entries, ctrl+A to select all), right click and copy, and then paste to excel.

9.2.6 Grouping

The Grouping tab of the Analysis contains the same information as for “Quick results”. For more information, see section 9.1.6.

9.2.7 Locations

The Locations tab of the Analysis contains the same information as for “Quick results”. For more

information, see section 9.1.5.

9.2.8 Sankey diagram

The Sankey diagram is a graphical illustration of the impacts of processes in the product system to specific flows/impact categories. In the Sankey diagram the direct contribution of the process as well as the upstream total contribution of the process are displayed. To open a process in a new editor tab, simply double-click on it.

Right-click anywhere in the Sankey diagram editor and select:

- "Set Sankey diagram options" to select the flow or impact or cost category (If the box "Include Cost Calculation" is checked when setting the calculation properties, see section 8.4), and cut-off level to be displayed (Figure 132).

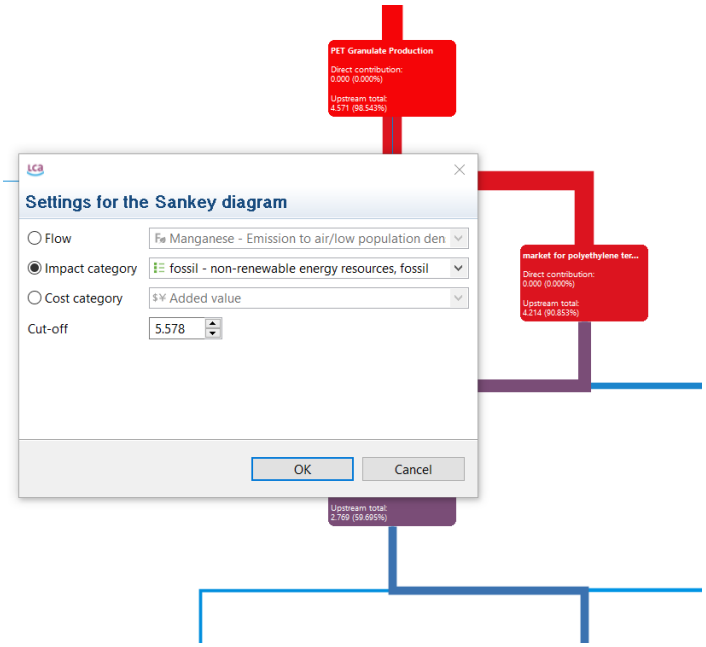


Figure 132: Analysis - Sankey diagram

- "Save as image" to save Sankey diagram as PNG file.
- "Open miniature view" which provides an overview of the diagram with zoom options. The blue box in the viewer shows the view displayed in the Analysis editor.
- "Routing enabled": Sort the layout of the connecting lines into a grid shape and bundle lines that have the same destinations to get a clearer overview.

9.2.9 LCIA checks

See section 9.1.4.

9.3 Regionalized LCA

Both the inventory and the impact assessment phase of a life cycle assessment (LCA) might be affected by site-specific conditions. As different spatial scales may be selected for defining each of these different

variables (e.g. per watershed, political unit, grid, etc.), as well as for defining the process locations, it is fundamental in a regionalized impact assessment to be able to deal with multiscale systems without compromising the correctness of the results. To overcome this challenge, geographic information systems (GIS) can be used not only for the calculation of regionalized impact factors but also for creating site-specific inventories and matching each of these. Since version 1.7.0, openLCA includes the functionality for handling GIS data, allowing the user to include this type of data in the process locations, as well as for defining site-specific impact factors in the method.

9.3.1 Locations

The list of locations available in the database is shown in the navigation tab under Database > Background data > Locations. It is also possible to create a new location by right-clicking on Locations and selecting the first option “New location” (specification of a name and a code is required).

Figure 133: Regionalized LCA – locations

KML data can be added to each location (by drawing polygons, lines and points), see Figure 134 . Other possibilities are importing kmz/xml files with geographic data or writing the coordinates in the “Text editor”.

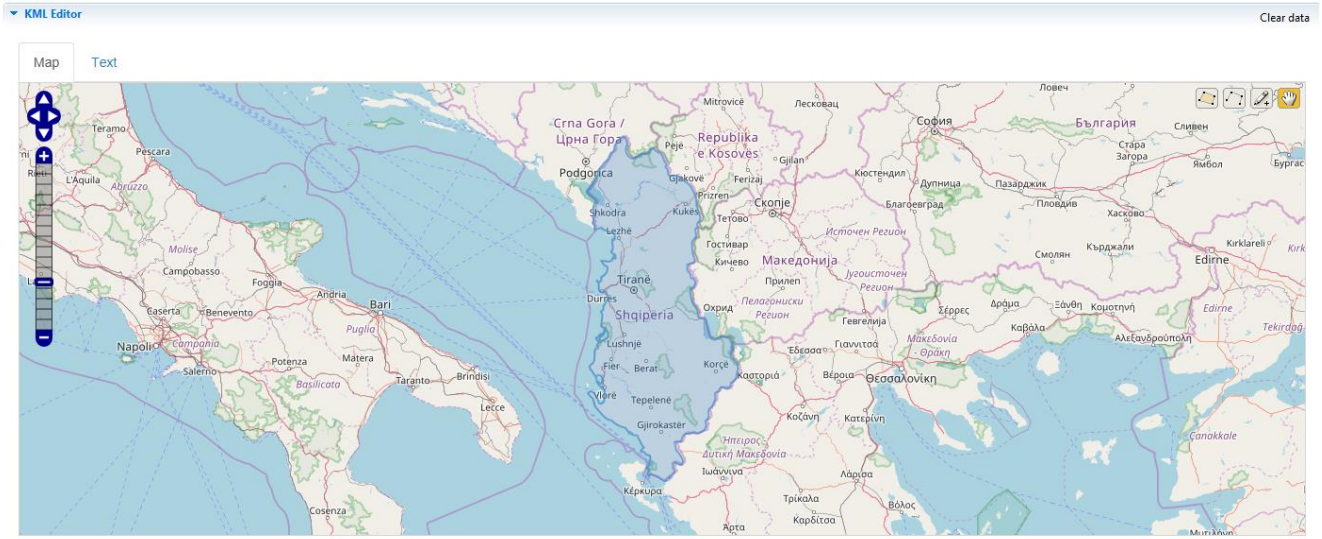


Figure 134: Regionalized LCA - KML editor

9.3.2 Calculation framework

Data for regional characteristics is contained in shapefiles, which can be imported to openLCA and stored in the database, see section 7.3.5.

The calculation framework sees the intersection between shapefiles features and process geometries. In this way process locations are linked to LCIA methods spatial units.

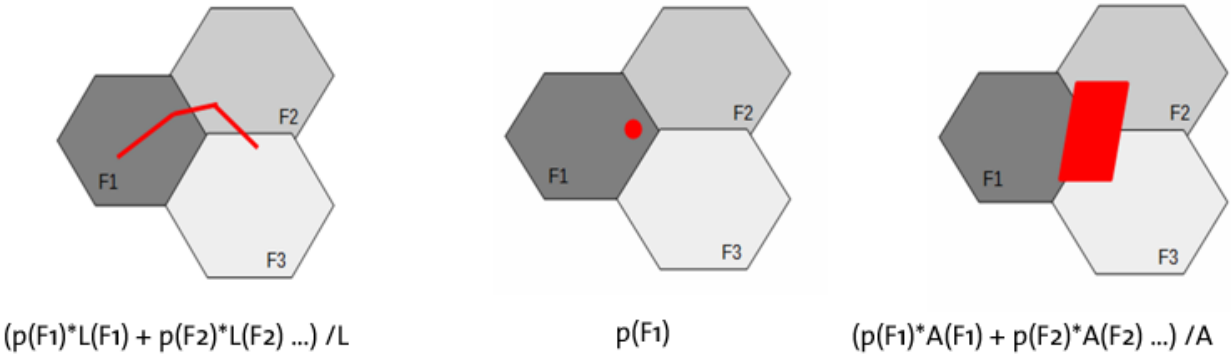


Figure 135: Regionalized LCA - Calculation framework

9.3.3 Parameterization of LCIA methods

Regional characteristics affecting the Characterization Factors (CFs) in LCIA Methods can be defined with parameters. First, shape files must be bound to parameters. Shape file parameters can be added to the LCIA method input parameters table by right clicking on the name of the shape file parameter and selecting “Add to method parameters”. The parameter will be automatically added to the “Input parameters” area in the “Parameters” tab. As it can be observed in Figure 136, the “External source” field refers to the shape file which provides the data for the parameter.

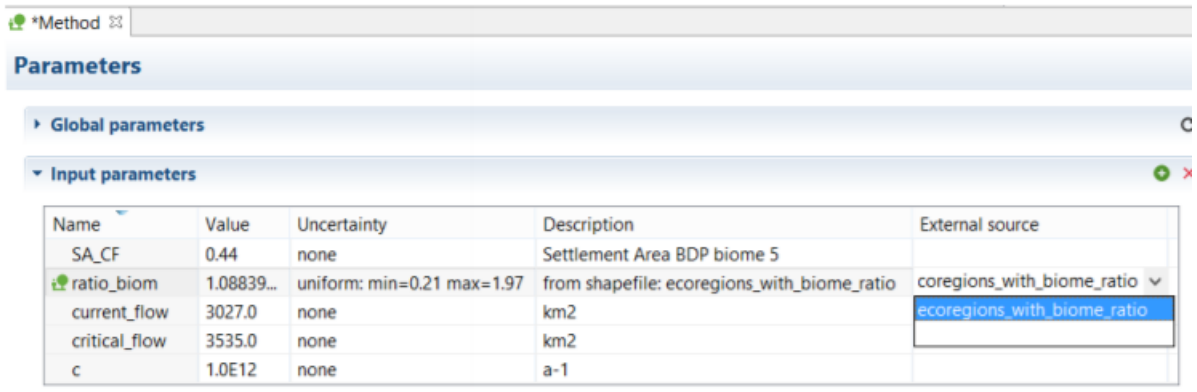


Figure 136: Shape file parameter in the "Parameters" tab of the LCIA method editor

The mean value, which is the default value to be used when no regionalized LCIA is performed, can be modified if necessary. The uncertainty is automatically calculated from the values contained in the shape file. If there is more than one shape file containing a parameter with the same name, the external source can be changed in the “Parameters” tab using the drop-down menu in the correspondent row of the column “External source”. The mean value and the uncertainty will be recalculated with the selected shape file.

At this point formulas for calculating the characterisation factors (CFs) in the LCIA method can be defined.

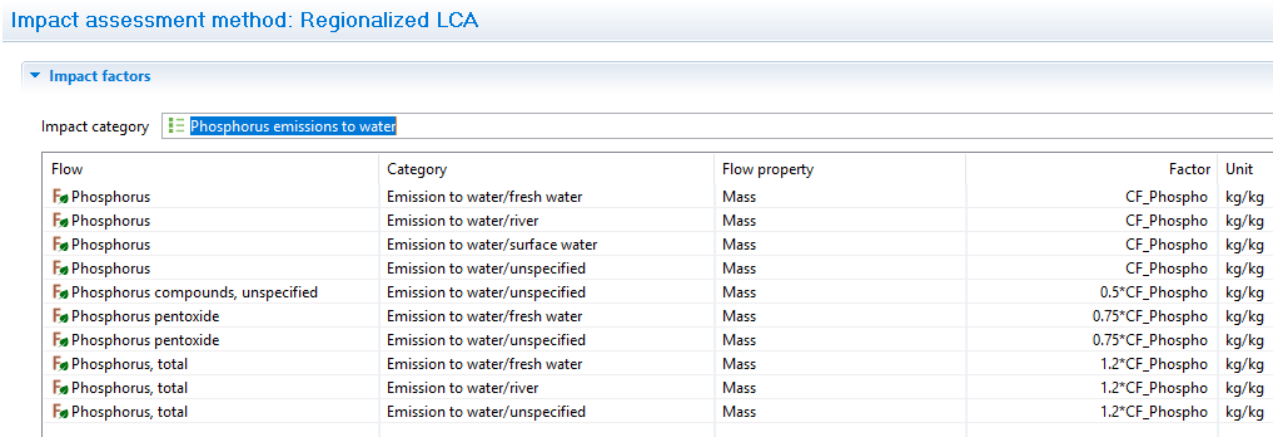


Figure 137: Regionalized LCA - Parameters applied to CFs

9.3.4 Calculation of regionalized LCA

To perform Regionalized LCA, “Regionalized LCA” must be specified when setting the calculation properties, see section 8.4.

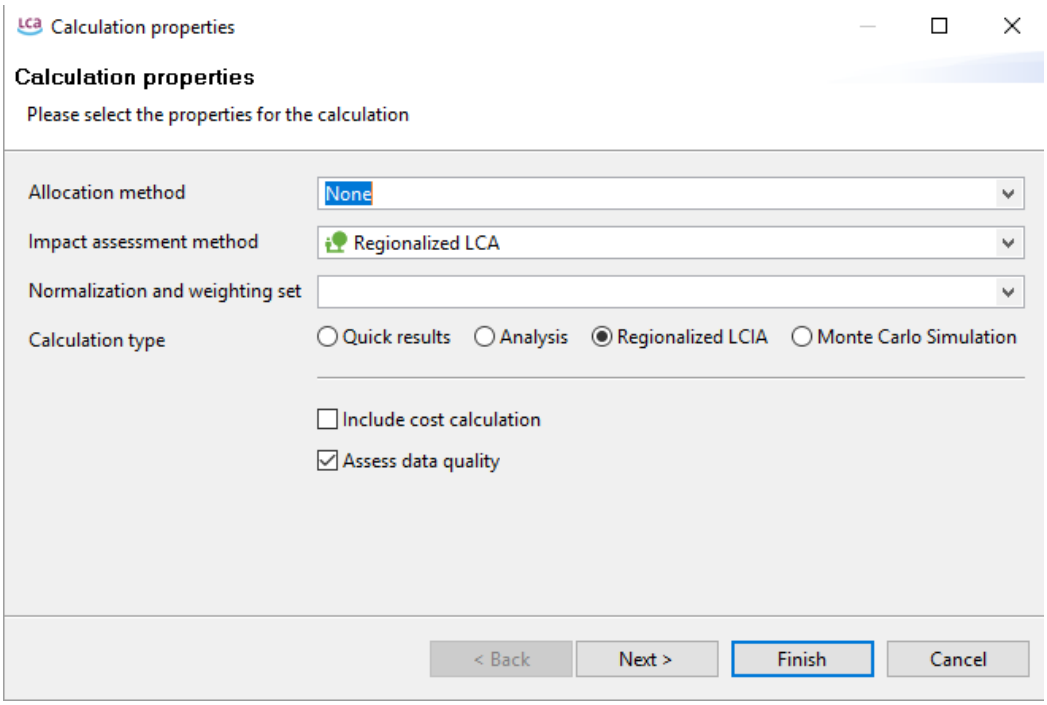


Figure 138: Calculation properties for Regionalized LCA performance

The sections of the Regionalized LCA analysis results are the same as the Analysis results, see section 9.2. In addition to "General information, inventory results, impact analysis, locations, process results, contribution tree, grouping, Sun burst, Sankey diagram", the section "Result map" is added. This tab illustrates specific information on localized flows, impact and cost categories.

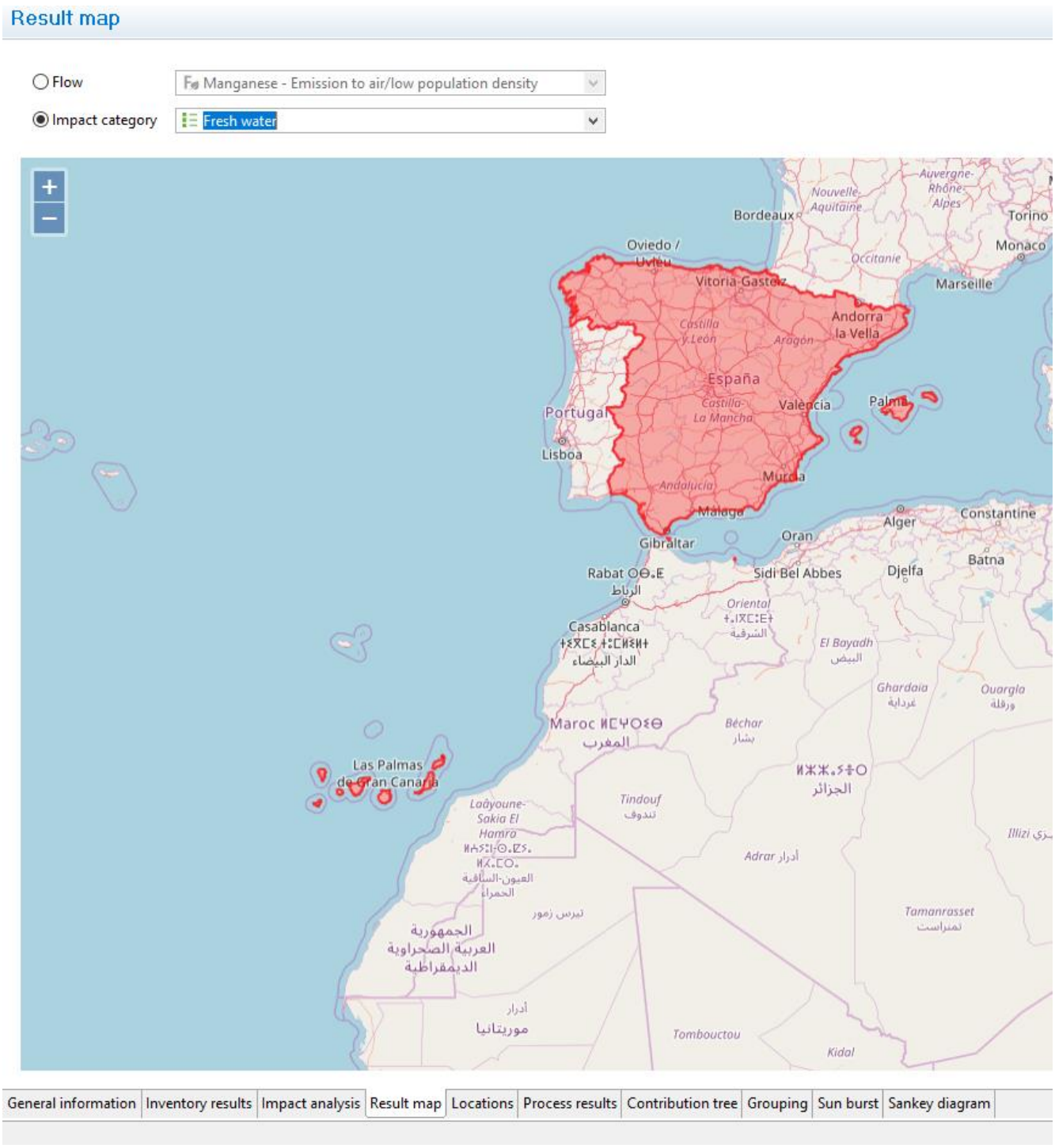


Figure 139: Regionalized LCA - Result map

9.4 Monte Carlo Simulation

A Monte Carlo simulation varies entry data of the model calculation randomly according to the uncertainty distributions. Then, an uncertainty distribution for the calculation result is provided. In general, several thousand iteration passes are usually required.

It is also possible to compare two processes in Monte Carlo simulation if you create a new process (see Figure 140) and then product system where one process is subtracted to the other to avoid double

counting of uncertainties.

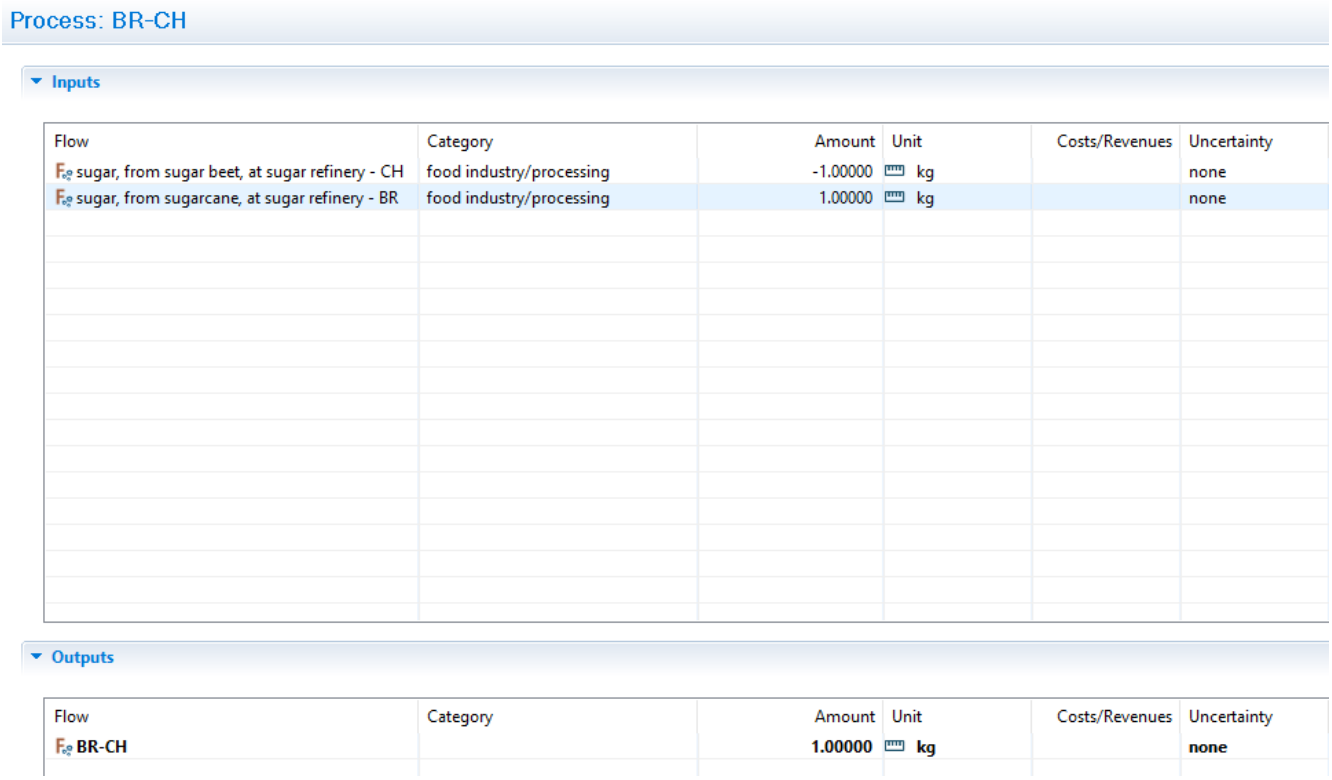


Figure 140: Monte Carlo simulation – creation of a process subtracting one process to the other for future comparison in Monte Carlo simulation

9.4.1 Adding uncertainty information

The first step in openLCA is to add uncertainty data to all input and output flows in the processes (distribution, standard deviation, min/max, etc. It is also possible to define uncertainty data for parameters as well as LCIA characterization factors.

To add uncertainty data to flows, open a process and select "Edit" in the uncertainty field of a process:

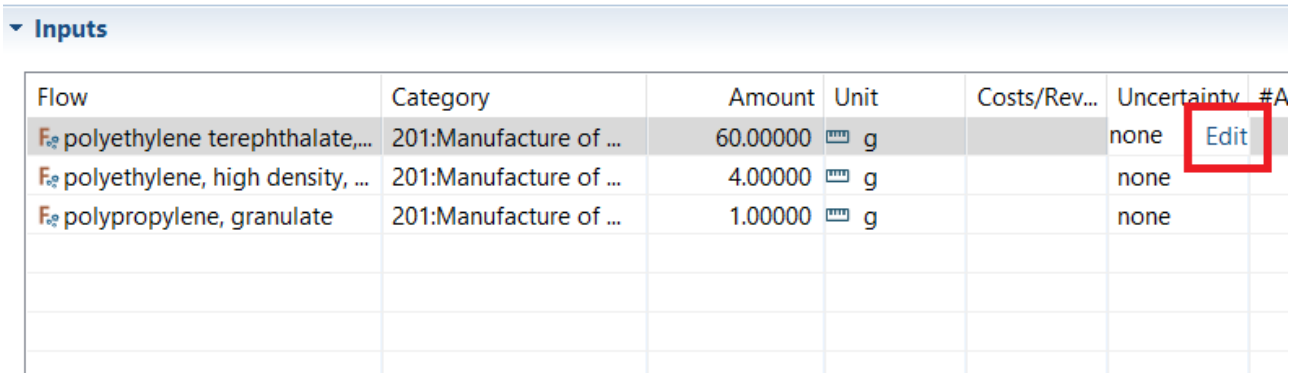


Figure 141: Adding uncertainty information, step 1

Then, select the uncertainty distribution (no distribution, logarithmic normal, normal, triangle or uniform) as well as the geometric mean and geometric standard deviation:

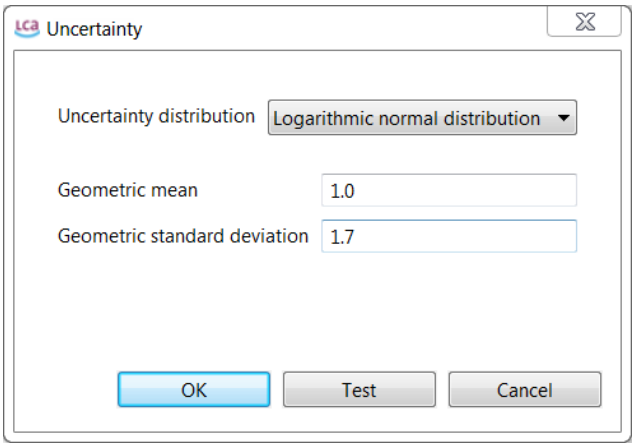


Figure 142: Adding uncertainty information, step 2

To add uncertainty information to LCIA characterisation factors, click on the "Edit" button in the "Impact factors" tab of the impact assessment method:

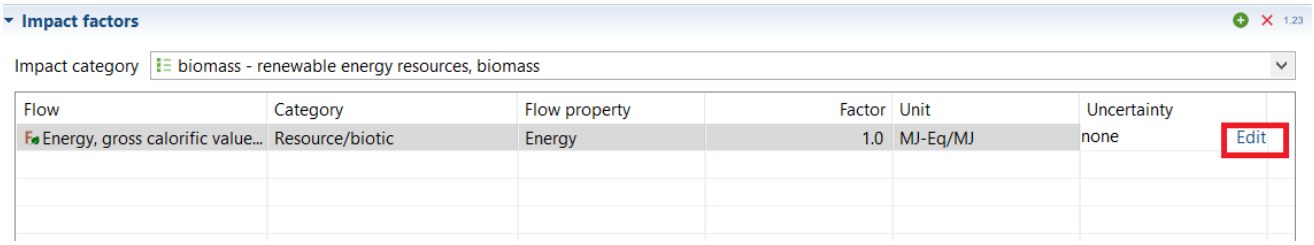


Figure 143: Adding uncertainty information to LCIA characterisation factors

Uncertainty information can be added to parameters in the same manner (in the parameters tab of processes, LCIA methods and product systems).

9.4.2 Starting the Monte Carlo Simulation

To carry out Monte Carlo Simulation, select it in the wizard when calculating a product system, as well as the number of simulations to be carried out. Then, uncertainties are calculated for each flow and impact category.

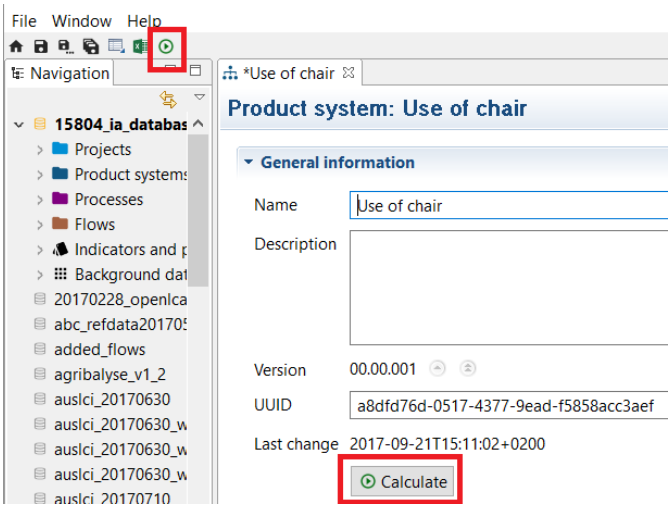


Figure 144: Monte Carlo Simulation, step 1

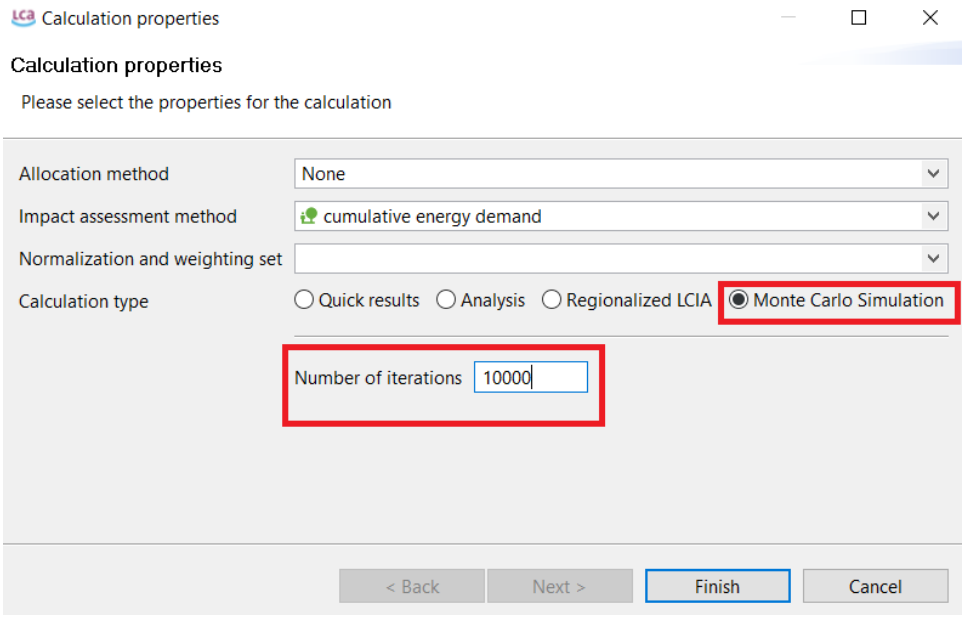


Figure 145: Monte Carlo Simulation, step 2

The Simulation will open in the editor. Select “Start” to begin calculations (Figure 146). The calculation time required depends on database and product system complexity, and selected number of simulations.

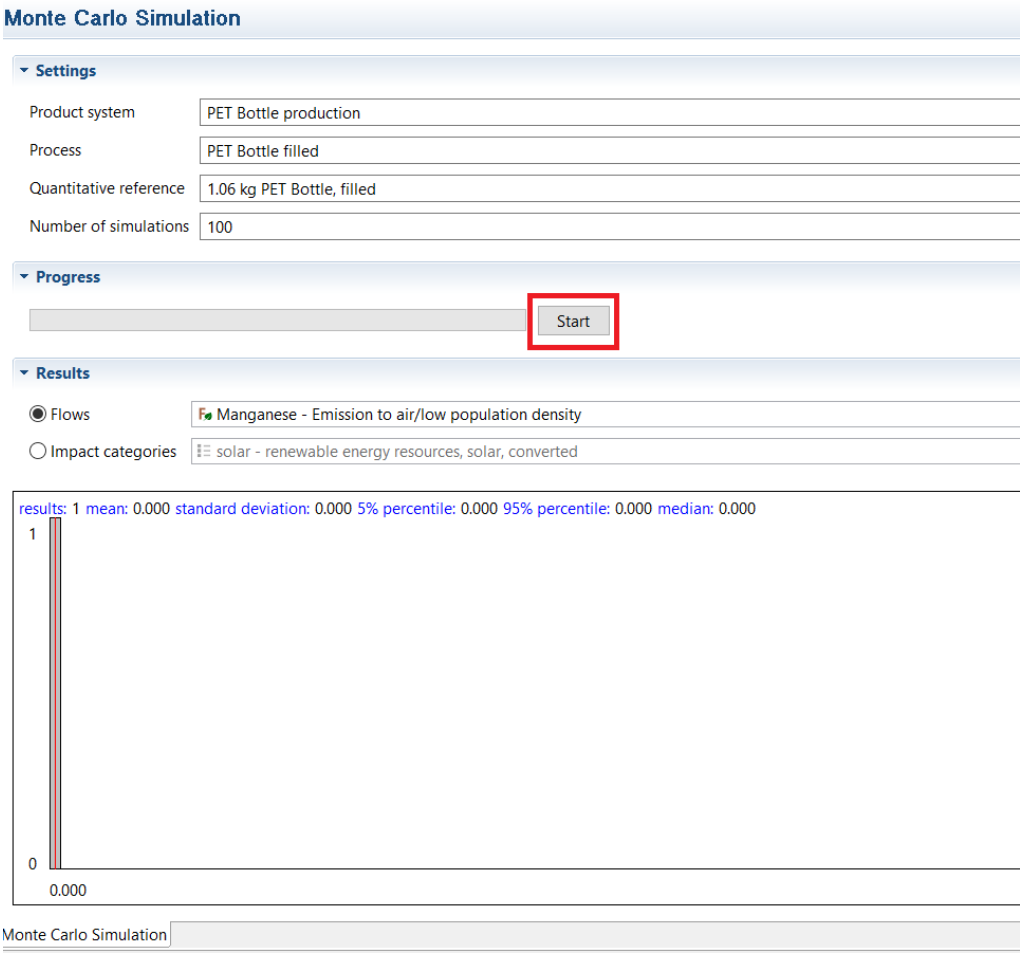


Figure 146: Monte Carlo Simulation, step 3

The results for each flow and impact category will be displayed while the simulation runs.

9.4.3 Monte Carlo Results

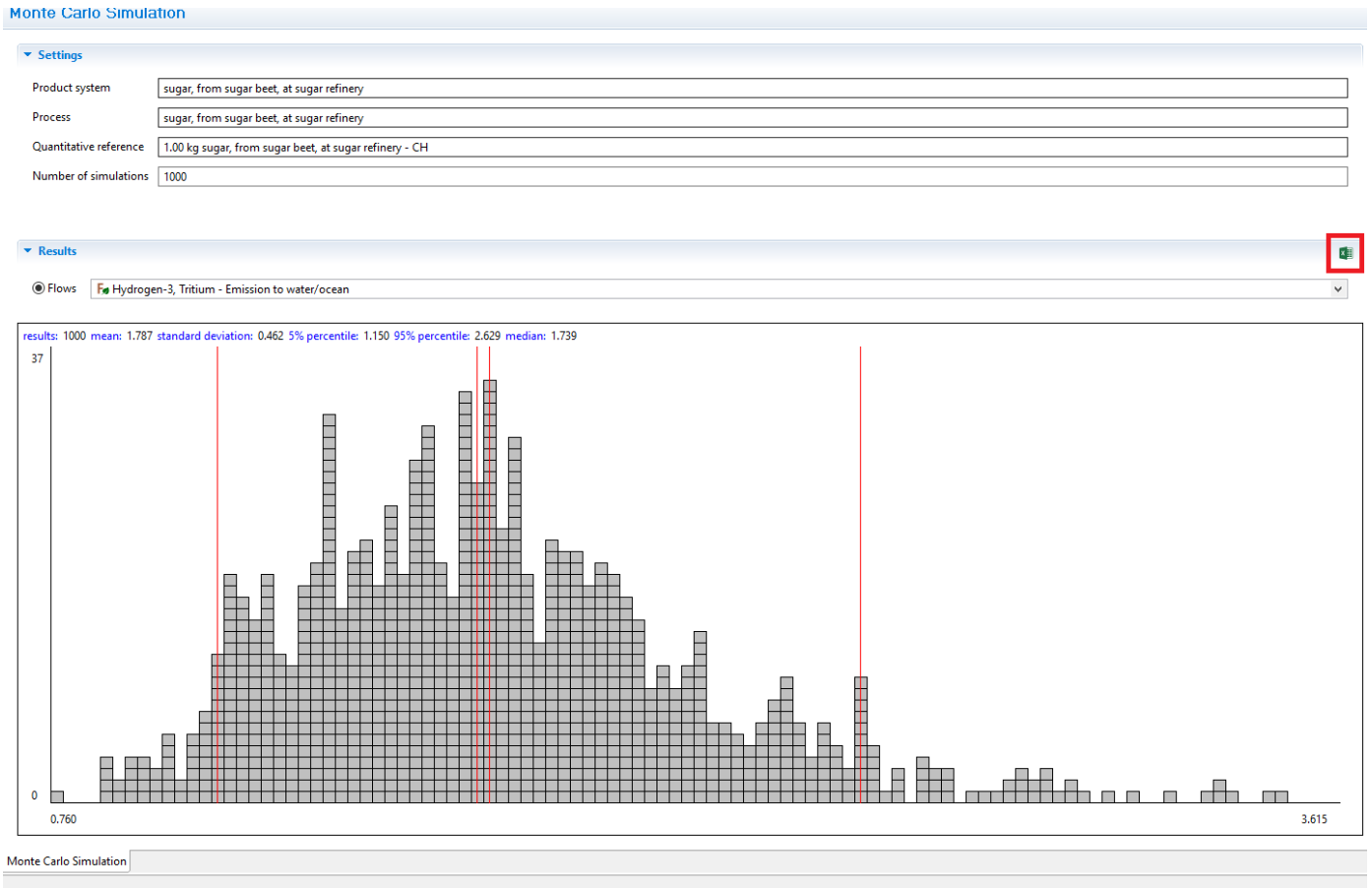


Figure 147: Monte Carlo Simulation results with the progress of the calculation

The results of the simulation can be exported as an excel document. Simply click on the excel icon on the right-hand side of the editor (as shown in Figure 147).

The results of all processes and sub product systems that are part of a product system can be displayed separately in the Monte Carlo Simulation tab. For easy navigation, individual processes can be pinned (Figure 148).

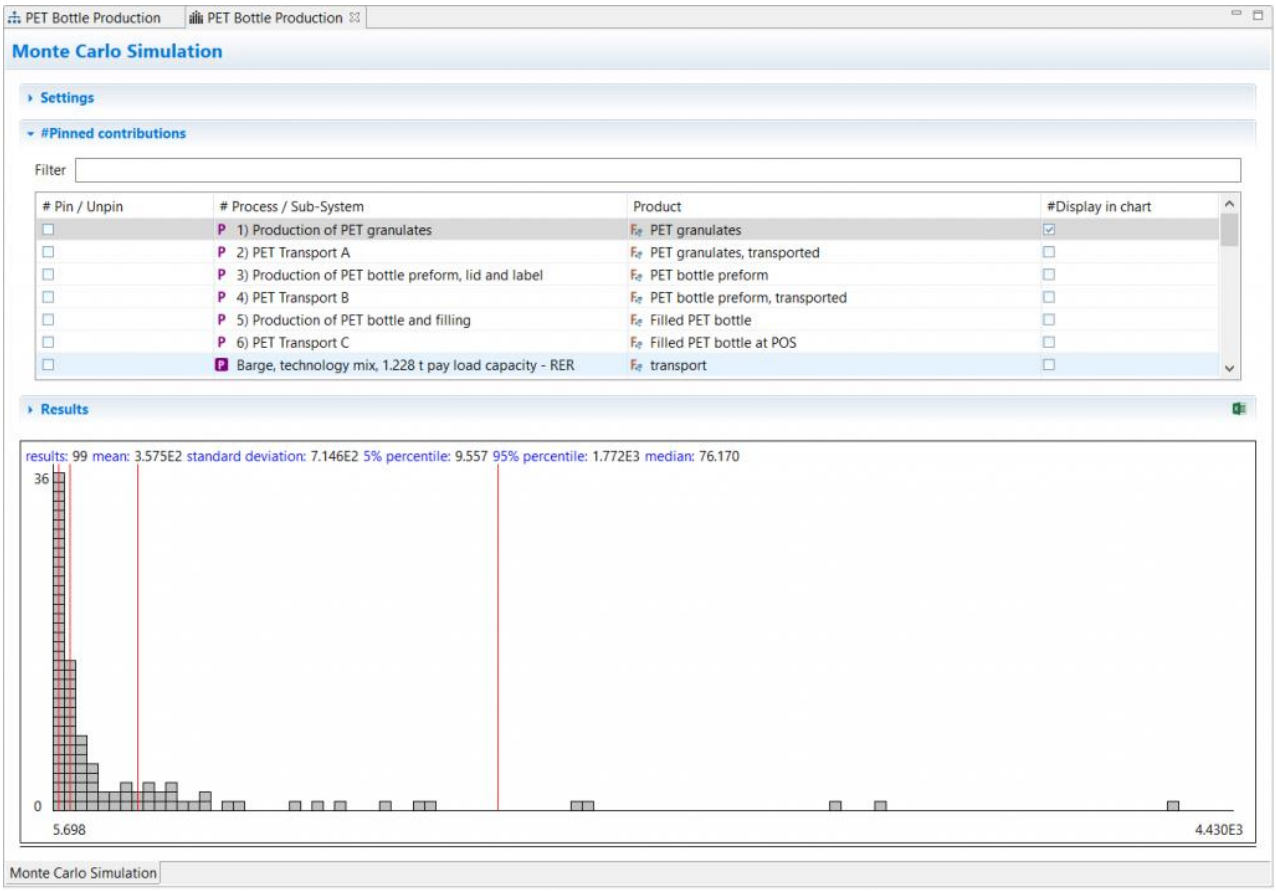


Figure 148: Pin/unpin processes or subsystems

9.5 Life Cycle Costing in OpenLCA

Information on Life Cycle Costing is already provided in several previous sections when the topic comes to attention. In this section a general view of the topic is presented.

Costs are modelled in the software as associated to products, waste or elementary flows, which are inputs and outputs of processes (Figure 149). To know more about flows and processes see sections 5 and 6 respectively. There is no need of the creation of a method for the LCC calculation. Furthermore, costs can be positive or negative and a negative cost is regarded as an added value.

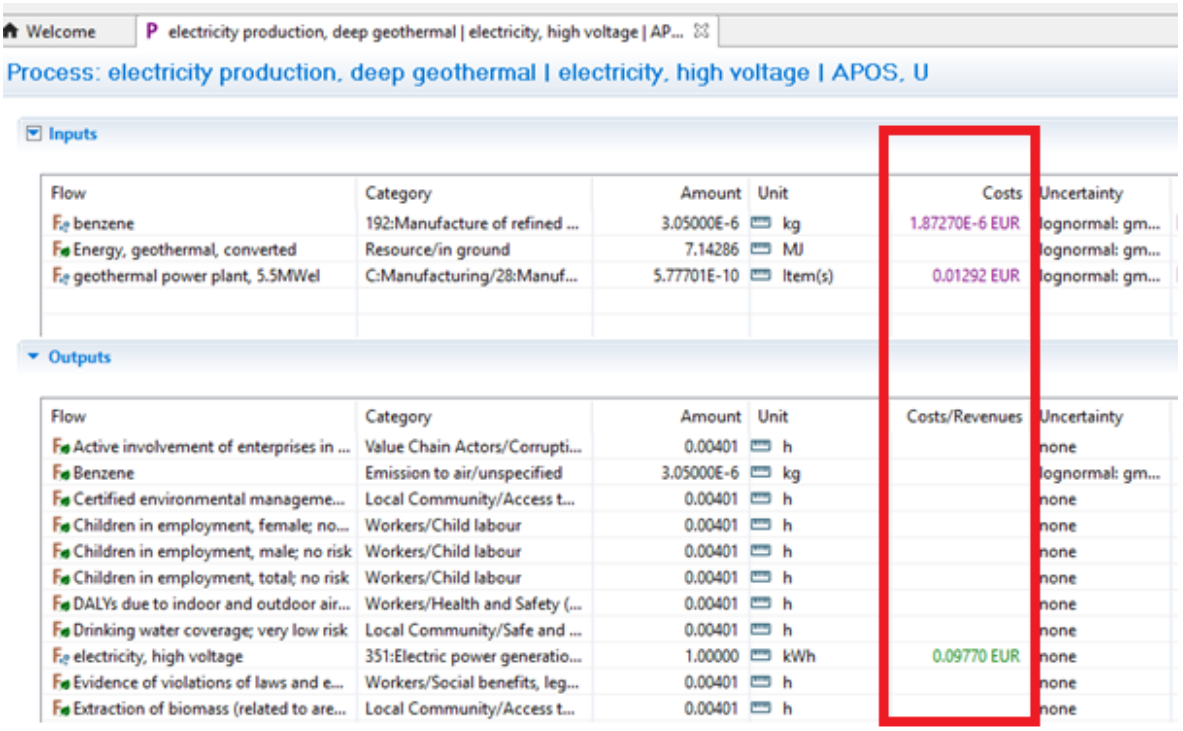


Figure 149: Specification of costs and revenues for input and output flows in a process

In case of multi-output processes, consideration of the costs of by-products depends on the choice of allocation options when setting the calculation properties (see section 8.4). If no option is selected the price of the by-products is considered as a revenue, while if a type of allocation is chosen the allocation factors are applied to the exchanges and the by-product cost is not considered. The third possibility is to select the option “as defined in processes”: as before, if a type of allocation is chosen the allocation factors are applied to the exchanges and the by-product cost is not considered; on the contrary, if no allocation is selected all costs are calculated excluding those from the by-products.

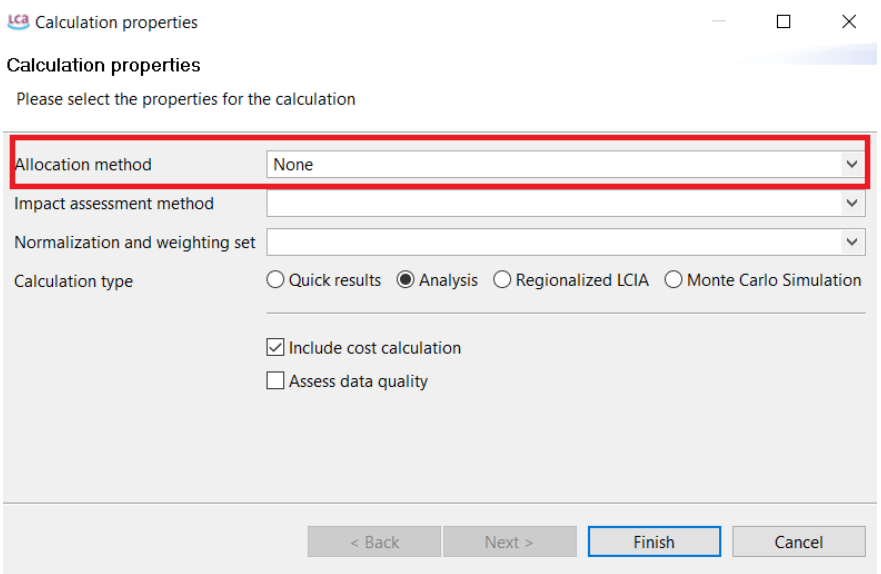


Figure 150: LCC - Choice of allocation method when performing LCC

Another important issue is the consideration of market variability through uncertainty models. OpenLCA

presents a column to assign uncertainty to input and output flows of a process, but not to the price directly. This can be solved with making the price a parameter (see section 6.2.6) and assigning uncertainty directly to it (Figure 151).

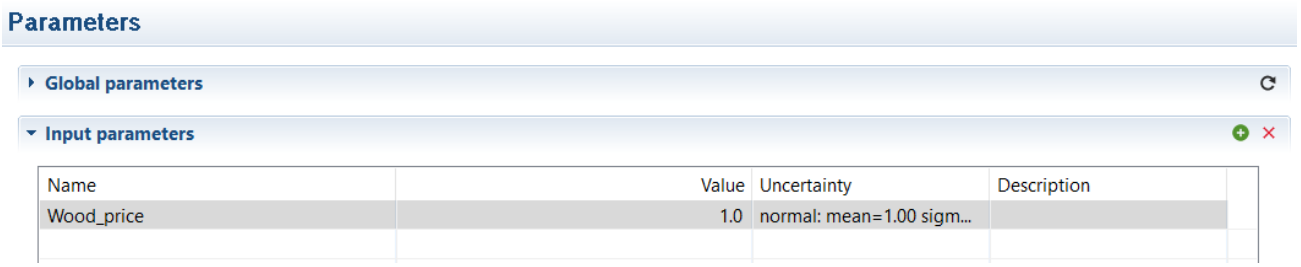


Figure 151: Uncertainty given to a price parameter

9.5.1 Available Data

Each database in EcoSpold or ILCD format can be imported into the software (see section 4.4). Ecoinvent database v. 3.3 in openLCA format provides prices for all products except for waste materials and their disposal. Otherwise, prices can be inserted manually in the input/output section for each process.

Several currencies are available in the database and for an entire database one currency can be selected as a reference for all the others.

The software allows to model different prices for the same material referred to different processes or countries, as the price per reference unit associated to a material is open and can be changed in the processes of the product system. In this way, there is no need of the creation of the same material with a different name and price associated.

9.5.2 Results and analysis available

The software displays both costs and added value (Figure 152), together with inventory and impact assessment results.

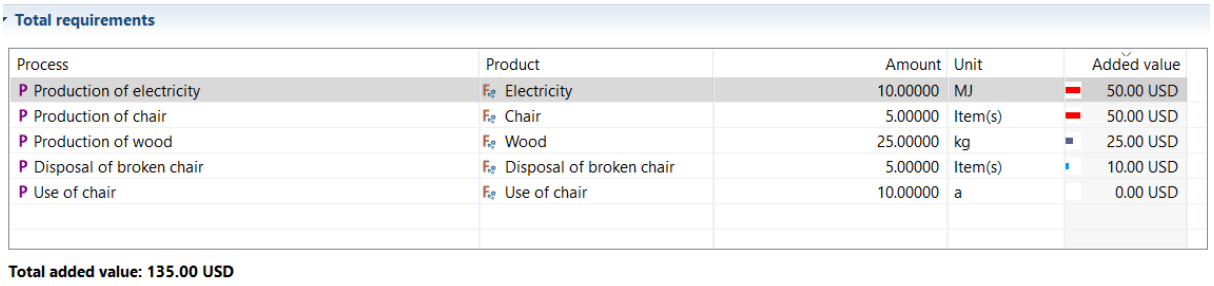


Figure 152: Calculation of total and stage added value

With this, the software offers a good combination of environmental assessment and cost analysis. Furthermore, LCC can be performed stand-alone by selecting no method when the impact assessment is run (see Figure 153).

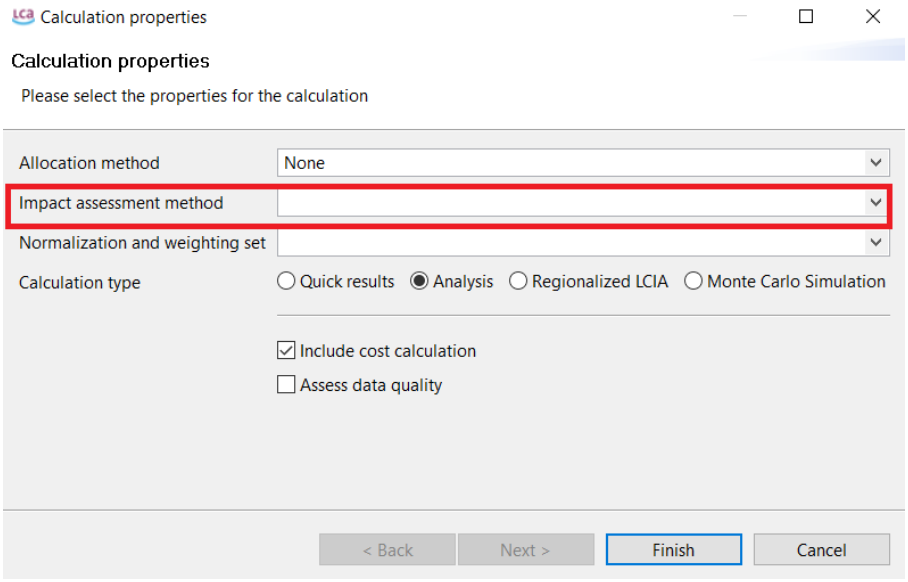


Figure 153: Performance of LCC stand-alone, no impact assessment method is selected

OpenLCA is quite flexible as the software allows specification of separate costs for each exchange in a process and the section called “contribution tree” offers a good overview of the added value or net costs expressed in percentage (Figure 154).

Figure 154: Contribution tree referred to the selected cost category

10 Projects

In general, projects can be used to compare product systems.

10.1 Creating a new project

To create a new project, begin by right-clicking on the "Projects" folder and select "New project".

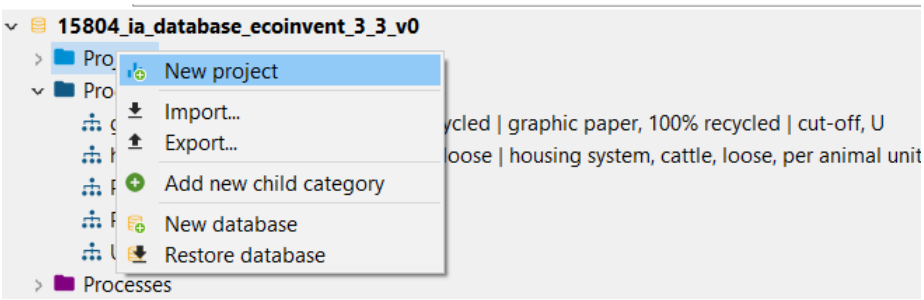


Figure 155: Creating a new project, step 1

Name the new project and provide a description (optional):

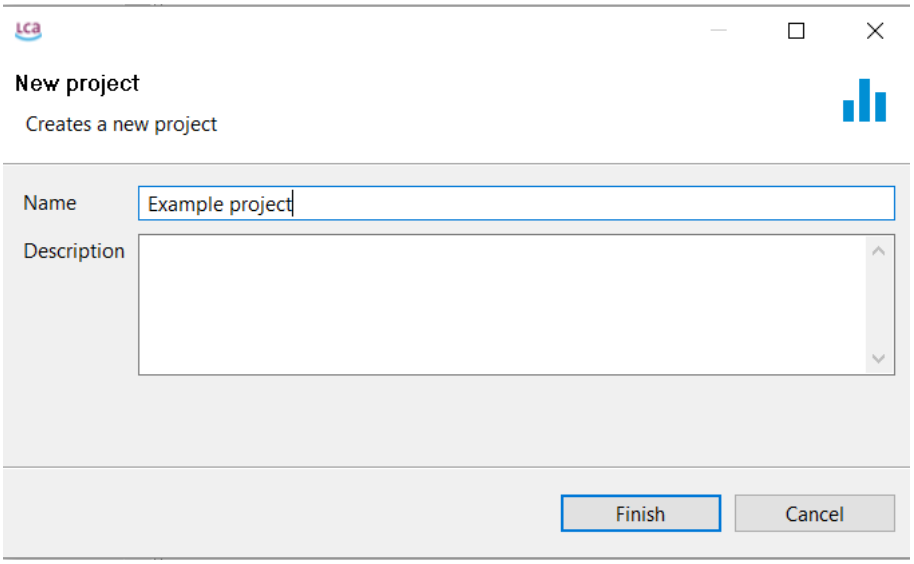


Figure 156: Creating a new project, step 2

The new project will open in the Editor.

10.2 Project tab contents

10.2.1 Project setup

In the “General information” section you can type in a name for the project and a description. Once you have configured your report in the “Project setup” and “Report sections” tabs, click on the “Report” button to have the software carry out the calculation and create your report.

In the “LCIA Method” section select the method for the calculation as well as a normalization and weighting set, if applicable (you will be able to select from the methods you have imported in that database). You also have the option to select the impact categories you would like to have displayed in the report (Figure 157).

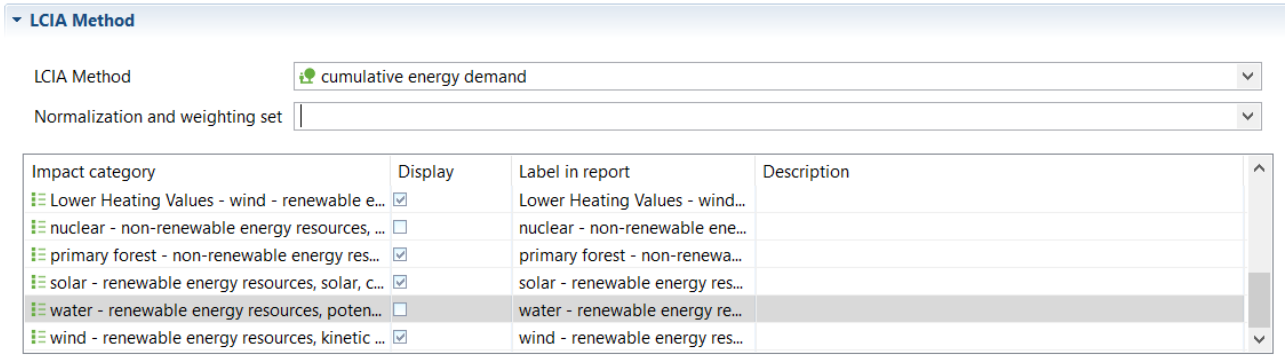


Figure 157: Project setup, LCIA method selection

In the "Compare product systems" section, click on the green "+" icon on the right to add product systems you want to compare. Multiple product systems can be added simultaneously to the Compared product systems overview/list of a project. In addition, drag and drop is supported too.

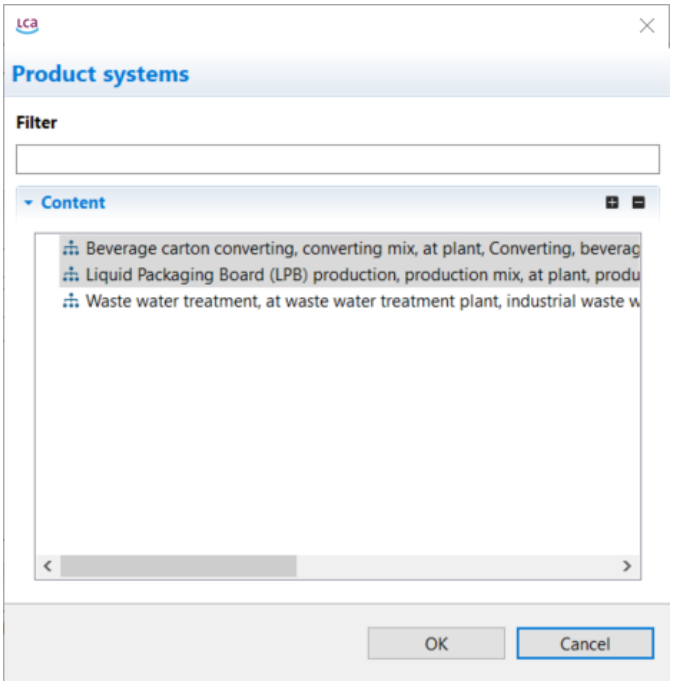


Figure 158: Adding product systems to a project

Each selected product system acts as a 'variant' for the calculation. It is possible to select the same product system multiple times and/or different product systems. For each variant, you can then select a new name, if desired, as well as a different allocation method and amount. For example, to compare the results of a product system calculation using three different allocation methods, select the same product system three times and then select a different allocation method for each. It is also possible to choose which product systems are to be displayed in the results.

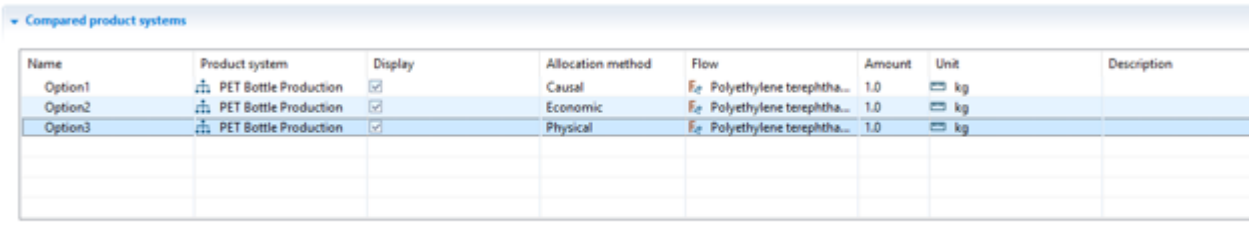


Figure 159: Project setup, Variants

In the "Parameters" section, it is possible to change parameter values for variants. For example, let's say we want to compare the impacts of a PET bottle production when PET Granulates are transported for 200, 500 or 350 km. You can select a product system for PET Bottle that contains a parameter (see Figure 160) for the transport distance ('D') three times in the "Compare product systems" section.

Rename each variant (see next figure). Then in the "Parameters" section add the parameter 'D' by clicking on the green "+" button on the right and selecting the correct parameter from those available. Then enter a new parameter value for each variant.

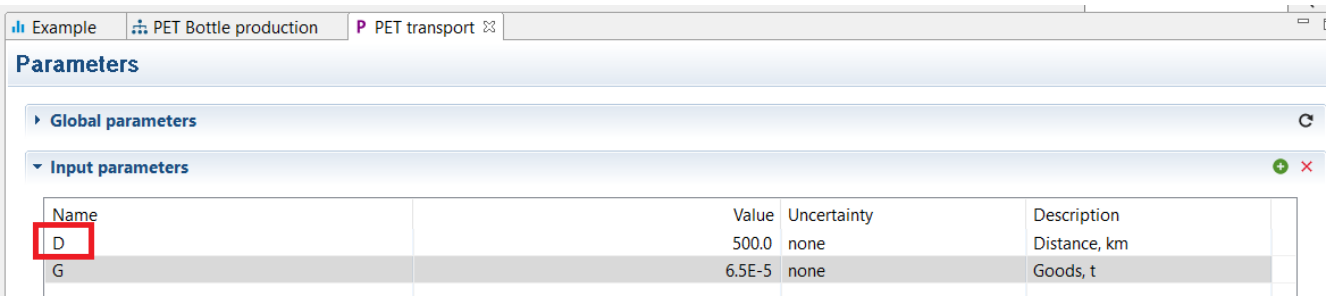


Figure 160: Parameter definition in processes

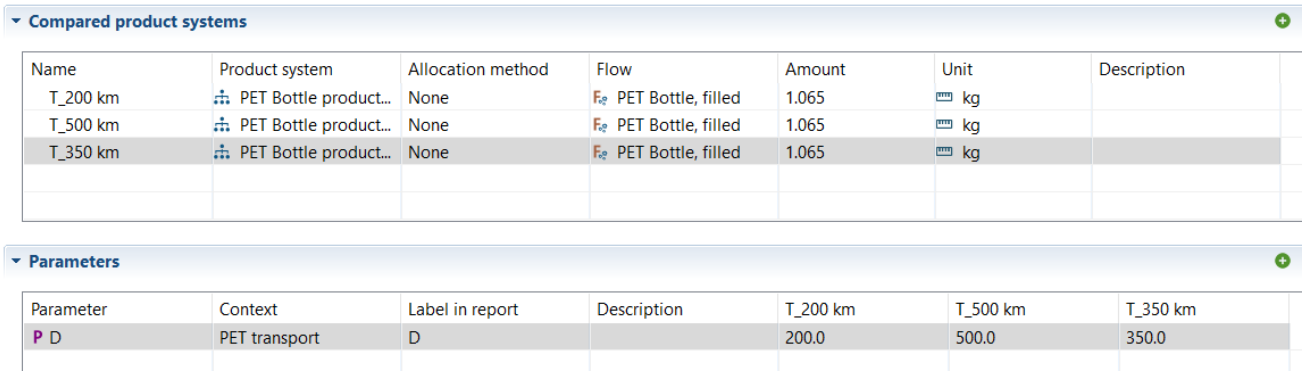


Figure 161: Project setup, Parameters

In the "Process contributions" section you can select processes whose impacts should be displayed separately in the report. Select processes using the green "+" icon on the right. You can then change the name shown in the report under "Label in report", if desired. Please note that the results shown are direct (single indicator) results, not including upstream processes. The process contributions will be displayed in the report in a diagram:

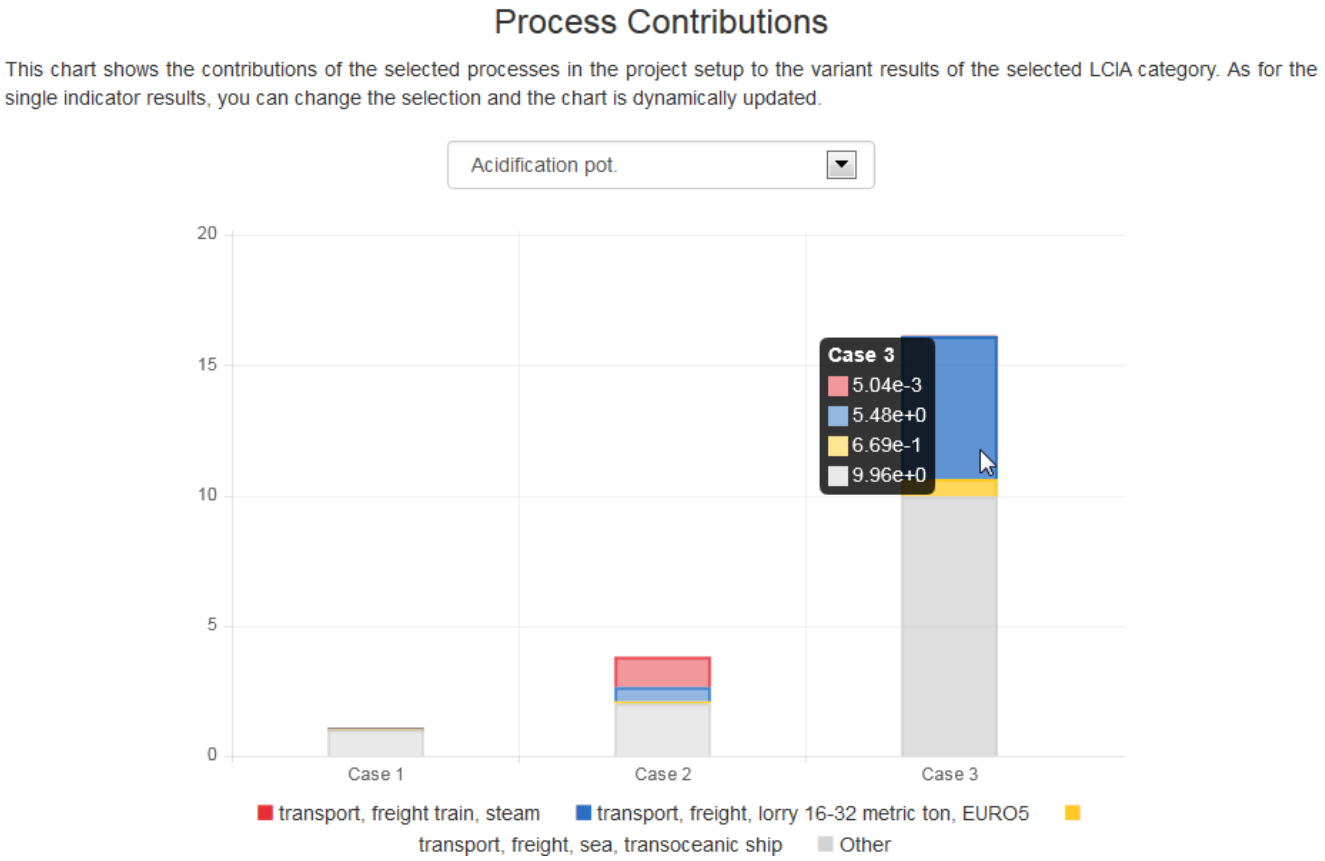


Figure 162: Project setup, Process contributions result example

10.2.2 Report sections

In this tab you can configure the report that will be created when you click on the “Report” icon in the “Project setup” tab. Here you can name the report and add/delete report sections. As per default, a report contains the sections ‘Introduction’, ‘Project Variants’, ‘Selected LCIA Categories’, ‘LCIA Results’, ‘Single Indicator Results’, ‘Process contributions’, and ‘Relative Results’. You can rename sections and edit the description text for each section. Delete sections by selecting the red “X” icon on the right. Change the order of sections using the up/down icons on the right. It is also possible to select a component for each section, for example, which type of chart or table should be displayed. Once you have all sections configured, make sure to save the project before creating the report. (Create a report by selecting the “Report” icon in the “Project setup” tab).

10.3 Exporting a project report

Once a report has been generated, a “Report Viewer” window will automatically open in the editor. Select the “Export report” icon in the top left-hand corner to export this report in html format.

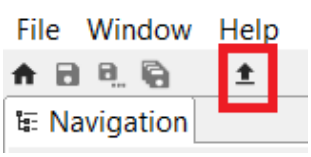


Figure 163: Exporting a report in html format

11 Advanced Topics

11.1 Expanding memory on a MacOS system

To expand memory on a mac, begin by opening the “Finder”, then “Applications” and find openLCA. Right-click on openLCA and select “Show package contents”:

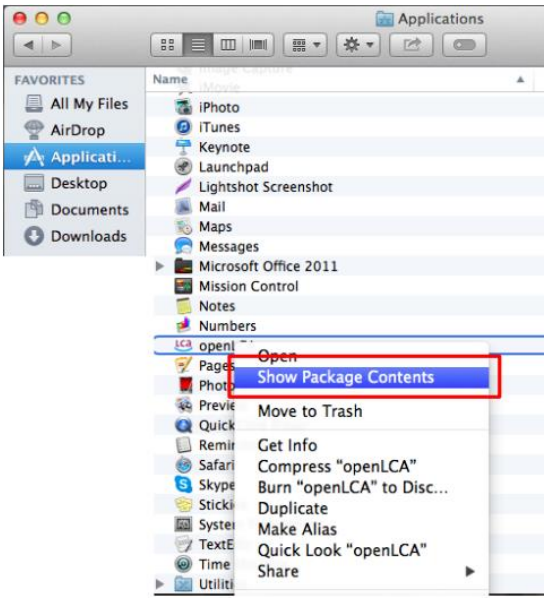


Figure 164: Expanding memory on a mac, step 1

Then open the file “Contents”, then the file “MacOS”. Double-click on the file “openLCA.ini” (if the text editor is not automatically opened, select the program “Text Edit” manually). In the editor, change the memory manually, for example to 4096M. Then close all windows and restart openLCA.

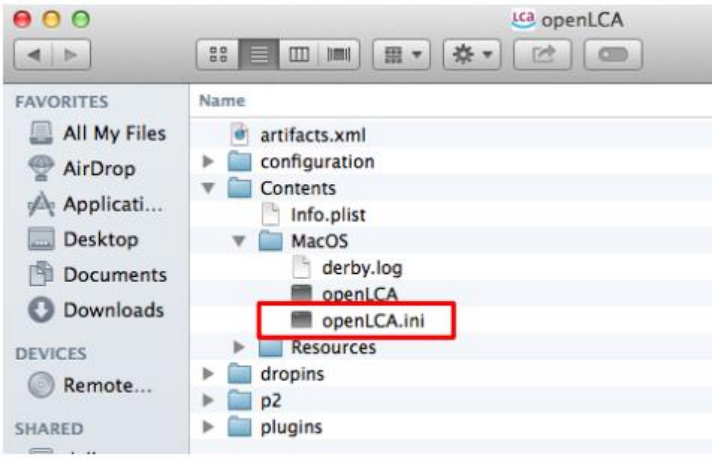


Figure 165: Expanding memory on a mac, step 2

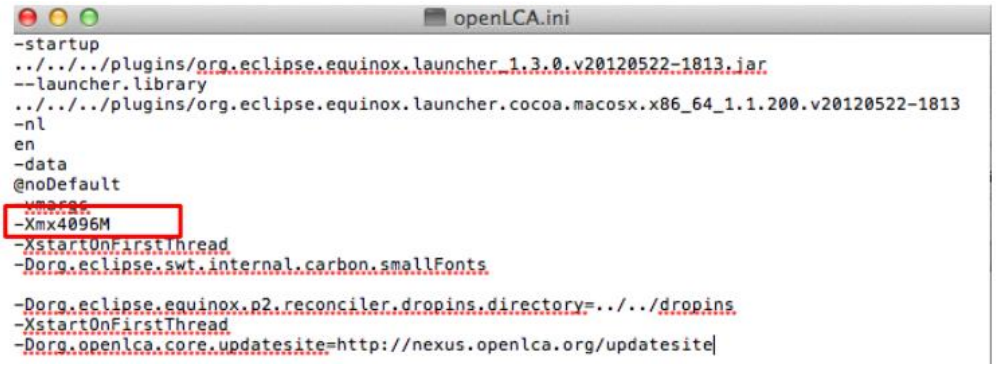


Figure 166: Expanding memory on a mac, step 3

11.2 Scripting in openLCA

Since version 1.7.0, openLCA supports the possibility to run Python and JavaScript programs directly in openLCA. With this feature you can automate calculations in openLCA, write your own data imports or exports, perform sensitivity analysis calculations by varying parameter values, and much more. You can find the scripting feature under the developer tools menu in openLCA:

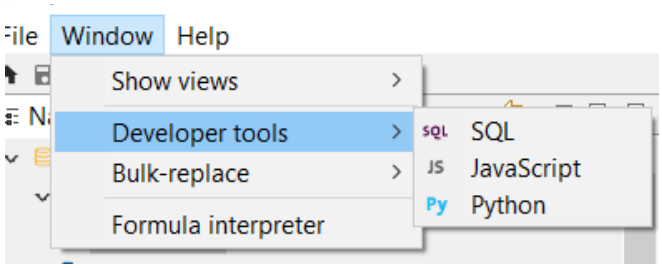


Figure 167: Accessing scripting

As shown in Figure 167, just click on the language you want to use, and an editor will open where you can write your scripts. To make debugging output visible, we directly connected the openLCA logger to the scripting environment. Thus, when you write scripts it is helpful to open the logging console in openLCA (File/Preferences/Logging):

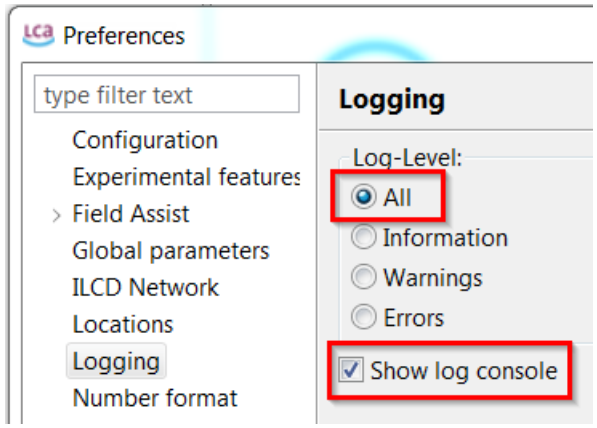


Figure 168: openLCA logger for scripting

After this we can execute our first script (note that it takes a bit to initialize the Python interpreter when

you execute a script the first time in your current openLCA session). Just write the following text in the editor and press the run button in the toolbar:

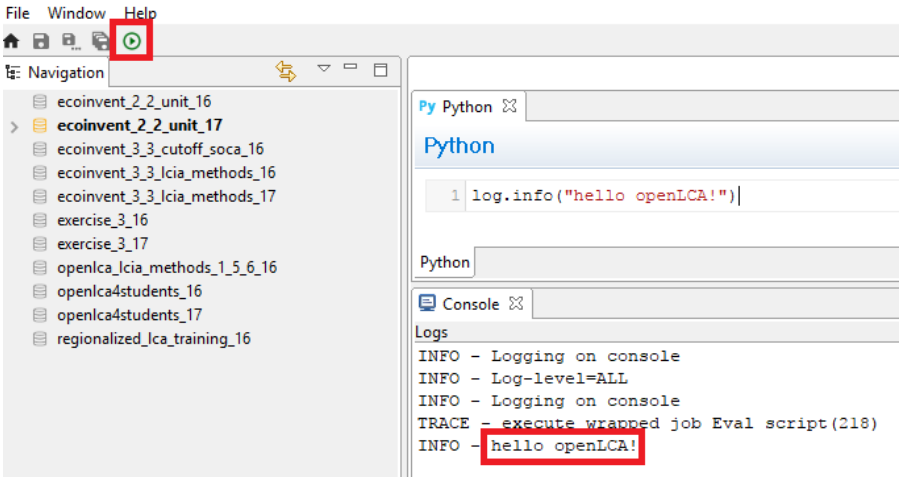


Figure 169: Running script example

We can also log an error via

```
log.error("Hello openLCA")
```

This will show an error popup in openLCA:

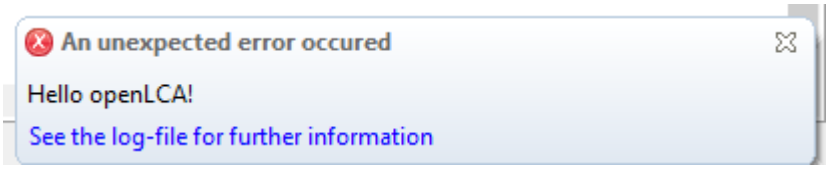


Figure 170: Error popup

As it is written in the popup, these logs are also written to the openLCA log-file which you can find in your user directory.

To execute a script, you click on the Run button in the toolbar of the Python editor (Figure 171):

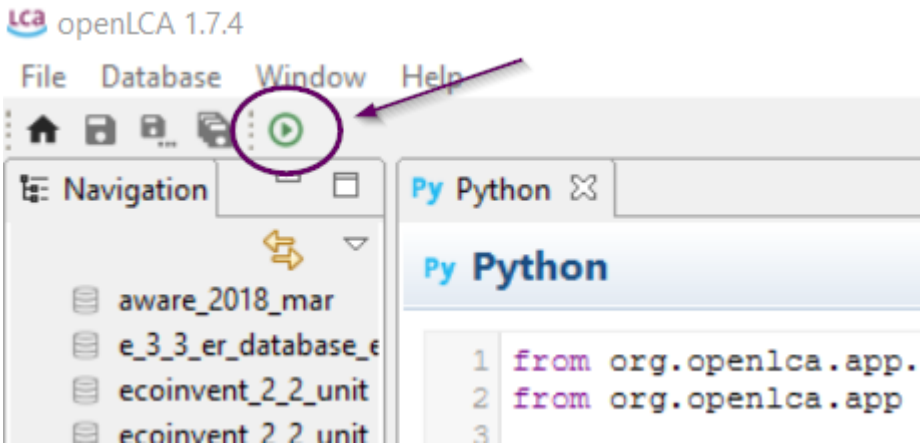


Figure 171: Run button in the toolbar

The script is executed in the same Java process as openLCA, thus with have access to all the things that you can do with openLCA via this scripting API (and to everything that you can do with the Java and Jython runtime). Here is a small example script that will show the information dialog below when executed in openLCA:

```
from org.openlca.app.util import UI, Dialog
from org.openlca.app import App

def say_hello():
    Dialog.showInfo(UI.shell(), 'Hello from Python (Jython)!')

if __name__ == '__main__':
    App.runInUI('say hello', say_hello)
```

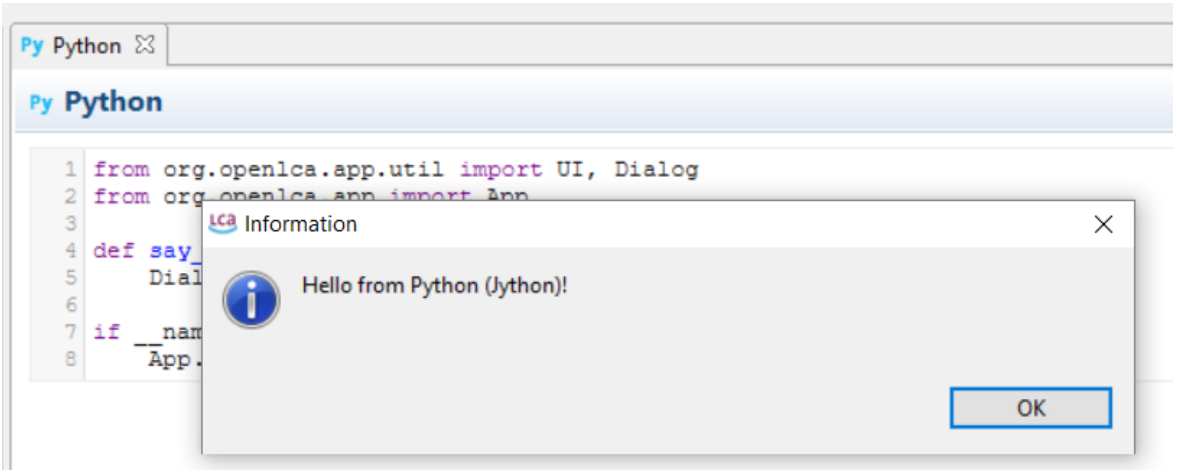


Figure 172: Output of code

As said above, Jython runs on the JVM. It implements a great part of the Python 2.7 standard library for the JVM, which can be found here. For example, the following script will work when you set the file path to a valid path on your system:

```
import csv
```

```
with open('path/to/file.csv', 'w') as stream:
    writer = csv.writer(stream)
    writer.writerow(["data you", "may want", "to export",])
```

The Jython standard library is extracted to the python folder of the openLCA workspace which is by default located in your user directory ~/openLCA-data-1.4/python. This is also the location in which you can put your own Jython 2.7 compatible modules. For example, when you create a file tutorial.py with the following function in this folder:

```
# ~/openLCA-data-1.4/python/tutorial.py
def the_answer():
  f = lambda s, x: s + x if x % 2 == 0 else s
  return reduce(f, range(0, 14))
```

You can then load it in the openLCA script editor:

```
import tutorial
import org.openlca.app.util.Info as Info

Info.showBox('The answer is %s!' % tutorial.the_answer())
```

An important thing to note is that Python modules that use C-extensions (like NumPy and friends) or parts of the standard library that are not implemented in Jython are not compatible with Jython. If you want to interact from standard CPython with openLCA (using Pandas, NumPy, etc.) you can use the openLCA-IPC Python API.

A tutorial of how to use python in openLCA can be found here.

11.2.1 The Python interpreter

openLCA integrates Jython 2.7 (http://www.jython.org/) as Python interpreter. Python is a fully featured programming language that supports imperative, object oriented, and functional programming paradigms. Additionally, Python is very easy to learn so that you can quickly get started. Jython also comes with a powerful part of the Python standard library. For example, you can use the csv module (http://www.jython.org/docs/library/csv.html) to read and write csv files directly in openLCA:

Python

```python
import csv

f = open('C:/Users/Besitzer/Desktop/test-out.csv', 'wb')
writer = csv.writer(f)

for i in range(0, 100):
   writer.writerow([i, 1.0/(i+1)])

f.close()
```

Figure 173: Read and write csv files in openLCA

11.2.2 The JavaScript interpreter

The JavaScript interpreter that openLCA uses directly comes with the Java 8 runtime which is included in openLCA (http://www.oracle.com/technetwork/articles/java/jf14-nashorn-2126515.html). Like Python, JavaScript is a programming language that supports imperative, object oriented, and functional programming paradigms. Due to the modern web, JavaScript is currently one of the most popular programming languages. Note, that the JavaScript runtime in openLCA fully supports the JavaScript language but that the runtime platform is not the browser but openLCA (i.e. there is no window-object or DOM as in a web-browser). However, you can load other JavaScript files; e.g. you could use math.js (http://mathjs.org) in openLCA via the load-function (the same works with execfile for Python):

JavaScript

```javascript
// load math.js module
load('C:/Users/Besitzer/Desktop/math.js');

log.info(math.eval('5.08 cm to inch'));
```

JavaScript

Console

Logs

```
INFO - 2.0000000000000004 inch
```

Figure 174: Loading JavaScript files

11.2.3 The olca-object and the inspection function

To provide an entry point to the openLCA API[2] you have access to an object with the name *olca* which provides a set of useful methods. For example, you could log the names of all processes in a database with the following Python script[3]:

[2] API stands for Application Programming Interface

[3] Note that you need to open a database when you want to access data via the script API.

```
for descriptor in olca.getProcessDescriptors():
log.info(descriptor.name)
```

One of the most useful methods of the *olca*-object is the *inspect* function which takes an object as an argument and writes the object´s protocol to the logging console. For example, if we want to know which methods we can call on a process descriptor object we could write:

```
olca.inspect(olca.getProcessDescriptors()[0])
```

This will take the first process descriptor object from the database (or give an error if we do not have a process in the currently opened database) and print the protocol of the object to the logging console:

```
protocol:
  compareTo(BaseDescriptor) : int
  compareTo(Object) : int
  equals(Object) : boolean
  getCategory() : Long
  getClass() : Class
  getDescription() : String
  getId() : long
  getLocation() : Long
  getModelType() : ModelType
  getName() : String
  getProcessType() : ProcessType
  getQuantitativeReference() : Long
  getRefId() : String
  hashCode() : int
  isInfrastructureProcess() : boolean
  notify() : void
  notifyAll() : void
  setCategory(Long) : void
  setDescription(String) : void
  setId(long) : void
  setInfrastructureProcess(boolean) : void
  setLocation(Long) : void
  setName(String) : void
  setProcessType(ProcessType) : void
  setQuantitativeReference(Long) : void
  setRefId(String) : void
  setType(ModelType) : void
  toString() : String
  wait() : void
  wait(long) : void
  wait(long, int) : void
```

In this protocol each line describes a method you can call in the following form:

```
[method name] ( [type of argument 1] [type of argument 2] ... ) : [return type]
```

For getter methods with no arguments we can use a simplified form in the Python interpreter: instead of

```
descriptor.getName()
```

we can also write

```
descriptor.name
```

To see the methods of the olca-object we can also call the inspect method on the olca-object itself:

```
olca.inspect(olca)
```

In the protocol that is now written to the console we can see for example the following method:

```
getProcess(String) : Process
```

This means that we can get a process for a string (which is the name of the process). The following script will load the process with the name "compost plant, open" from the database and write the process name and protocol to the console:

```
process = olca.getProcess("compost plant, open")
log.info(process.name)
olca.inspect(process)
```

11.2.4 Modifying database content

The *olca*-object also contains methods like *insertProcess* or *updateProcess* which – like the names say – will insert a new process object or update an existing process in the currently opened database. The following example loads the process with the name p1 from the databases, clears the current parameter list of this process, adds 10 new parameters to this process, and finally updates this process in the database:

```
# get the process p1 from the database
process = olca.getProcess("p1")

# remove the current process parameters
process.parameters.clear()

for i in range(1, 11):
    # create a new parameter object
    param = Parameter()
    param.name = "p%s" % i
    # set it as an input parameter
    param.inputParameter = True
    param.value = i/42.0 * 5000
    # add the parameter to the process
    process.parameters.add(param)

# update the process in the database
olca.updateProcess(process)
```

11.2.5 Running Calculations

In the following example the product system "dung slab" is calculated with the LCIA method "CML 2001" and the results are written to the console (if you are not sure which methods you can call on a result type, just call the *inspect* method of the *olca*-object with the result type as parameter):

```
result = olca.calculate(
```

```
    olca.getSystem("dung slab"),
    olca.getMethod("CML 2001") )

for i in result.totalImpactResults:
  log.info("LCIA category = {}, result value = {}, unit = {}",
           i.impactCategory.name, i.value, i.impactCategory.referenceUnit)
```

By using the Python standard library, we can easily export our results to a CSV file:

```
# import the Python CSV module
import csv

# calculate the product system
result = olca.calculate(
    olca.getSystem("dung slab"),
    olca.getMethod("CML 2001") )

# open the export file
f = open('C:/Users/Besitzer/Desktop/results_out.csv', 'wb')
writer = csv.writer(f)

# write the results to the file
for i in result.totalImpactResults:
  row = [i.impactCategory.name, i.value, i.impactCategory.referenceUnit]
  writer.writerow(row)

# close the file
f.close()
```

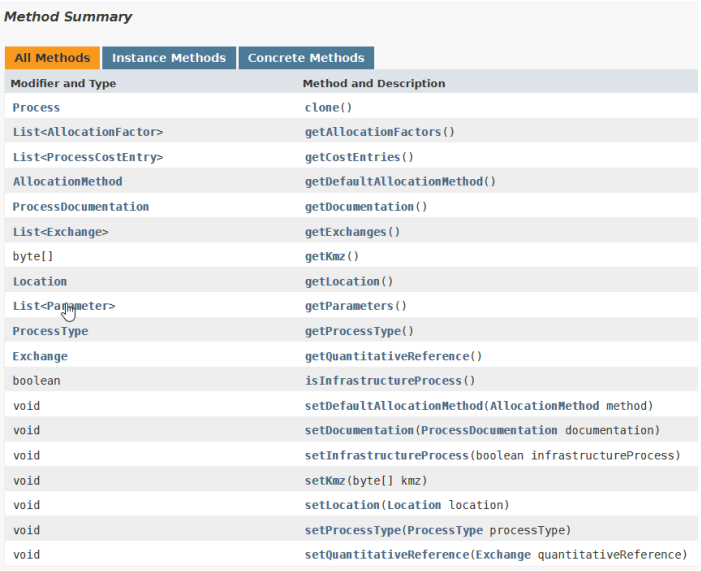
You could now combine the calculation with a parameter modification as described above to make advanced sensitivity analyses with openLCA. Also, the analysis function and Monte Carlo Simulation are available via the olca-object:

```
# run a Monte Carlo Simulation with 10 iterations
system = olca.getSystem("dung slab")
result = olca.simulate(system, 10)

# for each flow write the result of each iteration
for flow in result.flowDescriptors:
  i = 1
  for r in result.getFlowResults(flow):
    log.info("flow: {}, iteration: {}, result: {}", flow.name, i, r)
    i += 1
```

11.2.6 Using more functions from the API

With the scripting interface in openLCA you have full access to all functions in openLCA. The *olca*-object just provides some entry points to the API. The full API documentation of the core openLCA modules is available on our GitHub repository: http://greendelta.github.io/olca-modules/. The picture below shows some method declarations in the API documentation of the Process class. These methods are the same as the *inspect* method will print when you call it with a process object as parameter.



Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method and Description
Process	clone()
List<AllocationFactor>	getAllocationFactors()
List<ProcessCostEntry>	getCostEntries()
AllocationMethod	getDefaultAllocationMethod()
ProcessDocumentation	getDocumentation()
List<Exchange>	getExchanges()
byte[]	getKmz()
Location	getLocation()
List<Parameter>	getParameters()
ProcessType	getProcessType()
Exchange	getQuantitativeReference()
boolean	isInfrastructureProcess()
void	setDefaultAllocationMethod(AllocationMethod method)
void	setDocumentation(ProcessDocumentation documentation)
void	setInfrastructureProcess(boolean infrastructureProcess)
void	setKmz(byte[] kmz)
void	setLocation(Location location)
void	setProcessType(ProcessType processType)
void	setQuantitativeReference(Exchange quantitativeReference)

Figure 175: Method declarations in API documentation

To use a class in your script you need to import it. There is for example a class *CategoryPath* available in the package *org.openlca.io* which translates a category hierarchy in openLCA into a string path. The following script shows how you could use this class in a Python script:

```
# import the class
import org.openlca.io.CategoryPath as path

# load a process
process = olca.getProcess("compost plant, open")

# print the full category path of the process
log.info(path.getFull(process.category))

# print the short category path of the process
log.info(path.getShort(process.category))
```

And here is the same in JavaScript:

```
// import the class
var path = Java.type('org.openlca.io.CategoryPath');

// load a process
var process = olca.getProcess("compost plant, open");

// print the full category path of the process
log.info(path.getFull(process.category));

// print the short category path of the process
log.info(path.getShort(process.category));
```

For the core model classes, you do not need to add import declarations. So, you can directly write the

following to create a new process object in Python and inspect it (see also the parameter example above):

```
process = Process()
olca.inspect(process)
```

And in JavaScript:

```
var process = new Process();
olca.inspect(process);
```

Finally, the following picture shows the classes and the dependencies between these classes of the core model:

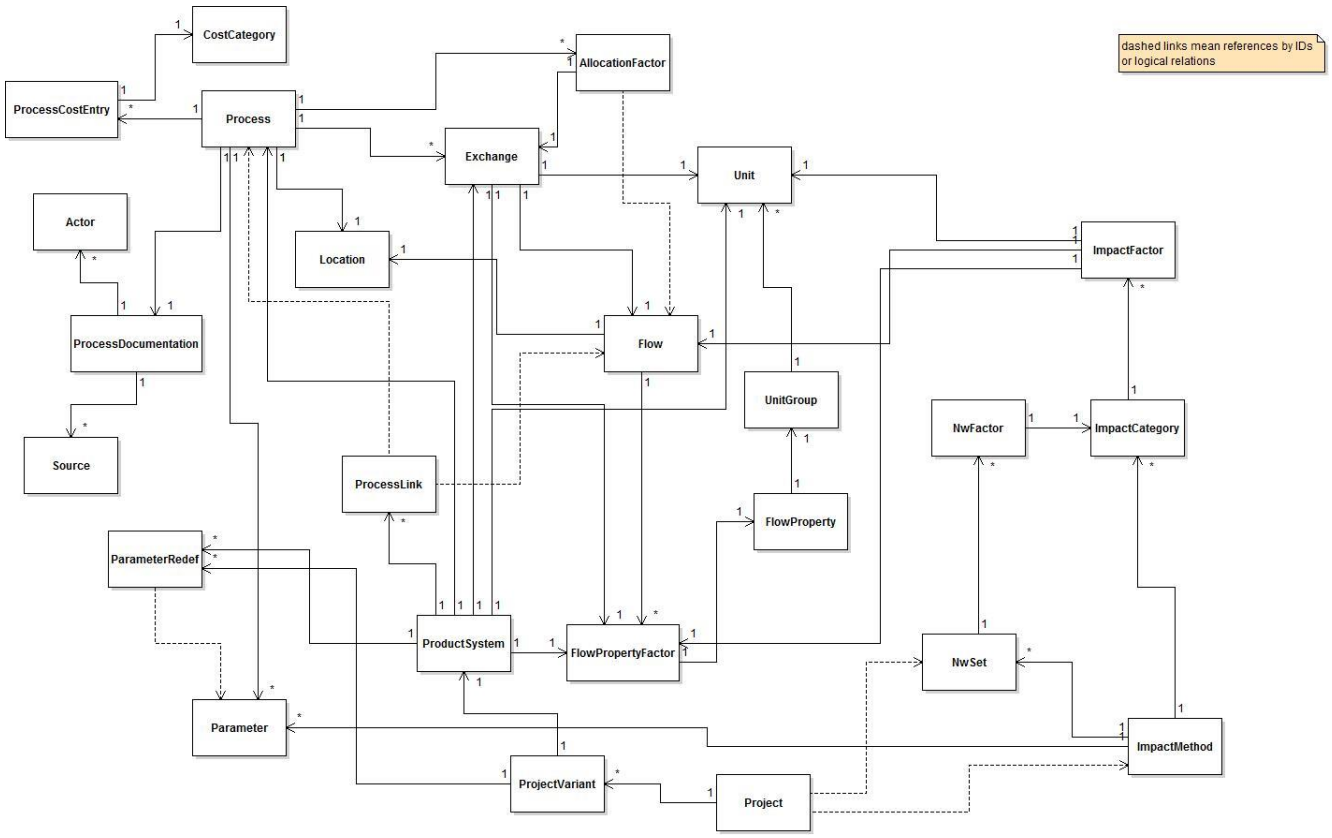


Figure 176: Classes and dependencies of core model

12 Quality assurance and performance

12.1 Quality assurance

In the development of openLCA, extensive tests were performed to verify and validate the software algorithms and overall usage, also by external testers, e.g. by our Italian partner, be-LCA (www.be-LCA.com). Tests have been performed for example to check, among others, openLCA results versus SimaPro calculation results and openLCA results versus ecoinvent system process results. Further,

'constructed' use cases have been used to reflect specific modelling aspects of LCA (allocation, system expansion, uncertainty calculation, parameter usage, etc.) which have been calculated in other software systems, including MATLAB / Scilab and excel. Here is not the space to fully explain all the tests, but we would like to illustrate some of our more relevant results. More information is available on request (see contact information).

Figure 177 shows a comparison of inventory results for SimaPro and openLCA, for the process electricity production mix high voltage, country mix, BE, for ecoinvent 3.0.1, default allocation model. Only those flows are considered where flow names and compartments are identical between SimaPro and openLCA, which is the case for more than 700 flows for the said system. The models were calculated in both software systems, and then exported from SimaPro via text export (which is limited to exponent and two digits) and from openLCA via excel export.
As the figure shows, results are very similar. The most extreme ratios are still very close to 1 meaning that both software systems calculate almost fully identical results:

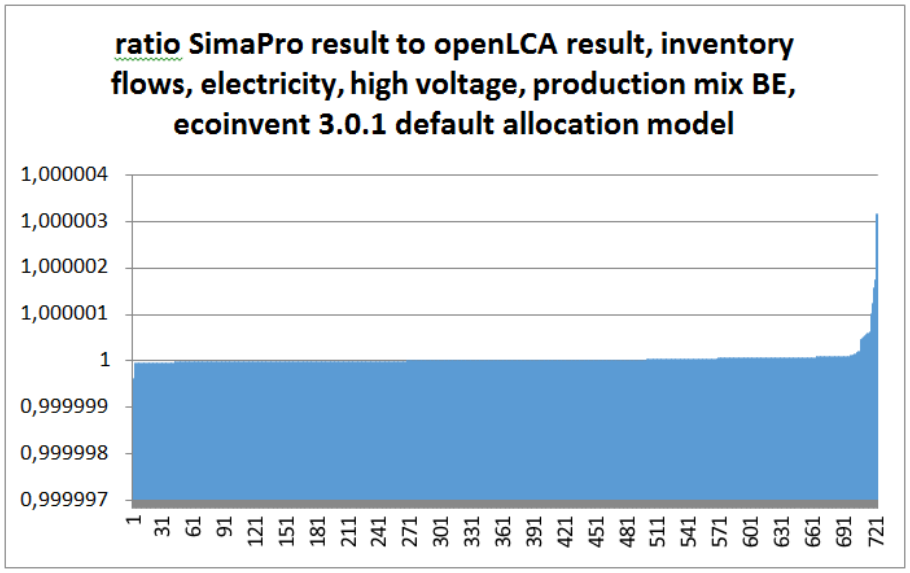


Figure 177: Comparison of inventory results for SimaPro and openLCA

Top 5 minimum ratios are shown in table 1; table 2 shows the top 5 maximum ratios:

Table 1: Ratio of results from SimaPro and openLCA in a calculation comparison, 5 lowest

Input	Category	Sub-category	Flow_name	ratio_SP_to_openLCA
ja	resource	land	Transformation, to unknown	0.999999629
ja	resource	land	Transformation, from mineral extraction site	0.999999816
nein	soil	agricultural	Carbaryl	0.999999946
nein	air	high population den	Ethane	0.99999995
nein	soil	agricultural	Nicosulfuron	0.999999953

Table 2: Ratio of results from SimaPro and openLCA in a calculation comparison, 5 highest

Input	Category	Sub-category	Flow_name	ratio_SP_to_openLCA
ja	resource	in ground	Gallium	1.000001574
ja	resource	land	Transformation, to forest	1.000001739
ja	resource	land	Transformation, to shrub land, sclerophyllous	1.000001744
ja	resource	land	Transformation, from dump site, inert materia	1.000003167
ja	resource	land	Transformation, to dump site, inert material la	1.000003167

This is result of a life cycle calculation with about 7500 unit-processes included.

12.2 Performance

Improving the performance also for large life cycle systems is one of the core tasks in the development of openLCA. Table 3 presents some results (Windows 64-bit version, Windows 10, identical, modern notebook computer, calculation using CED method), also in comparison to SimaPro 9.0.0.30.

Figure 178: PC features

Table 3: openLCA and SimaPro performance comparison in network/analysis calculation

	Time required [seconds]		Ratio: required time
	openLCA 1.9	SimaPro 9	SimaPro/openLCA
Ecoinvent 3.5 barley grain, feed production | Cutoff, U (CA-QC)			
Creating a product system *	11	*	*
Analysing/ calculating a product system*	30	*	*
Total**	41	17	0.41
Ecoinvent 3.5 electricity high voltage production mix BE, cut-off,U			
Creating a product system *	4.8	*	*
Analysing/ calculating a product system*	26	*	*
Total**	30.8	29	0.94

* not available in SimaPro as separate action, always performed when calculating a network in SimaPro

** for openLCA, addition of time for creating and calculating the product system; opening an already created product system in openLCA takes about one second.

SimaPro is used as a benchmark here since it is the only other broadly used LCA software system (at least to our knowledge) that can calculate large unit-process based product systems from ecoinvent 3 so far. For the performance tests, the identical computer was used, no other user was interfering with the SimaPro multi-user Developer version; in openLCA, the 'quick analysis' calculation option was selected which produces results like the "analysis calculation" in SimaPro which in turn was used for SimaPro.
It is also possible to compare the "analysis calculation" option with the "network calculation" in SimaPro as they produce similar results.

The table shows that openLCA is between 41% to 94% as fast as SimaPro when calculating a large ecoinvent unit process life cycle. This is also shown in the figure below. Both software systems become faster when repeating identical tasks, e.g. when the same product system is deleted and created again, due to caching procedures; in SimaPro, especially a repeated calculation is finished much quicker. In openLCA, a once created product system can be stored; reopening it takes only about one second and saves the time for recreating it. openLCA offers a further 'quick results' calculation where only main contributors and inventory and impact assessment tables are calculated; this quick calculation is about twice as fast as the analysis calculation.

13 Contact

openLCA is developed and managed by GreenDelta in Berlin. If you have any feedback, comments, questions, please let us know.

GreenDelta GmbH
Müllerstrasse 135
13349 Berlin, Germany
Tel. +49 30 48 496 - 030
Fax +49 30 48 496 - 991
gd@greendelta.com
www.greendelta.com

Follow openLCA:

14 Acknowledgements

Finally – we would like to thank our supporters and contributors. The following institutions provided financial support (incomplete overview):