Fuel Life Cycle Assessment Model User Manual

June 2024

Environment and Climate Change Canada

Environnement et Changement climatique Canada

Canada

Table of Contents

List of Figures

List of Tables

Definitions

Allocation: partition of input or output flows of a process between the product system under study and one or more other product systems (ISO 14040).

Carbon dioxide equivalent (CO₂e): quantity of carbon dioxide that would be required to produce an equivalent warming effect over a given time period.

Carbon intensity: in relation to a pool of a given type of fuel, this means the quantity of CO_2e in grams that is released during the activities conducted over the fuel's life cycle — including all emissions associated with the extraction or the cultivation of feedstock used to produce the fuel, with the processing, refining or upgrading of that feedstock to produce the fuel, with the transportation or distribution of that feedstock, of intermediary products or of the fuel and with the combustion of the fuel — per megajoule of energy produced during that combustion.

Ecosphere: consists of the entire natural environment. Examples include air, water, and natural resources.

Elementary flow: flow that is exchanged with the environment, e.g. GHGs.

Feedstock: resource that is extracted, cultivated, collected, harvested, and/or processed and delivered at the gate of the production facility from which fuel is produced.

Flow: material or energy that enters or leaves a process.

Fuel pathway: a collection of unit processes, modeling parameters, and background data in the *Fuel LCA Model* that allows the determination of the CI of a fuel from a particular feedstock type.

Functional unit: quantified performance of a product system for use as a reference unit (ISO 14040).

Intermediate flow: flow that is exchanged within the technosphere i.e. human control. In the context of the *Fuel LCA Model*, any flow that is not an elementary flow.

Life cycle assessment (LCA): compilation and evaluation of the inputs, outputs, and the potential environmental impacts of a product system throughout its life cycle (ISO 14040).

Life cycle impact assessment (LCIA): phase of LCA aimed at understanding and evaluating the magnitude and significance of the potential environmental impacts for a product system throughout the life cycle of the product. (ISO 14040).

Life cycle inventory (LCI): phase of LCA involving the compilation and quantification of inputs and outputs for a product through its life cycle (ISO 14040).

Life cycle stage: collection of unit processes connected by a network of flows that models a main stage of the life cycle of a fuel. In the *Fuel LCA Model*, there are five life cycle stages: feedstock production, feedstock transportation, fuel production, fuel distribution, and fuel combustion.

Life cycle: consecutive and interlinked stages of a product system, from feedstock acquisition to combustion of the produced low carbon-intensity fuel.

Low-carbon-intensity fuels (LCIF): fuels, other than the fossil fuels, with a lower carbon intensity than fossil fuels. This definition includes hydrogen.

Product system: collection of unit processes that models the life cycle of a product (ISO 14040).

System process: process that contains the LCI of a group of unit processes.

Technosphere: consists of all anthropogenic developments. Once materials from the ecosphere are extracted and in human-control, they are part of the technosphere.

Unit process: smallest element for which input and output data are quantified (ISO 14040).

Waste flow: flow that is used in modelling waste treatment processes (openLCA).

Acronyms

CI	Carbon intensity
CFR	*Clean Fuel Regulations*
ECCC	Environment and Climate Change Canada
GWP	Global warming potential
GHG	Greenhouse gas
HHV	Higher Heating Value
IPCC	Intergovernmental Panel on Climate Change
AR5	IPCC's Fifth Assessment Report
AR6	IPCC's Sixth Assessment Report
LCA	Life cycle assessment
LCIA	Life cycle impact assessment
LCI	Life cycle inventory
LCIF	Low carbon-intensity fuel

Chapter 1: Fuel LCA Model introduction

The Government of Canada's *Fuel Life Cycle Assessment (LCA) Model* (the Model) is a tool that allows users to calculate the life cycle carbon intensity (CI) of fuels and energy sources produced and used in Canada. The Model is publicly available and is intended to inform and reduce the carbon intensity of Canadian fuels. Users of the Model could include industry, academia, LCA practitioners, governmental organizations, non-governmental organizations, and other organisations with interest in the energy sector. The Model may also be used in the context of specific programs.

The Model consists of the following three components:

1) **the Fuel LCA Model Database**: Contains a library of CI datasets and fuel pathways developed to model a CI specific to a fuel or an energy source.
2) **the Fuel LCA Model Methodology**: Describes the methodology, data sources and assumptions that were used in the development of the Model. The document provides the rationale supporting the methodological approach.
3) **the Fuel LCA Model User Manual** (this document): Provides information on general definitions and concepts related to LCA as it relates to the Model. Also provides technical guidance on how to perform basic operations in the openLCA software that are required for CI calculations.

Chapter 2: Purpose of the *Fuel LCA Model* User Manual

The purpose of this document is to provide users of the Model with technical guidance on how to perform basic operations in the openLCA software that are required for CI calculations. It also provides information on general definitions and concepts related to LCA from the perspective of the Model in **Chapter 6:.** Users unfamiliar with LCA are encouraged to read **Chapter 6:** first.

This document describes how to use the Model and perform operations in the openLCA software for general purposes. Other programs, such as the *Clean Fuel Regulations* (CFR)*,* may refer to specific sections of this document.

The overall process to calculate a CI using the Model is shown in **Figure 1**. Steps 1 and 3, which involve accessing the Model Database and performing LCA modelling of processed data, are included in the scope of this user manual. Steps 2 and 4, which include data collection and preparation and CI reporting, are not discussed in this document but may be outlined in documentation for programs that require the use of the Model. Users of the Model are encouraged to have this document open alongside the openLCA software when modelling.

The instructions and screenshots in this document were developed using openLCA version 2.1.
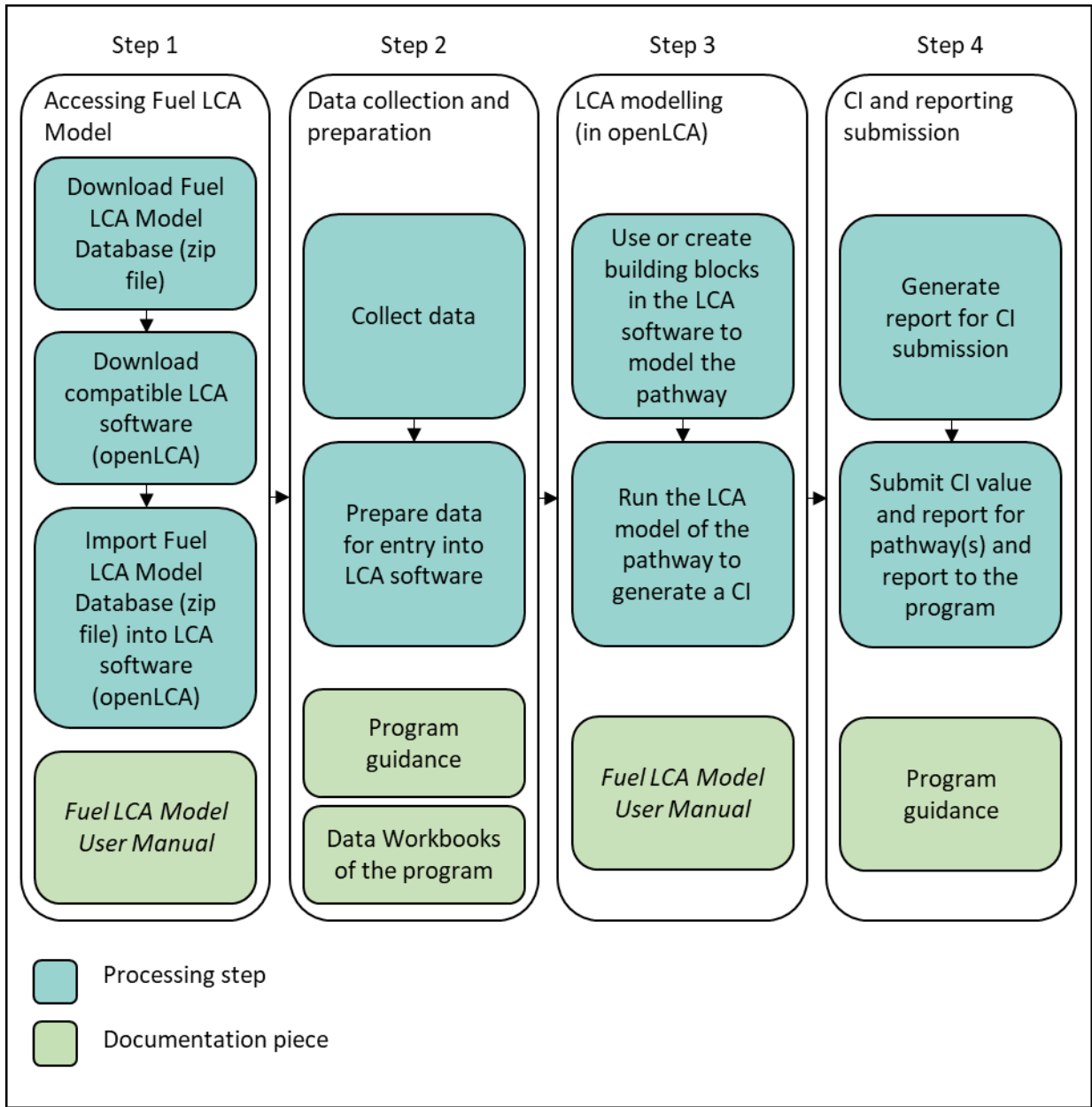
Figure 1: Steps involved in calculating a CI using the Model

Chapter 3: Setting up the *Fuel LCA Model*

To use the Model, the user must install a stable version 2.0 or higher of the <u>openLCA software</u>, a free and open-source software, download the Model Database, which is a zip file, and then import the database in openLCA.

3.1 Importing the *Fuel LCA Model* Database

The steps below outline the process for importing the Model Database into openLCA. There are also screenshots of steps 4 to 10 of the import procedure in openLCA in **Figure 2**.

1. Download the Model Database (zip file) from the Environment Climate Change Canada (ECCC) website.
2. Install the openLCA software on your computer (connect with your local IT/network provider if issues arise).
3. Open openLCA.
4. In the toolbar at the top-left, click "Database".
5. Select "New database".
6. Enter the database name.
    a) For the database content, select "Empty database" (crucial step).
    b) Click "Finish".
7. Right-click your new, empty database, and click "Import" and then "File".
8. In the dialog box that opens, navigate to select the Model Database (zip file).
    a) Select the Model Database (zip file).
    b) Click "Open".
9. Select "Overwrite all existing data sets" and then click "Finish".
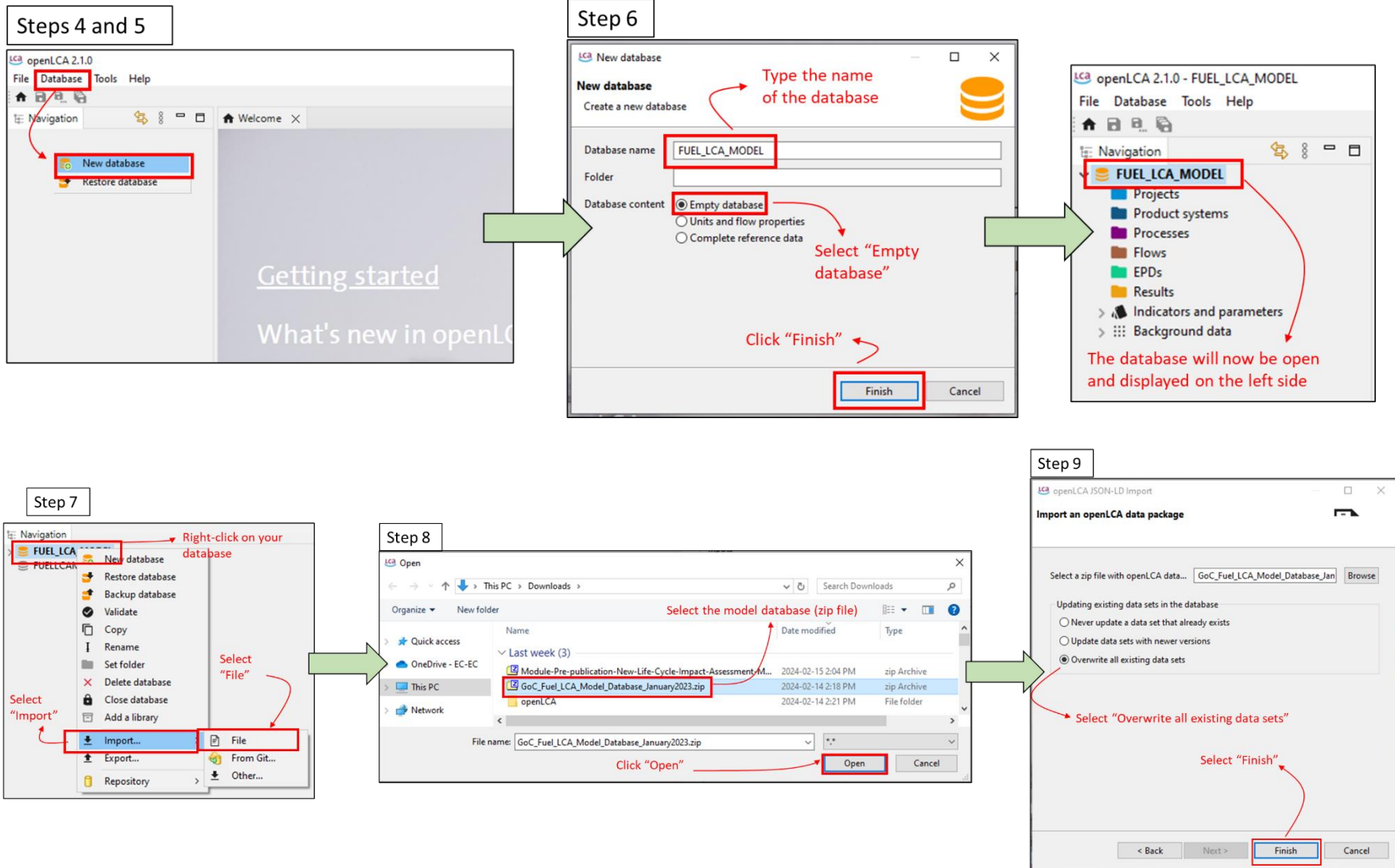
Figure 2: Screenshots of the steps to import a database into openLCA

3.2 Updating the openLCA software and the Model Database

3.2.1 Transition of databases from openLCA 1.11 to openLCA 2.0 and higher

Previous versions of the Model Database are compatible with openLCA versions 2.0 and higher. To convert databases from openLCA versions 1.11 and lower to version 2.0 and higher, follow the steps in one of the two methods below. Note that this process is irreversible. Once it has been converted it cannot be re-opened in openLCA 1.11. Instructions are included in each method for backing up a copy of the openLCA 1.11 format of the database.

Method 1

1. Open openLCA 1.11.
2. Right-click the database you wish to copy.
3. Click "Copy".
4. Enter the name for the copy and click "OK".
5. Close openLCA 1.11.
6. Open openLCA 2.0.
7. Double-click the database that was made in openLCA 1.11.
8. A dialog box will open that says "The selected database needs an update. Do you want to run it?".
9. Uncheck the box that says, "Create a backup of the current database first".
10. Click "OK" to convert the database.

Method 2

1. Open openLCA 2.0.
2. Double-click the database that was made in openLCA 1.11.
3. A dialog box will open that says "The selected database needs an update. Do you want to run it?".
4. Make sure that the box that says, "Create a backup of the current database first" is checked.
5. Click "OK".
6. A dialog box will open for the creation of the backup .zolca file.
7. Enter a new name for the backup of the database in the openLCA 1.11 format.
8. Click "Save".

With method 2, the software will create the backup of the database, convert the database to the openLCA 2.0 format, and then open the openLCA 2.0 database. It should be noted that the backup of the database in the openLCA 1.11 format is not saved in openLCA. To reopen the database in openLCA 1.11, follow the steps below.

9. Open openLCA 1.11.
10. Click "Database" at the taskbar at the top left of the screen.
11. Click "Restore database".
12. Click on the backed up database from steps 1-6 and click "Open".
13. The backed up database is now ready for use in openLCA 1.11.

3.2.2 Importing previous modelling into the current version of the Model Database

If you have performed modelling using the Fuel Pathways in an older version of the Model Database and wish to link them to a newer version of the Model Database, follow the steps below. This procedure assumes that the structure of the Fuel Pathways did not change between different versions of the Model Database.

1. First, make sure you are working in version 2.0 or higher of openLCA. See the previous subsection for further information.
2. Open your old database.
3. Follow the steps described in **Chapter 5.2.1** to export part of your old database. In step 5 of that procedure, select ONLY the Fuel Pathway unit processes that you modified. Do NOT check any processes from the "Configurable processes" or "Data Library" folders.
4. Follow the steps described in **Chapter 3.1** to set up the new version of the Model Database.
5. Follow the steps described in **Chapter 3.1**, starting with step 7, and choosing the zip file you exported from step 3 as the file to import. For step 9 of the procedure, choose "update data sets with newer versions".

Note: you must remodel the inputs/outputs of any configurable processes you used because the configurable processes are updated with each publication of the Model. However, you will not need to update their usage as an input within fuel pathway processes. Additional information related to the changes between Model databases is available on the [Fuel LCA Model Website](#).

Chapter 4: Organization of the *Fuel LCA Model* Database

4.1 Layout of openLCA

The layout of openLCA is shown in **Figure 3**. The navigation pane is on the left side of the software and allows access to the different components of the database. Once an item such as a unit process is opened, it can be viewed and used on the right side in the workspace. When multiple items are opened, they are accessible in tabs at the top of the workspace.
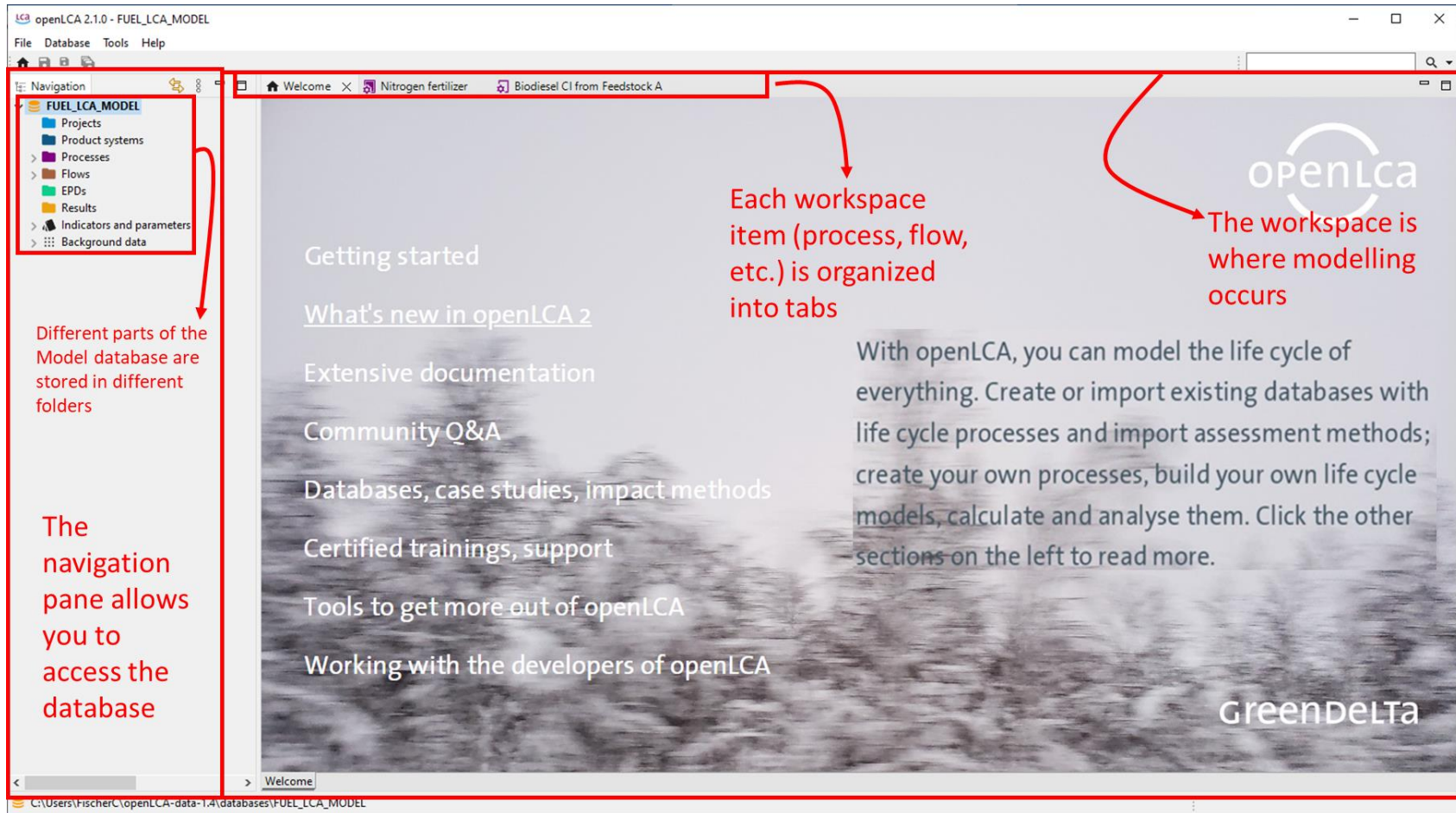
Figure 3: Layout of the openLCA software

4.2 Processes

There are two types of processes in the Model Database: unit processes and system processes.

The Model Database stores the processes into two main folders: "Data Library" and "Fuel Pathways". The former contains system processes that model the life cycle emission factors for various activities in the low-carbon-intensity fuel (LCIF) life cycle, while the latter contains unit processes that can be used to model a CI of the LCIF. See **Figure 4** below for a brief description of each folder.

Unit process: smallest element for which input and output data are quantified (ISO 14040).

System process: Process that models the emissions for a group of unit processes.

FUEL_LCA_MODEL
- Projects
- Product systems
- Processes
  - Data Library
  - Fuel Pathways

Data Library
- System processes only
- Developed by ECCC, cannot be modified by user
- Building blocks used as inputs in unit processes

Fuel Pathways
- Unit processes only
- Structured by ECCC, filled in by user
- Includes unit process pathway that users can complete using the system processes from the Data Library
- Includes configurable unit processes where users can regionalize some system processes

Figure 4: Left: The main process folders as viewed in openLCA. Right: Description of the main process folders of the Model Database.

Each process contains modelling documentation that is called metadata. Users can add information in the fuel pathway unit processes that they model, or in new unit processes that they create. The metadata can include:

- Process description (functional unit, scope, system boundaries, allocation information)
- Time (related to the data collection period)
- Geographical scope
- Technology being modelled
- Administrative information (intended application, access and use restrictions, data set owner)
- Modelling information (data completeness, data selection, data treatment)
- References

More information on the concepts of unit processes and system processes is available in **Chapter 6.2.1** or **Chapter 6.2.4**, respectively.

More information on the layout and contents of the metadata is available in **Chapter 4.5**.

4.2.1 Data Library

The Data Library acts as the backbone of the Model Database. The system processes within the Data Library are used to populate and model the unit processes found in the Fuel Pathways folder of the database. The Data Library was developed by ECCC using a combination of primary and literature data. In addition to the metadata documentation in each system process, the development, modelling assumptions and data sources of the Data Library are available in the *Fuel LCA Model Methodology*. The organization and contents of the Data Library is shown in **Figure 5**.
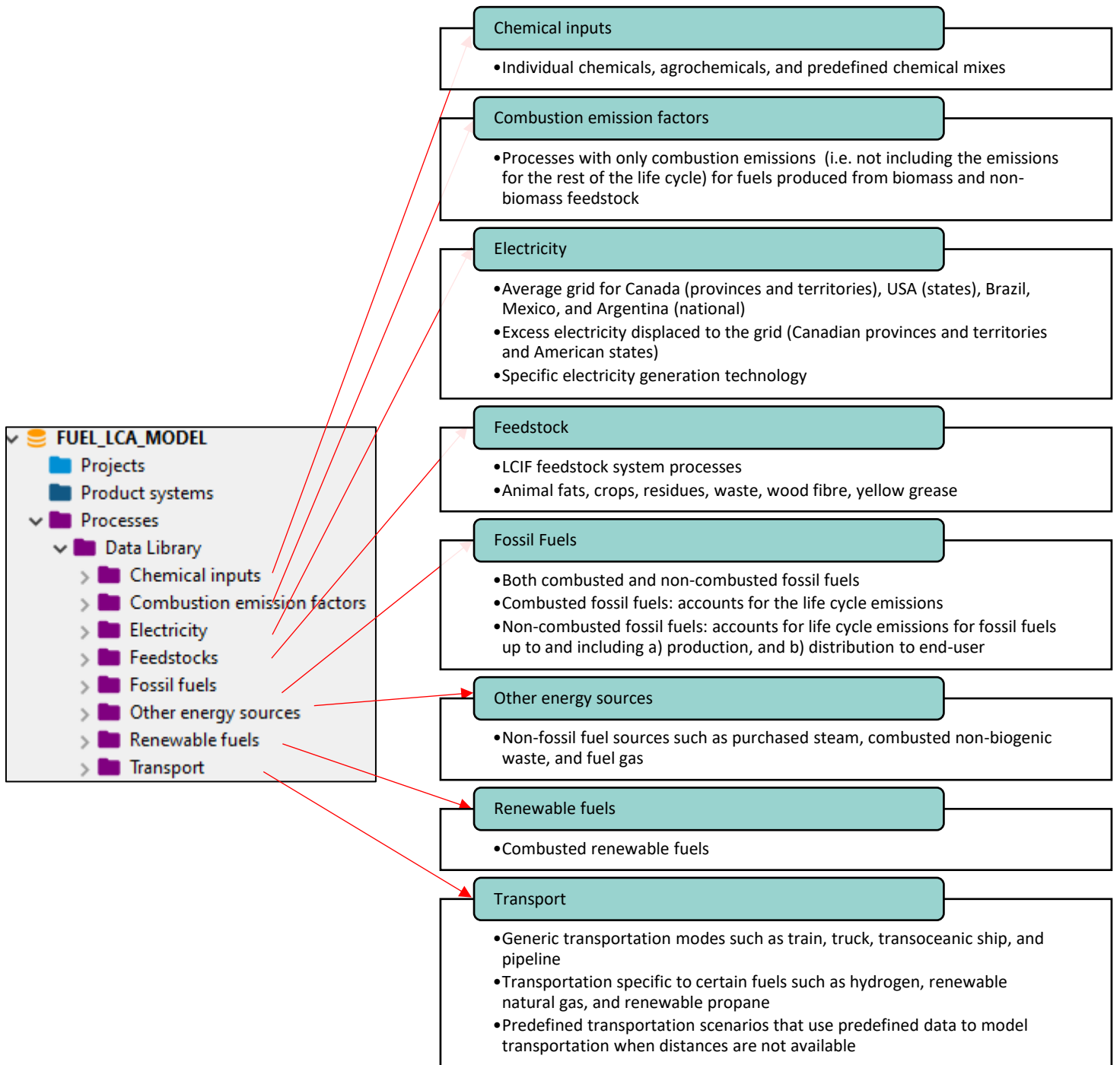
Figure 5: Left: The sub-folders of the Data Library as they appear in openLCA. Right: Overview of the system processes found in the Data Library of the Model Database.

4.2.2 Fuel pathways

The "Fuel Pathways" folder of the database contains unit processes that can be edited and used by users of the Model, as shown in **Figure 6**. Each of these unit processes is structured to model various functions needed to calculate a CI while using the system processes from the Data Library. When using the Model outside of a specific program, **Chapter 5.1** shows how the fuel pathways can be used to model specific CIs. The development of the fuel pathways and configurable processes is documented in the *Fuel LCA Model Methodology*.
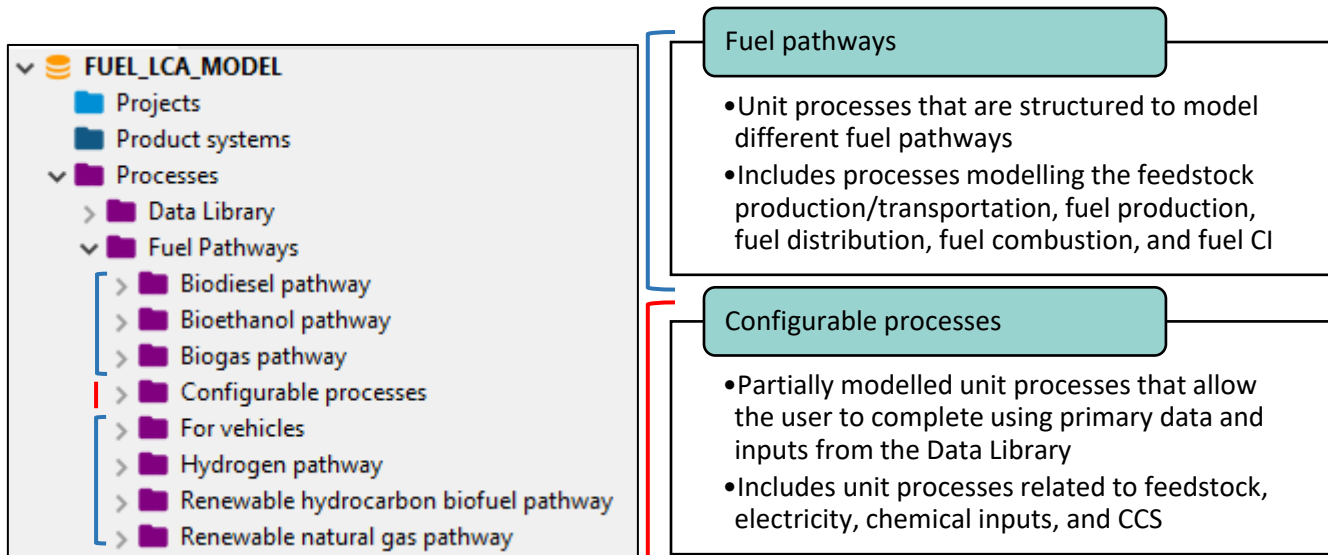
Figure 6: Left: Layout of fuel pathways section of the Model Database as seen in openLCA. Right: Overview of the contents found in the fuel pathways folder.

4.2.3 Layout of a process in openLCA

Whether the process is a system process or a unit process, all processes have the same layout in openLCA. **Figure 7** displays the main screen that is shown when opening a process. As shown in the figure, each process has eight tabs that contain different information related to modelling the process (see the red box at the bottom of the screenshot). The process name is always indicated at the top of the process, regardless of which tab is open. Here is a brief description on the contents and use for each tab:

- **General information**: Contains the process name, general description, unique identifier, and location in the database. This tab also contains some of the metadata information for system processes. To learn what information is available in the metadata, please see **Chapter 4.5**.
- **Inputs/Outputs**: Displays the flows that enter and exit the process. This is where most of the modelling is performed. The inputs contain flows entering the process, while the outputs contain flows leaving the process. For information about the interaction between processes and flows, please see **Chapter 6.2.1**.
- **Administrative information**: Contains metadata, including the intended application and access and use restrictions of the Model.
- **Modelling and validation**: Contains additional metadata related to data collection and modelling assumptions. The top of the tab displays whether the process is a system process or unit process. To learn what information is available in the metadata, please see **Chapter 4.5**.
- **Parameters**: Outside the scope of the Model.
- **Allocation**: Is used to calculate allocation factors when there are co-products (excluding GHGs) in a process. For more information, see **Chapter 6.2.7**.
- **Social aspects**: Outside the scope of the Model.
- **Impact analysis**: For the Data Library, shows the CI of the system process. For a unit process, shows the CI of the direct contributions of the process (no upstream contributions). For more information, see **Chapter 5.2.10**.
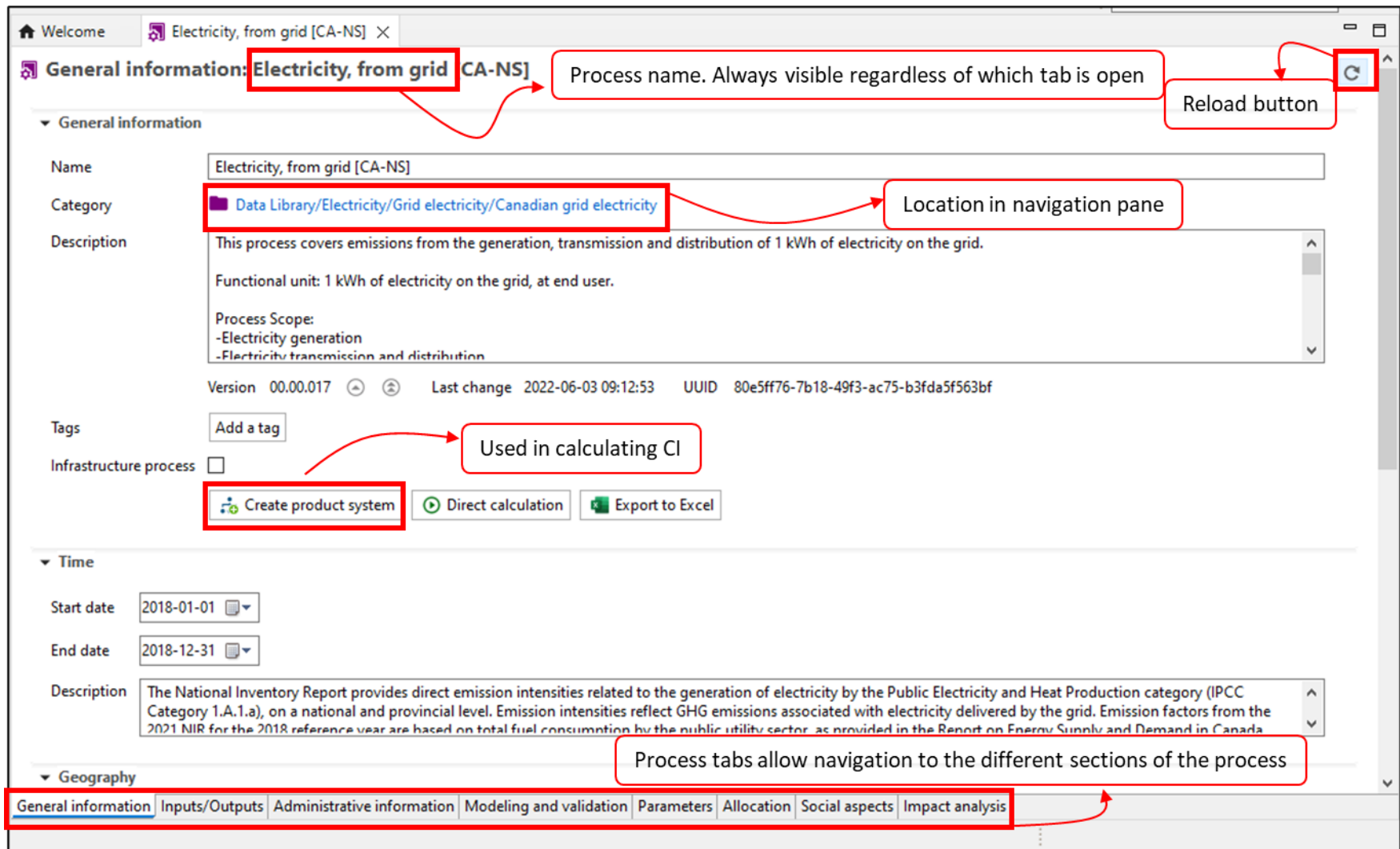
Figure 7: Layout of a process in openLCA

4.3 Flows

This section explains the organization of the flows within the Model. There are three types of flows: elementary flows, intermediate flows, and waste flows. The database groups the elementary flows in one folder and in another folder, the intermediate and waste flows.

More information about the concept of flows is available in **Chapter 6.2.1**.

Flow: material or energy stream entering or leaving a process. Can include elementary flows (GHG emissions) and intermediate flows (between unit processes), including waste flows.

4.3.1 Elementary flow organization

The elementary flows are separated into two main categories: emissions to air and supplemental emissions to air. The contents of each of the categories are in **Figure 8**. The emissions to air category regroups the most common GHG used for CI calculations. The system processes in the Data Library only use elementary flows from this category. However, users have the ability to use any of the elementary flows included in the database whenever relevant or allowed by relevant program specifications. Each elementary flow is in units of mass and has a corresponding impact factor (or global warming potential) according to the 100-year time horizon based on either the Intergovernmental Panel on Climate Change's (IPCC) Fifth Assessment Report (AR5) or Sixth Assessment Report (AR6) in the life cycle impact assessment (LCIA) methods (see **Chapter 4.4**).

Elementary flow: flow that is exchanged with the environment, e.g. GHGs

All elementary flows

- Units of mass
- Associated with an impact factor

Emissions to Air

- Contains the most commonly released GHGs:
- Carbon dioxide (biogenic, fossil, land transformation, equivalent)
- Nitrous oxide
- Methane (biogenic, fossil)
- Sulfur hexafluoride

Supplemental Emissions to Air

- Contains additional GHGs and are organized into several categories:
- Bromocarbons, Hydrobromocarbons and Halons
- Chlorocarbons and Hydrochlorocarbons
- Chlorofluorocarbons
- Fully Fluorinated Species
- Halogenated Alcohols and Ethers
- Hydrochlorofluorocarbons
- Hydrofluorocarbons

Figure 8: Elementary flow organization. Left: The elementary flows as seen in openLCA. Right: Types of elementary flows included in the Model Database.

4.3.2 Intermediate flow organization

The intermediate flows are organized according to the same folder structure as the processes. Each intermediate flow must come from a corresponding unit process or system process. For the Data Library, each system process has a corresponding output flow that matches its name. These flows correspond to the **reference product** of each process, as described in **Chapter**

> **Intermediate flow:** flow that is exchanged within the technosphere i.e. human control. In the context of the Fuel LCA Model, any flow that is not an elementary flow.

6.2.5 (in openLCA, these flows are called "product flows"). For the unit processes in the fuel pathways, this is also mostly true, but there are additional flows corresponding to common coproducts in each fuel pathway folder. **Figure 9** shows the organization of the intermediate flows in the database.
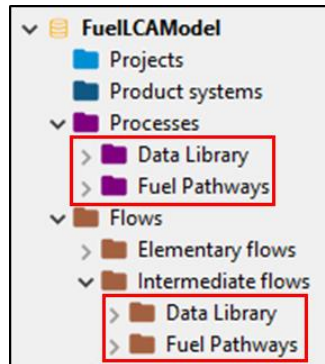
Figure 9: Intermediate flow organization.

4.3.3 Waste flow organization

Waste flows are also found in the intermediate flow folder and are organized similarly to the coproduct flows. They are found in each "Fuel pathway" folder that includes waste process modelling.

4.3.4 Layout of a flow in openLCA

Each type of flow has the same layout in the openLCA software. **Figure 10** displays the main screen that is shown when opening a flow. The red box at the bottom of the screenshot shows the tabs available in each flow. Each flow has either two tabs (intermediate flow) or three tabs (elementary flow) that contain different information related to its modelling. The flow name is always shown at the top of the flow, regardless of which tab is open. Here is a brief description on the contents and use for each tab:

- **General information**: Shows the flow name, any applicable description, and other general information like the molecular formula. Note: It also shows whether the flow is an elementary flow or an intermediate flow.
- **Flow properties**: Shows the units of measurement for the flow. While the openLCA software can allow multiple types of units for a particular flow, the Model only uses one unit type per flow.
- **Impact factors**: This tab only appears for elementary flows. It shows the LCIA method (see **Chapter 4.4**) and the associated impact factor, in units of g CO_2e per kg of CO_2. Note that impact factors are assigned to flows through the impact assessment method, not in the "Flow" itself.

Figure 10: Layout of a flow in openLCA

4.4 Life cycle impact assessment (LCIA) method

Life cycle impact assessment (LCIA) methods are used to calculate the environmental impacts of a life cycle assessment. LCA can be used to assess different kinds of environmental impacts, such as climate change or air pollution. The Model is capable of calculating one kind of environmental impact, being carbon intensity, expressed in grams of CO_2 equivalents per MJ of HHV energy.

> **Life cycle impact assessment (LCIA) method:** method that quantifies the environmental impacts of a product life cycle. In the Fuel LCA Model, converts GHG emissions into CO_2e.

There are two pieces in openLCA that are used to calculate CIs: LCIA methods and impact categories. The LCIA methods list the different impact categories used for their calculations. Impact categories contain conversion factors (called impact factors) that relate the impacts of different elementary flows to a reference elementary flow.

The Model has two LCIA methods and two impact categories (see **Figure 11**), all of which are shown below. Information about the concept of the LCIA method is available in **Chapter 6.2.6**.
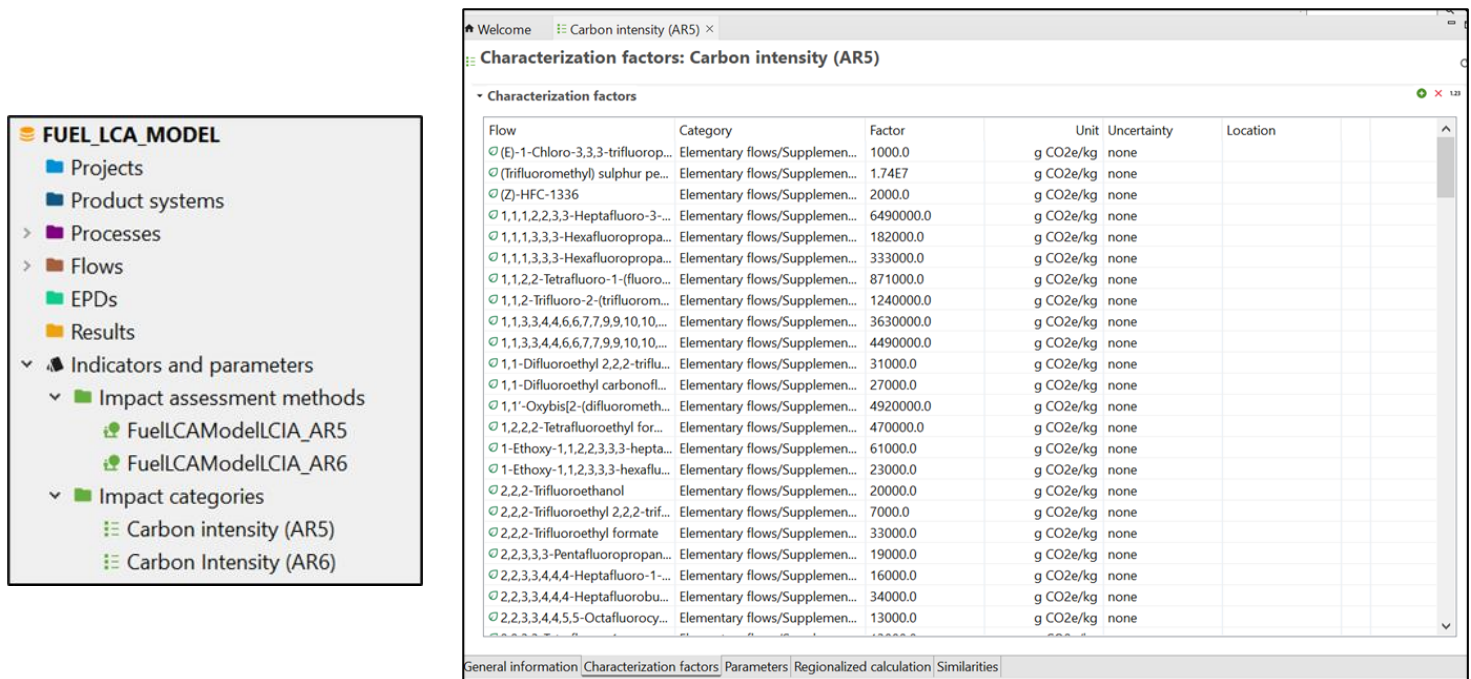
Figure 11: LCIA methods and impact categories of the Model. Left: screenshot of where to access and view the Fuel LCA Model LCIA methods and impact categories in the navigation pane in openLCA. Right: screenshot of the characterization factors tab of the carbon intensity impact category in openLCA.

4.5 Model metadata and supporting information

4.5.1 Model metadata

Each of the system processes in the Data Library contains metadata to document the process scope, usage, modelling assumptions and technical details. The metadata organization is explained below. Each heading corresponds to an information tab and the field within the specific tab, accompanied by an overview of the information found.

The fuel pathway unit processes use the main description field in the "General Information" tab to describe how the process should be used. The other fields are not used. Configurable unit processes contain similar metadata as the system processes, but also include additional instructions on how to use them.

General Information

Description

- General information on the process being modelled, including type of technology modelled, types of feedstock used for modelling the process (if any), and broad overview of life cycle stages.
- Functional unit. (See Chapter 6.2.5 definition of functional unit).
- Process Scope: defines the gate-to-gate boundaries of the process. The possible life cycle stages are:
  - Feedstock extraction
  - Feedstock transportation
  - Production/conversion
  - Distribution
  - Use/combustion
- Modelling overview which is more detailed than the general info at the beginning of the process description.
- Allocation method with the list of co-products that are included in the modelling, if applicable.

Time

- Main data sources (author, year of publication)
- The data collection period.

Geography

- The geographical scope, including information on the regionality of the data used. For example, it could note if international data or data from another region is used as a proxy.

Technology

- A detailed engineering description of the operations or activities involved in the process.
- If applicable, the literature source(s) that the engineering process was based on.
- For transportation & distribution processes, the modes of transportation and the assumed distances.

Inputs/Outputs

Outputs (description)

- Information about the flow related to its modelling. For the Data Library, this description will usually be "life cycle emissions".

Modeling and validation

Modelling constants

- Functional unit (same as in the General Information section).

Data completeness

- Cut-off criteria for inclusion of inputs and outputs.
- An overview of excluded processes and the rationale for why these were excluded (if applicable).

Data selection

- Main data sources, along with a description of the sources.

Data treatment

- Allocation method with the list of co-products that are included in the modelling, if applicable.

4.5.2 Technical flow properties and unit groups

Each flow in the Model Database has a technical flow property, which is defined by a unit group. The unit groups contain all unit conversions relative to the reference unit of the specified group. For example, the technical flow property of "Mass" is associated with the "Units of Mass" unit group, as seen in **Figure 12**. This unit group contains 14 different units of mass all converted to kg.

While users will not need to create any new unit groups or technical flow properties, they are available in the database for information. **Figure 12** shows the location of all flow properties and unit groups.
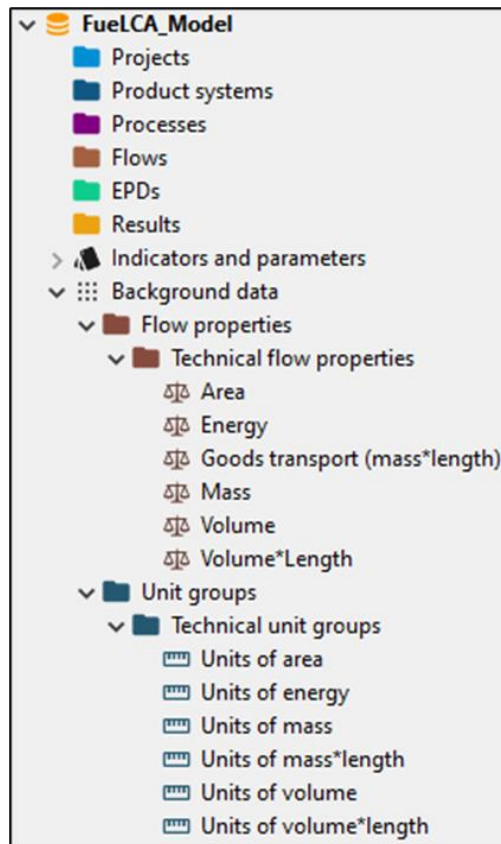
Figure 12: The flow properties and unit groups of the Model Database in openLCA

4.5.3 Sources

The "Sources" folder of the Model Database contains the references used to model the system processes in the Data Library. Each source represents a different reference and contains the full citation and if applicable, the URL and/or reference year. The sources used in a system process can also be seen in the

To view/access the sources of a particular system process, go to the sources section at the bottom of the modelling and validation tab. Double-click a source to open the full citation.

"Modelling and validation" tab of the system process. **Figure 13** shows the location of the sources in the Model Database in openLCA.
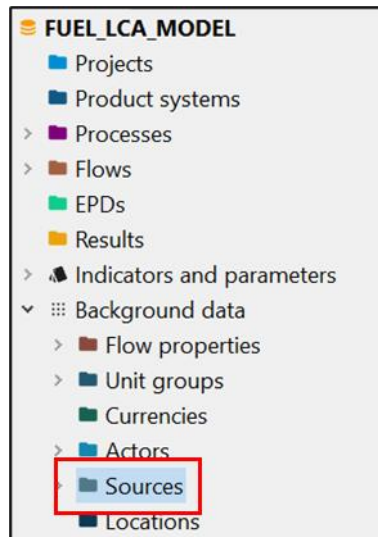
Figure 13: Sources location in openLCA

Chapter 5: Using the Fuel LCA Model

This section describes the steps involved in performing operations in openLCA when using the Model. In addition, it includes analysis and troubleshooting options in openLCA and other useful tips.

5.1 Using the fuel pathways and Data Library

The fuel pathways and the Data Library are the main parts of the Model Database that are used to model CIs. This section presents the modelling approach when using the Model for general purposes (i.e. when the Model is not used under a specific program). Be sure to refer to program specific documentation when using the Model for a specific program.

As mentioned in **Chapter 4.2**, the Data Library contains many system processes developed by ECCC that can be used as building blocks when modelling a CI using the fuel pathways. The relationship between the Data Library and Fuel Pathways is shown in **Figure 14**, and represents the modelling approach to follow when using the fuel pathways in the Model.
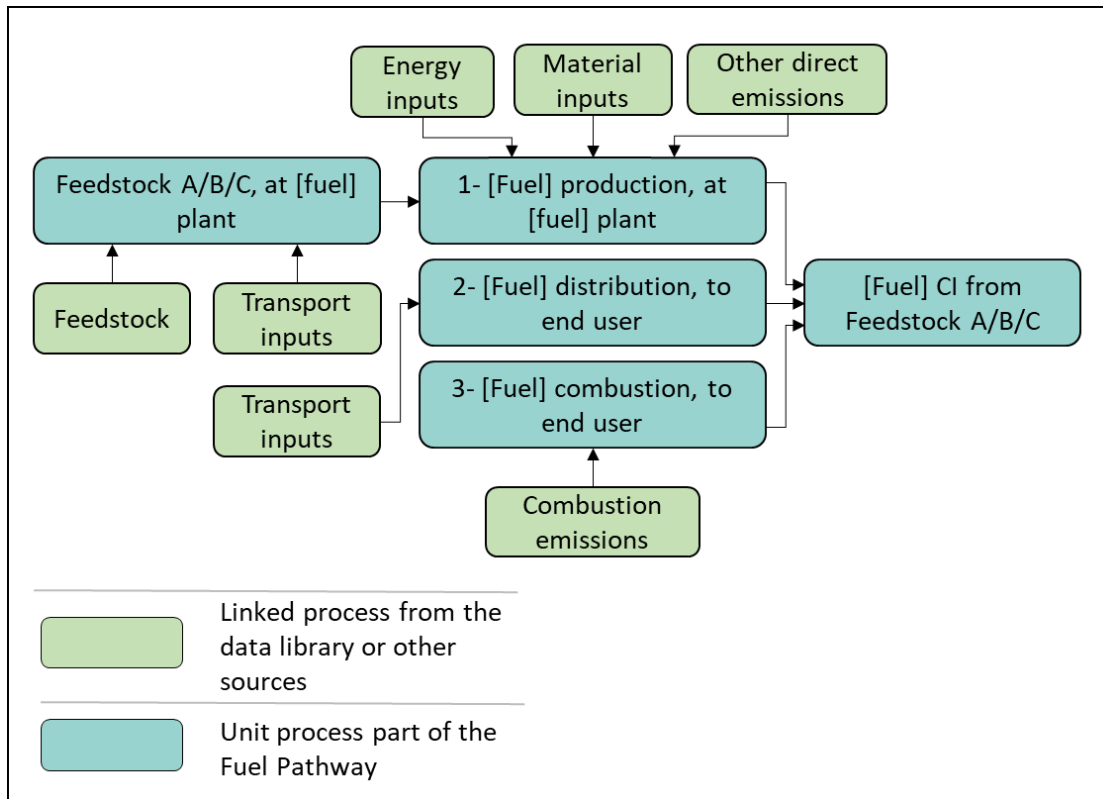
Figure 14: General interaction between the Data Library and fuel pathways

As shown in **Figure 14**, users can populate the fuel pathways by adding inputs into the fuel pathway processes. They can then enter data into the fuel pathway processes to quantify the amounts of each input and output. The fuel pathways also allow the input of data from sources other than the Data Library. A completed fuel pathway can then be used to calculate a CI.

The following section shows how to perform the operations needed to use the Model in openLCA. A complete example of using the fuel pathways and Data Library is available in **Appendix A**.

5.2 Performing operations in openLCA

5.2.1 Exporting a database from openLCA

Once a CI has been calculated, the edited processes can be exported. When exporting, to determine which processes/folders to select, follow relevant program instructions if available. Otherwise, users should at least export processes that were modified by the user and are connected to the pathway. To export a database, follow the steps below:

1. Make sure the database is open (its name should be in **bold**).
2. Right-click the database and click "Export…"
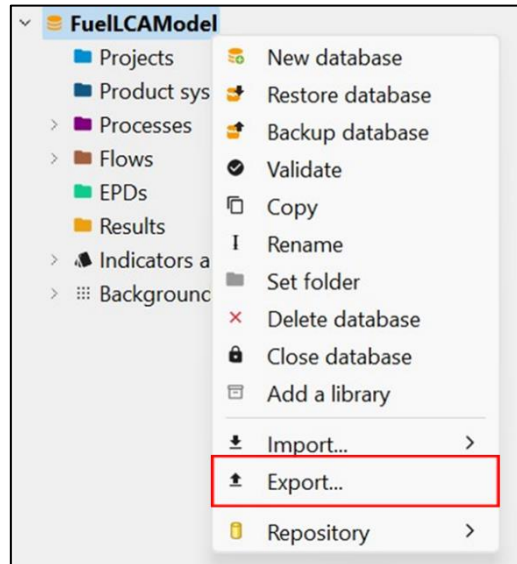
3. In the export window, click on JSON-LD at the bottom and click "Next".
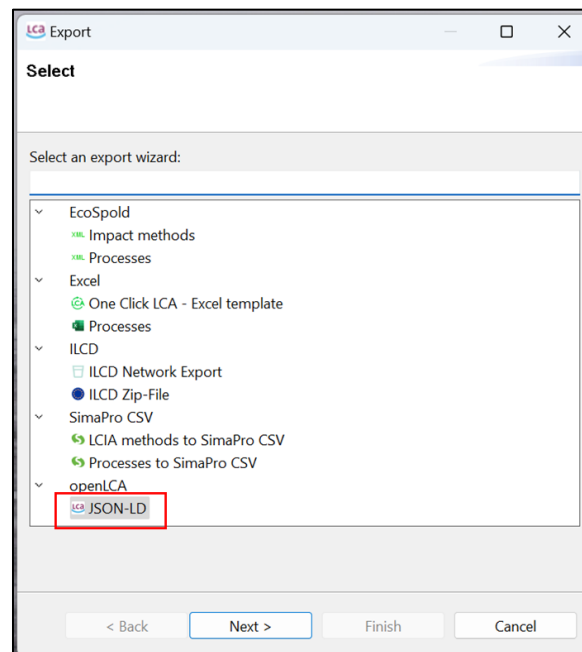
4. To change the name of the database and save location, click on the "Browse" button near the top-right of the window.

Note: Always save the database into a new zip file (i.e. do not overwrite an existing zip file). Sometimes issues arise in the exported zip file when it was created by overwriting an existing zip file.

5. Using the navigation boxes, click the boxes corresponding to the main unit processes that you edited/created to model your CI.

Note: It is not necessary to export any Data Library system processes.

6. Once you have selected your unit processes, save location, and database name, click "Finish".
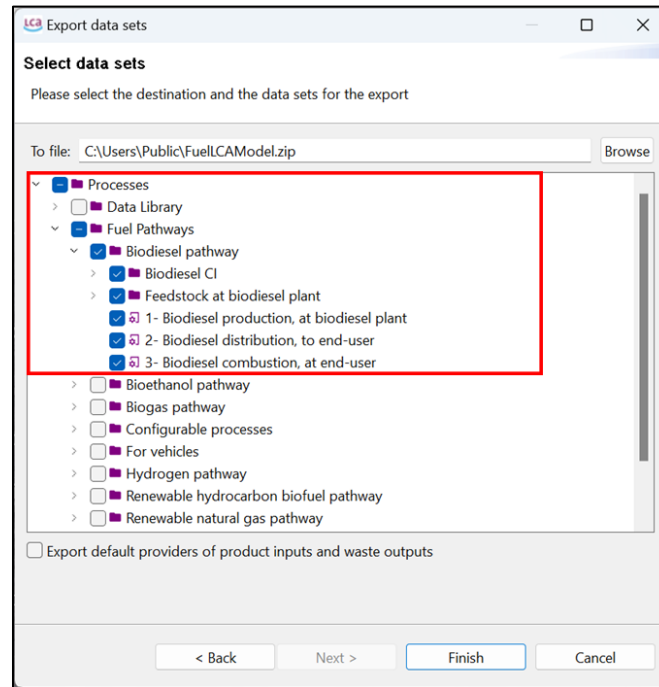
5.2.2 Inputting data into a unit process (flow information)

An important aspect of modelling a CI is double-checking the details of the entered information to ensure its accuracy. To enter values in openLCA, follow the steps below:

1. Open the unit process that requires data entry by double-clicking the unit process name on the navigation pane on the left.
2. At the bottom of the unit process window, click on the "Inputs/Outputs" tab.
3. The unit process should have some flows in the inputs and/or the outputs tab. (To learn how to add or remove flows to or from a unit process, see the next section.)
4. To enter in the value, left-click the "Amount" field for the flow in question.
   a) You can enter in any number, and the field also accepts scientific notation in the same format as Excel (ex. 3.4E-4).
   b) You can also enter in a formula if needed. To do this, simply type the formula as you would in Excel but without the equals sign (ex. 3*4-(5+1)).

   Note: if you use a formula, openLCA will display the formula by default instead of the result. To display the magnitude, click on the "1.23" button at the right side of the inputs or outputs table.

5. Make sure the value is in the correct units. To change the units, click on the "Unit" field and a drop-down menu should be available. Click on the arrow on the right side of the field to open up the menu and select the unit.

> **Note:** you can type in the unit provided that the unit is spelled exactly as it appears in the list (this is case sensitive).

6. If the value is an input for the process, make sure that the "Provider" is selected. To do this, click on the provider field for the flow and a drop-down menu should be available. Click on the arrow of the right side of the field to open up the drop-down menu and select the provider. There should only be one provider, and its name should usually be the same name as the flow.
   a) If the input flow is an elementary flow or a waste flow, a provider is not needed.
   b) For output flows, intermediate flows and elementary flows do not use providers. For waste flows, select the provider accordingly.
7. To view or edit the description of a flow, click on the "description" field, which is the right-most field for a flow. Click "Edit" to view and edit the text.

5.2.3 Adding or removing a flow to or from a unit process

As stated in **Chapter 4.3**, different types of flow may be needed to generate a CI. **Table 1** provides information on what each type of flow could represent and where to add the flow in each process. Please note that this is not an exhaustive list and as such, flows can be used for other scenarios than those listed in the table. If you are following a specific program, refer to the program documentation for instructions on which flows to add.

Type of flow	Possible scenario for the flow	Where to add the flow in the "Inputs/Outputs" tab
Elementary	Process emissions of a process	Outputs table
Intermediate	Connections to other processes. Main flows that are used in modelling	Inputs table
Intermediate	Co-product produced in a process	Outputs table
Waste	Waste produced by a process	Outputs table
	Waste used or treated by the process	Inputs table

Once you choose the type of flow you need to add, there are two ways to add a flow to a unit process. Both methods are explained below.

Adding a flow to a unit process: Method 1

1. Open the unit process where you want to add the flow.
2. Go to the "Inputs/Outputs" tab in the unit process.
3. Using the navigation pane in openLCA, find the unit process you want to link with the opened unit process.
4. Click and drag the unit process from the navigation pane into the Inputs Table or the Outputs Table of the target unit process. This adds the flow into the target unit process.
   a) If you are adding an elementary flow or an intermediate flow representing a co-product, click-and drag the flow from the flow folder.
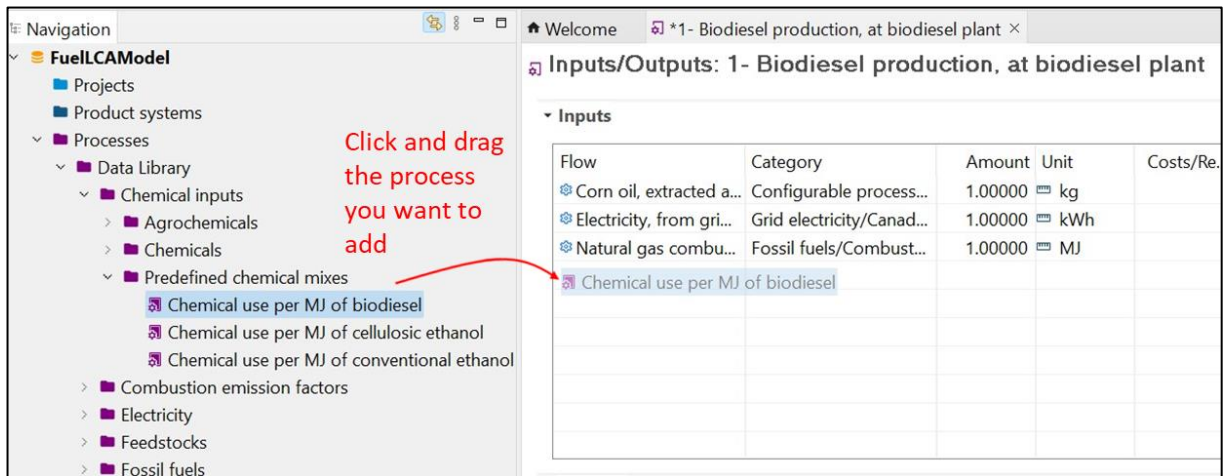
Adding a flow to a unit process: Method 2

1. Open the unit process where you want to add the flow.
2. Go to the "Inputs/Outputs" tab in the unit process.

3. At the top-right of the input or output table, click the green plus button.
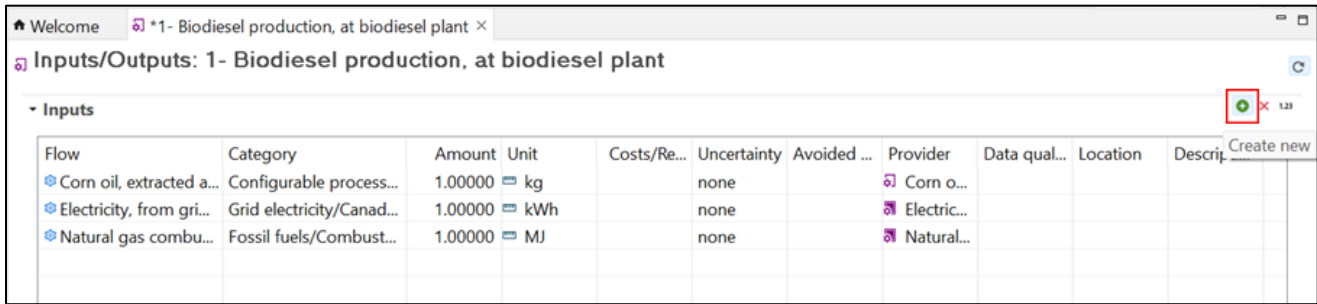
4. Using the navigation window to select the flow you wish to add. Alternatively, you can use the search bar in the window to look for the flow. Note that by doing this you may still need to expand some folder selections.
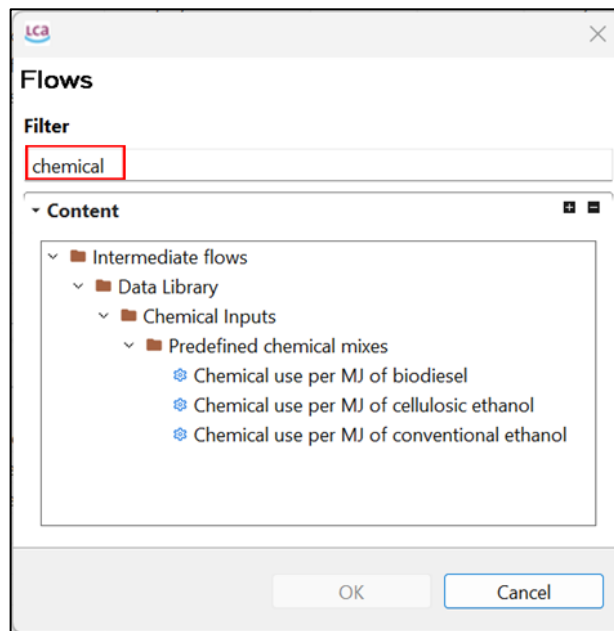
5. For intermediate flows and waste flows, since you added a flow without its linked unit process, you will need to add the provider of the flow if it is an input. To do this, click on the provider field for the flow and a drop-down menu should be available. Click on the arrow of the right side of the field to open up the menu and select the provider. There should only be one provider, and its name should usually be the same name as the flow.

> **Note**: Intermediate flows only require providers when used as inputs, while waste flows only require providers when used as output
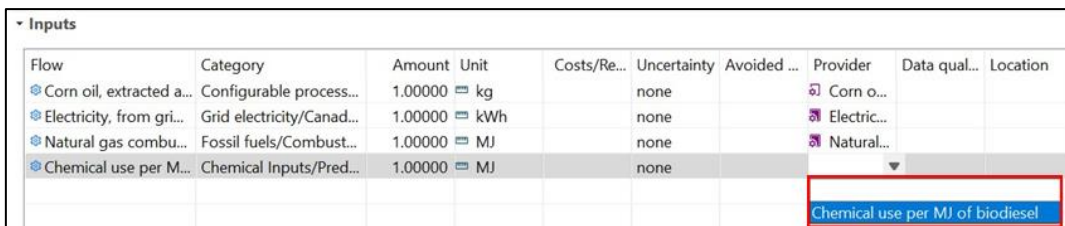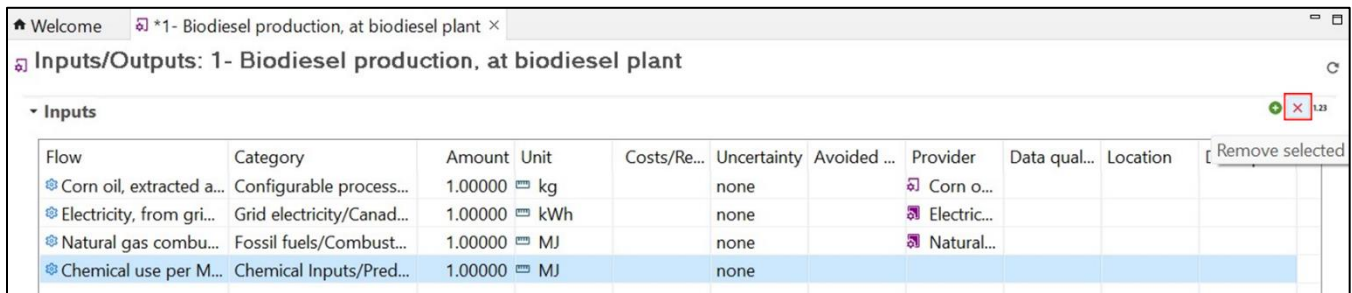
To remove a flow, follow these steps:

Removing a flow

1. Open the unit process where you want to remove the flow.
2. Go to the inputs/outputs tab in the unit process.
3. Select the flow by left-clicking the flow name.
4. Click the red X button at the top-right of the input or output table. Alternatively, you can press the delete key on your keyboard.

> **Note**: openLCA allows you to delete multiple flows at once. You can highlight multiple rows using the shift key or the ctrl key.

5.2.4 Creating a unit process

The steps for creating a unit process are shown below:

1. In the navigation pane on the left side of openLCA, right click the folder where you want to create the unit process, and click "New process".
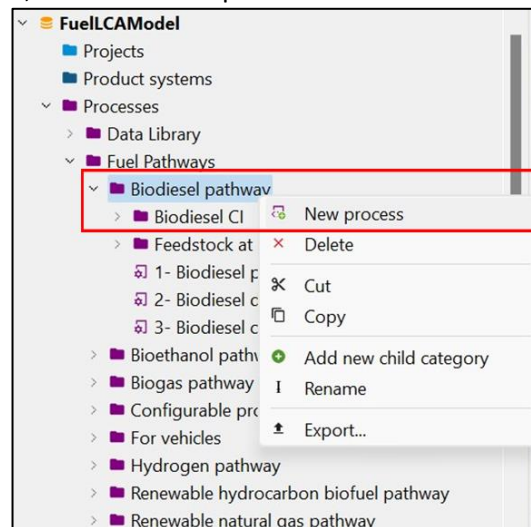
2. Enter the name of the new unit process.
3. Click the checkbox that says "Create a new flow for the process (as quantitative reference)".
4. Select the reference flow property for the new flow that will be created.
5. Click "Finish."

Note: the new flow that was created will not be organized in the right folder. Move the flow to the same folder that the unit process was created in (in the flows section of the database). For instructions on how to do this, see **Chapter 5.2.7**.
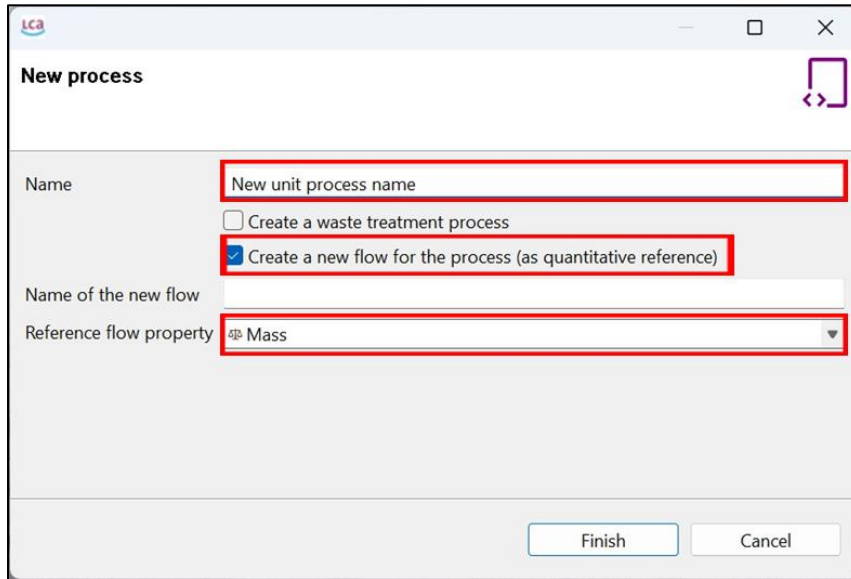
When using the model for general purposes, **Appendix A** also shows additional modelling scenarios where creating a unit process could be helpful.

5.2.5 Creating a flow

This section explains how to make a flow separately from a unit process, such as when needing to make an elementary flow or an intermediate flow (See **Chapter 4.3**). Note that when creating a new unit process (**Chapter 5.2.4**), a flow is simultaneously created (reference flow). Follow the steps below to create a flow:

1. In the navigation pane on the left side of openLCA, right click the folder where you want to create the flow, and click "New flow".
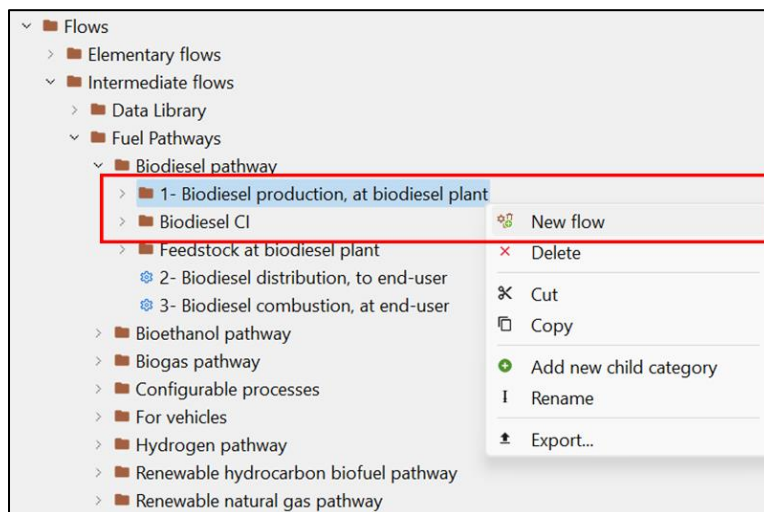
2. Enter the name of the new flow.

3. Select the flow type to choose either an elementary flow (if it is an emission), intermediate flow (product or co-product flow), or a waste flow (used for waste treatment processes).

    Note: This step is critical as you cannot change the type of flow after the flow has been created.
4. Select the reference flow property.
5. Click "Finish".
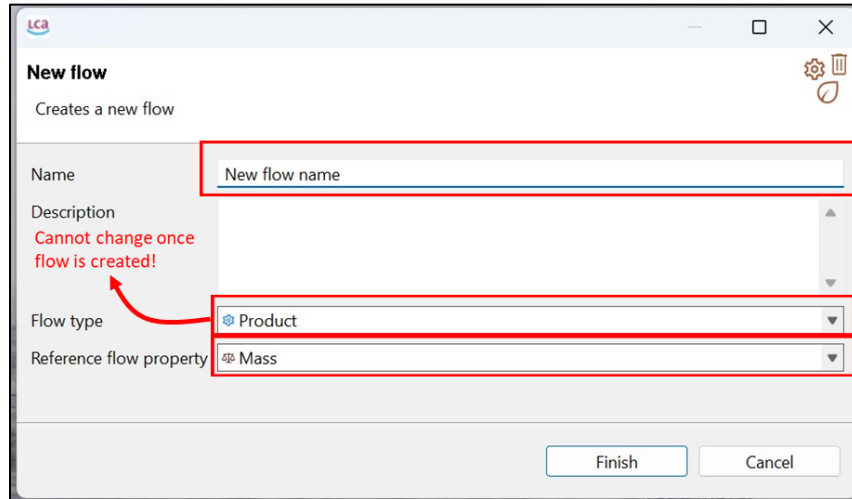
5.2.6 Renaming a unit process or flow

There are two methods to rename a unit process or a flow. They are both explained below. Note that method 2 is applicable to renaming any item in the navigation pane, including folders (but excluding system folders).

Renaming a unit process or flow: Method 1

1. Open the unit process or flow.
2. In the "General information" tab, you can edit the "Name" field at the very top of the workspace.
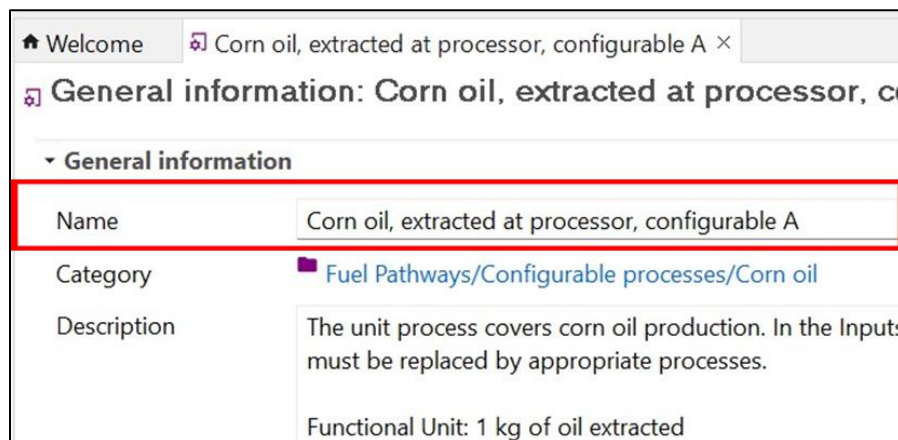
3. Save the unit process or flow.
4. If you changed the name of a unit process, you should also change the name of its reference flow. To change a reference flow name, go to the "Inputs/Outputs" tab of the unit process and then double-click on the reference flow (it will be an output and in bold).

5. Change the flow name as in Steps 1 to 3.
6. Save the changes to the flow.
7. To view the changes to the unit process, you can click the "Reload" button in the top right of the process.

1. Right-click the unit process or flow on the navigation pane.
2. Click "Rename" at the bottom of the dialogue box and enter in the new name. Note that if you change a unit process name this way, you should do the same for its reference flow. The reference flow will be saved according to the same folder structure but in the "Flows" section of the database.
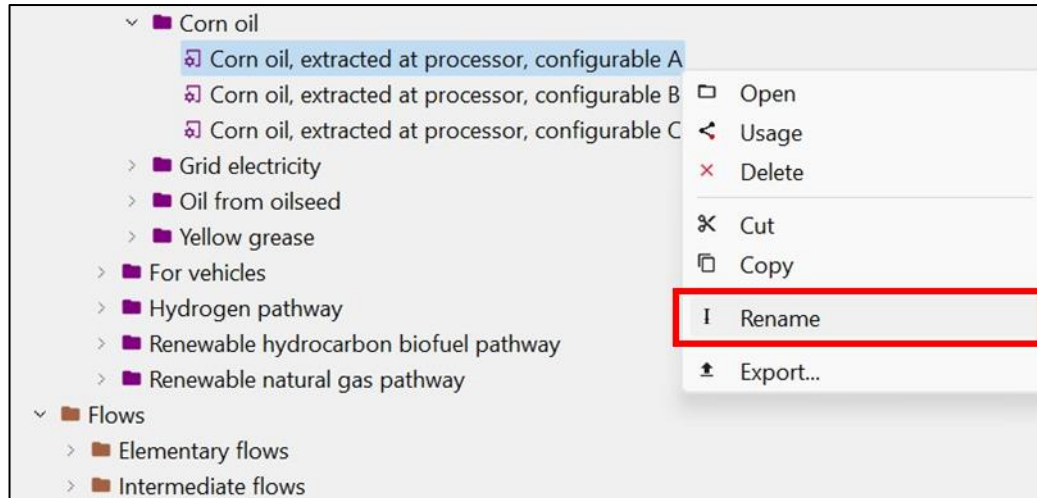
5.2.7 Moving a unit process and flow

The method to move unit processes and flows is similar to that of moving files in any folder. The steps are shown below:

1. In the navigation pane on the left side of openLCA, locate your unit process. If you are unable to find it and the unit process is open, you can see the folder location (called "Category") in the "General information" tab.
2. Once you have located your process, click and drag the process to the destination folder. Note that the process has to be moved onto the actual folder, and not onto another process within the folder. See the screenshot below.
   a) Alternatively, you can right-click on the process and click "Cut." Right-click the destination folder and click "Paste."
3. If you move a unit process make sure to do the same for its corresponding reference flow using the same steps. The flows are organized in the same structure as the processes.
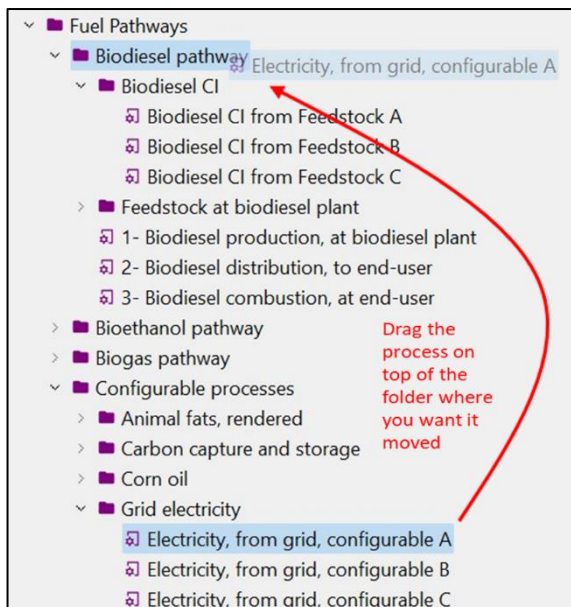
5.2.8 Performing allocation on a unit process

There are two main ways of performing allocation on a unit process in openLCA: physical allocation and the causal allocation. Economic allocation is not compatible with the Model. The steps for both methods are shown below. (IMPORTANT: The *Clean Fuel Regulations Specifications* require the use of the causal allocation method only).

Method 1: Completing the causal allocation table

1. After entering in all flow information in the unit process, go to the "Allocation" tab of the unit process.
2. Select the default method "Causal allocation".
   Note: If you do not select a default method for allocation then the allocation coefficients will not be included in the CI calculations, even if you fill in the allocation table.
3. In the "Causal allocation table", enter in the amounts for each flow by copy/pasting the entire value from a spreadsheet (e.g. Excel) into the table.
   Note: If multiple flows have the same name, you can determine which flow is which by looking at the "Amount" column.
4. When the table is completed, save the unit process.
   Note: Do not click on the "Calculate factors" button as this will replace all the values you manually entered.

1. After entering in all flow information in the unit process, go to the "Allocation" tab of the unit process.
2. Select the default method "physical allocation".
   Note: If you do not select a default method for allocation then the allocation coefficients will not be included in the CI calculations, even if you fill in the allocation table.
   Note: All of your products have to have the same flow property (i.e. unit type) for physical allocation to work.
   Note: Physical allocation will automatically use the flow property that the products are in (i.e. mass, energy, etc.).
3. Click the "Calculate factors" button.
4. A dialog box will open for calculating the default factors. Keep the default options and click "OK".

5. Save the unit process.

5.2.9 Creating a product system

A product system is a necessary step in calculating carbon intensity. The steps below describe how to create a product system.

> Once you create a product system, openLCA considers the system "finalized."
> This means that you will no longer be able to delete flows in any unit process
> included in the product system. Likewise, any new flows added will not be
> incorporated. To change any unit process, simply delete the product system
> and re-create the product system once ready.

Creating a product system

1. Open the process for which you would like to create the product system. (If you are unsure which process to use, see **Chapter 5.2.10**.)
2. In the General information tab, click on the "Create product system" button near the top of the page.
3. The New product system dialog box will open. The options here should be kept at their default settings (see image below). Click Finish to create the product system. The new product system will then open.
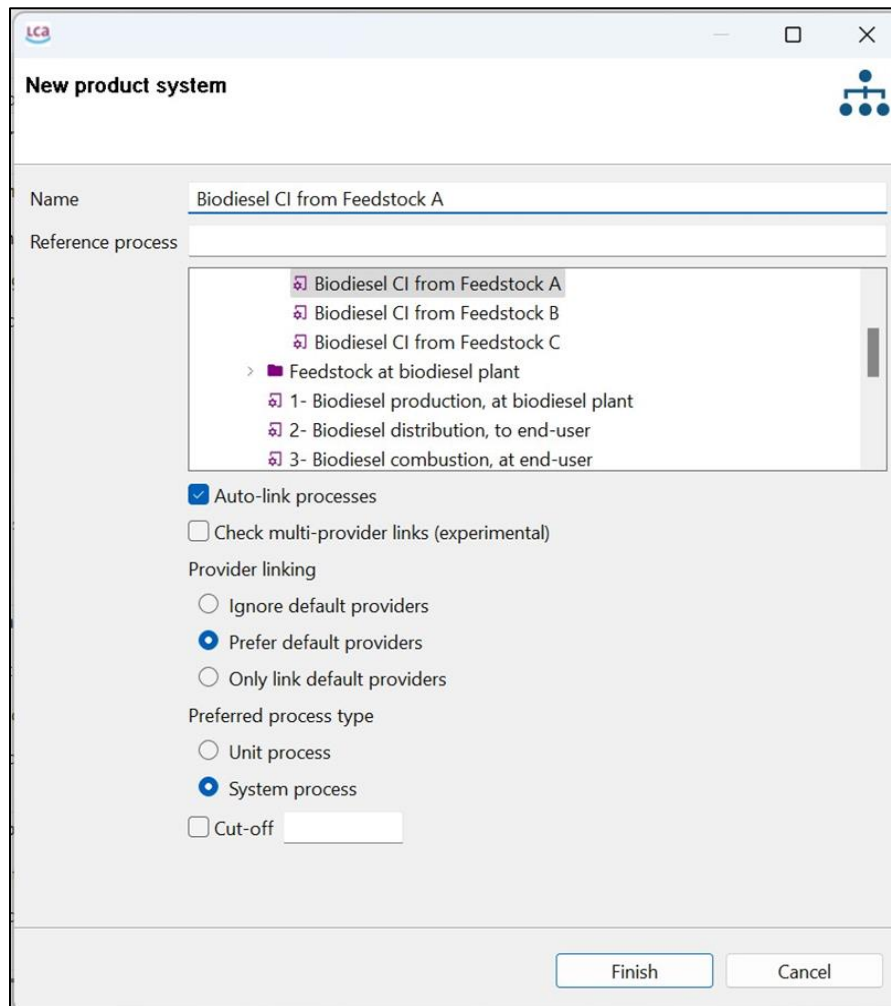
5.2.10 Calculating carbon intensity

The CI of a process or life cycle can be calculated using a product system, which is created from a process. First, you must decide what CI you are trying to calculate. Some tips are provided in **Table 2** below with respect to which process to select for the CI calculation. If you are following a specific program, refer to your program documentation for which process to use.

Table 2: Scenarios for selecting a process for CI calculation

Type of CI	Process to use for CI calculation
CI of a completed fuel pathway	"Fuel CI from Feedstock A/B/C" unit process, where "Fuel" represents the specific LCIF for the pathway in question (such as Bioethanol or Biodiesel)
CI at an intermediate step in your fuel pathway	Unit process at the desired intermediate step. All upstream processes will be included in the CI calculation, while downstream processes will be excluded
CI of a Data Library system process	Desired system process. Instead of following the remaining steps, go to the "Impact analysis" tab of the system process.

Once you have decided what process to use, follow the steps below if necessary to calculate the CI. Additional information for analysing the CI calculation is available in **Chapter 5.3**.

Note that a CI cannot be calculated from the empty fuel pathways. This is because the pathways have no inputs and zero values for the outputs. Therefore, there is nothing for openLCA to calculate by default. By filling in the fuel pathways with building blocks (processes) from the Data Library, a customized CI can be calculated. Information on filling in fuel pathways is available in **Chapter 5.1**, **Annex A**, or relevant program guidance.

Calculating a carbon intensity

1. Open the product system (if you have just made the product system it should already be opened).
2. In the "Reference" section of the "General information" tab, make sure that you are calculating your CI to the correct reference amount.

3. In the "General information" section, click on the "Calculate" button.
4. Once the Calculation properties dialog box opens, use these settings:
   a) Allocation method: As defined in process
   b) Impact assessment method: FuelLCAModelLCIA_AR5 or FuelLCAModelLCIA_AR6 (refer to specific program guidance on which method to choose)
   c) Normalization and weighting set: empty
   d) Calculation type: Eager/All
   e) Regionalized calculation: unchecked
   f) Include cost calculation: unchecked
   g) Assess data quality: unchecked
5. Click "Finish" to calculate the carbon intensity.
6. The calculated results will open in a new "Analysis result" tab in the workspace. To see the calculated CI, go to the "Impact analysis" tab. The CI will be available in the "impact result" column on the right side of the workspace. The units will be in g CO_2e / MJ. (Note: if, for example, you calculated the CI based on something other than 1 MJ (i.e. reference amount), the value will be in respect to that amount. For example, if you used 10 MJ then the CI would be in units of g CO_2e / 10 MJ.)

5.2.11 Creating a system process

The steps to create a system process are shown below.

> The system processes in the Data Library were created using another method that sums the inputs and outputs of all unit processes involved in a product system. However, this method is not needed when modelling using the Fuel LCA Model.

Creating a system process

1. Open the unit process that you would like to use to create a system process. If you are creating a new unit process, first follows the steps in **Chapter 5.2.4**.
2. When the unit process is opened, go to the "Modeling and validation" tab.
3. At the top of the workspace there will be a field for process type. Select from the fly-out menu "System process".
4. Save the process. Note that other than the reference flow, a system process should only have elementary flows (i.e. there should be no inputs).
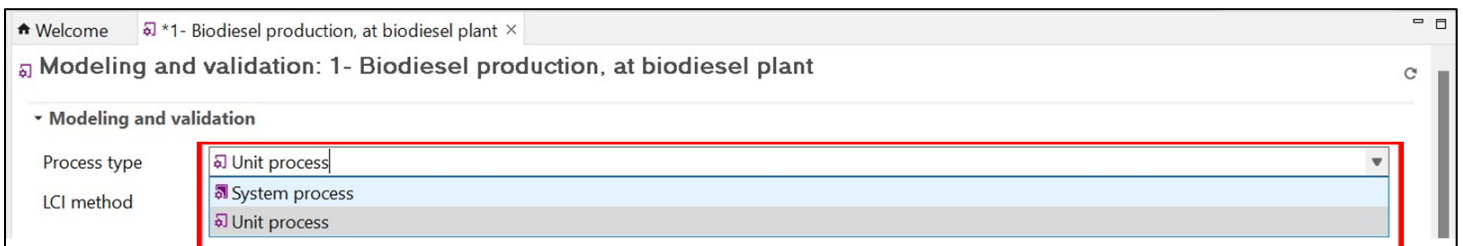
5.3 Analysis and troubleshooting options in openLCA

This section presents different tools and options in openLCA and how they can be used for both analysis or troubleshooting.

5.3.1 Model graph

The model graph is a feature that can be accessed in the dedicated tab of a product system. It can be used to view the entire life cycle of a system.

The example below shows the model graph of the product system for the process "Bioethanol from Feedstock A." Each model graph shows the processes involved in modelling the product system. Each process is represented by a different box. The process used to make the product system is shown at the end of the chain.

> The model graph is a feature that can be used for more advanced functions in openLCA outside the scope of the Fuel LCA Model. Therefore, after using the model graph for visualization purposes, do not save changes to the model graph when closing it.
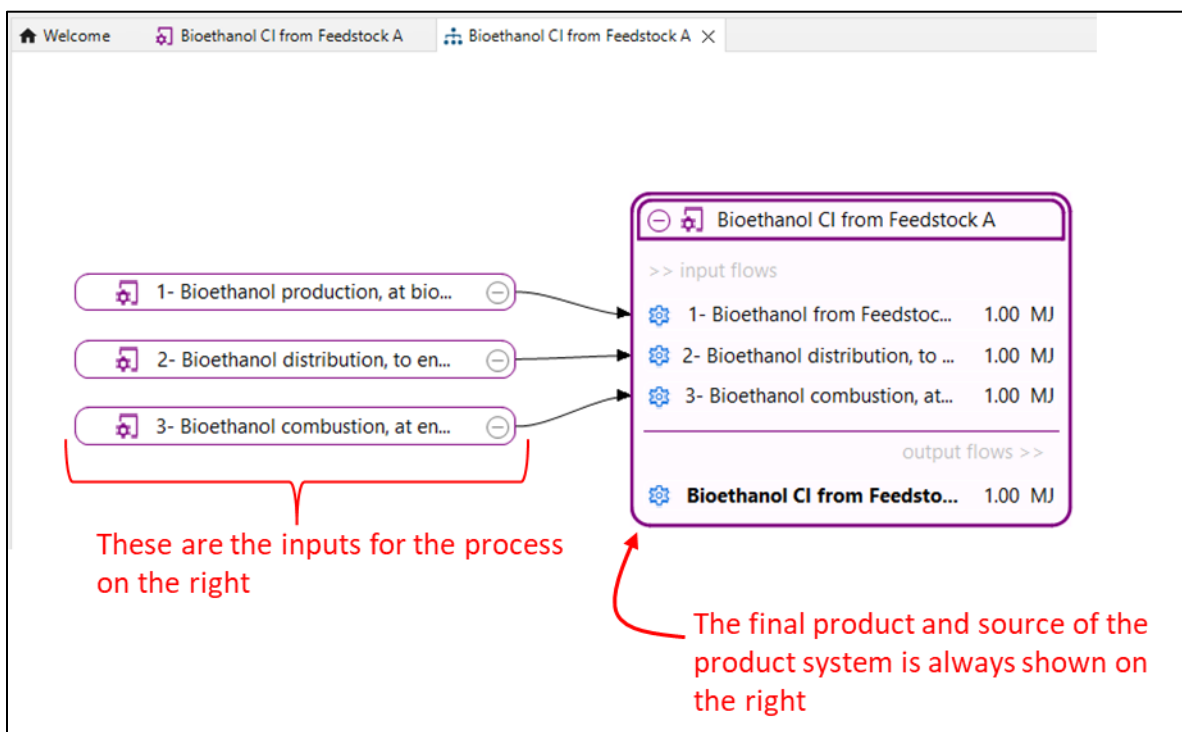
Clicking the + on the left side of a box will open up more boxes representing the inputs of a process. In the image below, the biodiesel production box has been opened up to show its inputs. Note two of the inputs (Natural gas combustion and Electricity, from grid [CA-AB]) are a different style of box because they represent system processes, in contrast with the other boxes that represent unit processes.
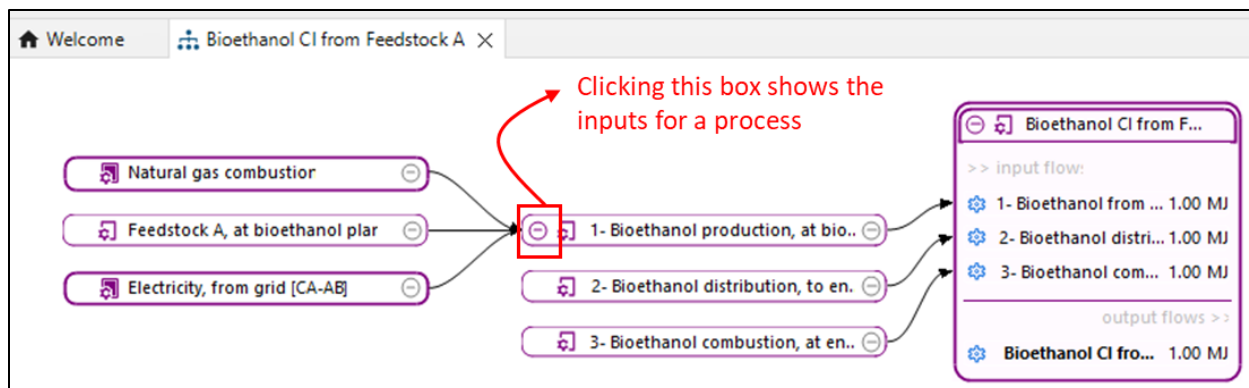
Double-clicking a box shows a different visualization for the inputs and outputs of a process. To close the expanded version of the box, simply double-click the box again.
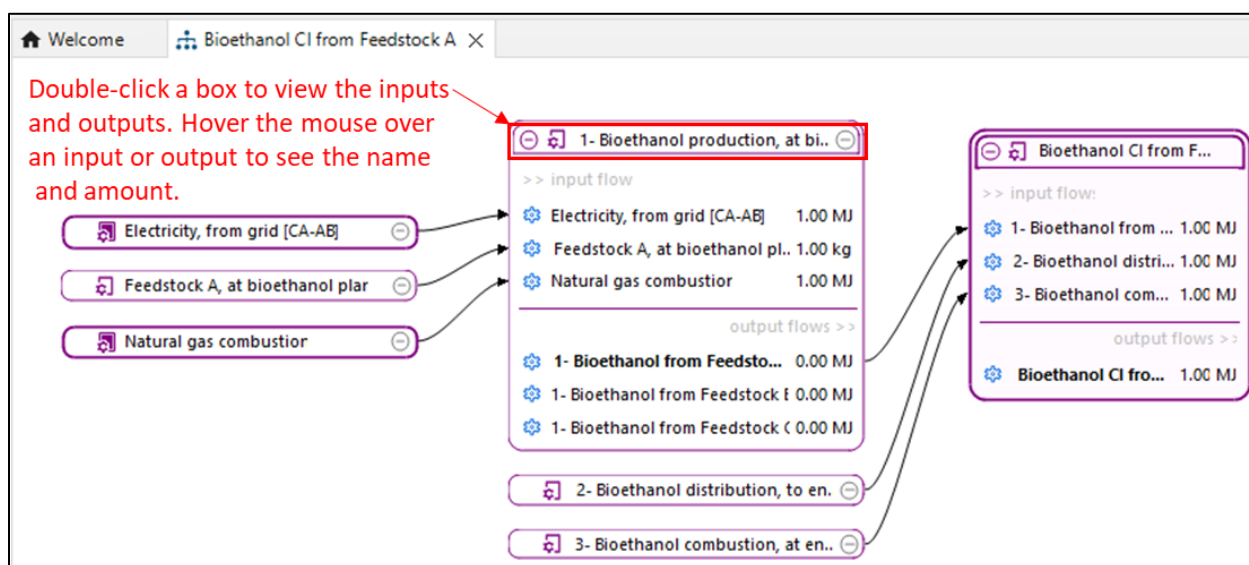
5.3.2 Inventory results tab

Once the CI of a product system is calculated, a results workspace containing several tabs opens. One of those tabs is the "Inventory results" tab.

The "Inventory results" tab is divided into three sections: Inputs, Outputs and Total requirements. Those sections respectively list the total:

- Inputs section: extractions from the environment (e.g., water)[1]
- Outputs section: emissions to the environment (e.g., fossil CO_2)
- Total requirements section: intermediate flows (e.g., electricity from the grid in Alberta)

required all along the life cycle to provide the previously defined functional unit.

Figure 15 presents a screenshot of an example of Inventory results tab created with mock-up values.

---

[1] In the Model this section is usually empty, as only GHGs are modelled and the carbon sequestrated in some land transformation scenarios is modeled as a negative CO_2 output.
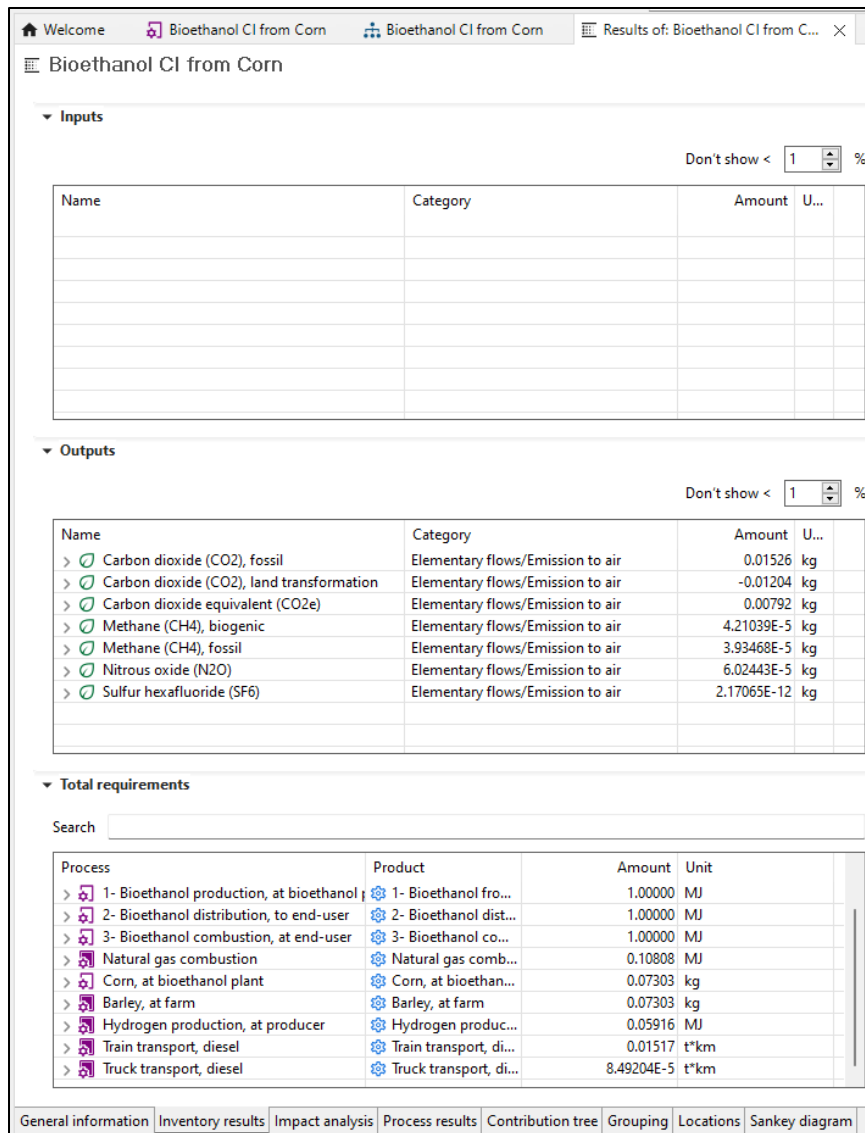
Figure 15: Example of an Inventory results tab

5.3.3 Impact analysis tab

The "Impact analysis" tab of a product system calculation result is a helpful tool. Most importantly, this is where the CI is displayed. The table below lists the intensities of all processes with emissions. Only processes from the Data Library that directly contribute to the product system are listed. Note, in the screenshot below, all CIs are mock-up values.

5.3.4 Contribution tree tab

The contribution tree is a tool available in the analysis of the results of the CI of a product system. It allows for the analysis of the entire life cycle of the product system. Recalling the model graph (refer to **Chapter 5.3.1**), the contribution tree acts like a table version of the model graph, showing the cumulative results along the pathway of a product system. The contribution tree can be used to understand the main impacts and contributors in a product system. Additionally, this is an excellent place to troubleshoot as it can help identify processes that may have abnormally high or low intensities. Any process can be opened from the contribution tree by double-clicking the process name. The numbers in the screenshot below are mock-up values.

5.3.5 "Reload" feature

The "Reload" feature is useful when implementing changes or to undo changes (that have not been saved). The "Reload" button is available in the workspace of any object that is not an "Analysis result" (i.e. processes, flows, product systems, etc.). It is located at the top right of the workspace.
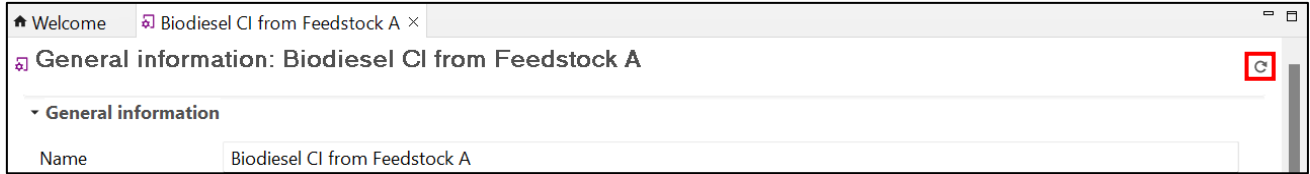
Some situations where the reload button can be used are presented below.

Example 1: You have a unit process and a flow opened. The flow is used as an input in the unit process. You change the flow name, and save the changes. The new flow name will not show up in the unit process automatically. You can reload the unit process to view the updated flow name.

Example 2: You accidentally make changes to a unit process and want to undo them. Since there is no undo button in openLCA, you can reload the unit process to revert to the original version, as long as the changes were not saved.

Example 3: You change a unit process to a system process and save your changes. Even though the changes have been implemented, the unit process icon at the top of the process (where the tab is) will not change to a system process icon until you reload the process.

5.3.6 "Usage" feature

The "Usage" function can be used to determine where a certain process or flow is used within the database. On the navigation pane on the left, right-click a flow and click "Usage". This will open a window in the workspace that shows all processes where the flow is used. If used on a process, then it will show all product systems where it is used.
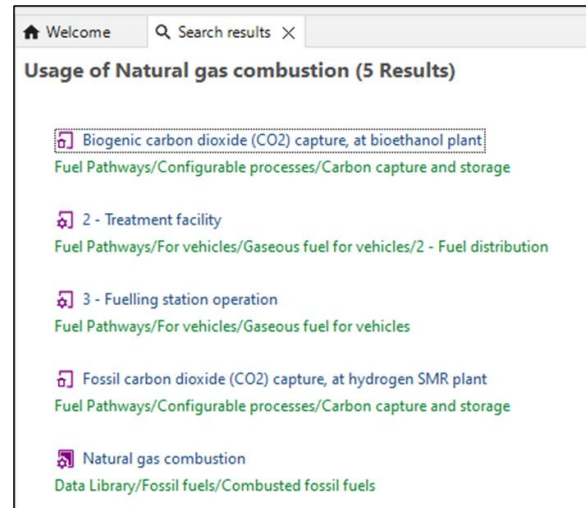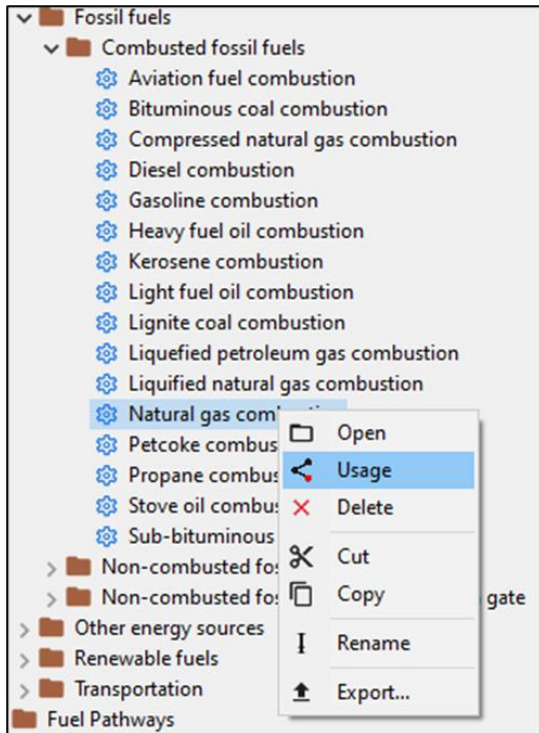
Alternatively, for flows you can open the flow and in the General Information tab you will see a section called "Used in processes." This will contain a list of any process that consumes the flow as well as any process that produces the flow. Note that if a flow is not used in any process, this section will not be visible. Likewise, if a flow is consumed but not produced, or produced but not consumed, then the "Produced by" or the "Consumed by" sections will not be visible, respectively.

5.3.7 Change flow property

The flow property defines the types of units for a flow (for example, mass, energy, distance, etc.). A change to the flow property is sometimes necessary, usually due to an incorrect flow property being assigned. Allocation, for example, can only be done on reference flows of the same flow property. To change a flow property, follow the steps below.

1. Open the flow, and go to the Flow properties tab.
2. Click the green + button to add the new flow property.

3. Once it is added, you will see the original flow property and the new flow property. Usually you only want one flow property per flow. Make the new flow property the reference property by checking the "Is reference" box.
4. Save the flow.

5. Next, you will need to change all unit processes that use the original flow property. You must do this before deleting the original flow property. (You will get an error message otherwise.) The usage feature (**Chapter 0**) can assist in identifying which unit processes use the flow. Open the unit processes that use the flow. Change the units to the new units. Note that when there is more than one flow property assigned to a flow, the individual units will all have a suffix indicating for which flow property they belong.
6. Once you have replaced all of the units, you can delete the original flow property.
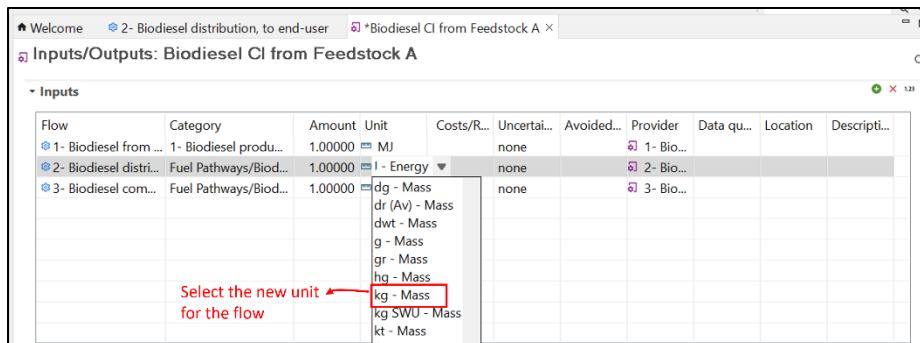
5.4 Other useful tips in openLCA

5.4.1 Copy/pasting flows from Excel

The openLCA software is compatible with certain features in Excel. Notably, it has the ability to transfer data tables of flow information from Excel to openLCA.

To do this, you need to make sure that the columns in Excel match the columns in openLCA. For example, process inputs/outputs in openLCA have the following order of information: flow, category, amount, unit, costs/revenues, uncertainty, avoided waste, provider, data quality entry and description. Therefore, the information in Excel must be formatted in the same order.

The minimum information you need is the flow name, amount and unit (the other fields can be left blank). However, you should also include the provider, and possibly a description. Note that all fields must match exactly with those in openLCA. The fields are also case-sensitive. The number format in Excel should also have no commas (even in French). An example of how the table should look in Excel is shown below:

	A	B	C	D	E	F	G	H	I	J	K
1	Flow	Category	Amount	Unit	Costs/Revenues	Uncertainty	Avoided Product	Provider	Data quality entry	Location	Description
2	Natural gas combustion	Intermediate flows/Da	3.45	MJ		None		Natural gas combustion			

Row 2 in the above picture can be copy/pasted (either via Ctrl+C/Ctrl+V or by right-clicking) into the inputs or outputs table in a process in openLCA. See the picture below for the result.

Flow	Category	Amount	Unit	Costs/R...	Uncertai...	Avoi...	Provider	Data ...	Location	Descripti...
Natural gas combustion	Fossil fuels/Combus...	3.45000	MJ		none		Natu...			

Note that you can also copy/paste the other way around, from openLCA to Excel. Always review the copy/paste data to ensure the right flows and values were carried over correctly.

Chapter 6: LCA Concepts used in the *Fuel LCA Model*

This chapter presents the concepts to understand when using the Model. Some concepts are specific to LCA, while others were developed to further define and clarify aspects of the Model. Since the Model was developed with openLCA, some of the terms or definitions match those in openLCA for consistency.

6.1 Overview of life cycle assessment

LCA studies are used to determine the impact of a product (or a service) with an approach looking at the entirety of the life cycle. For fuels, the life cycle includes the following stages: feedstock production, feedstock transportation, fuel production, fuel distribution and fuel combustion.

LCA is standardized by the ISO 14040 series to ensure transparent and reproducible results. In ISO 14040, LCA studies are defined in four phases:

- **Goal and scope definition**: Defines the depth and breadth of the LCA study depending on the goal of the particular LCA.
- **Inventory analysis**: Lists and quantifies all emissions and extractions to and from the environment involved in the life cycle of the system being studied (e.g., the total mass of methane emitted during the life cycle, expressed in kilograms of methane).
- **Impact assessment**: Converts the inventory into indicator results (e.g., the carbon footprint in kilograms of CO_2 equivalent).
- **Interpretation**: Summarizes and discusses results to draw conclusions, make recommendations, and for decision-making, in accordance with the goal and scope definition.

The four phases are shown in **Figure 16**. They are interrelated as the results of one phase can affect previous and subsequent phases. This iterative procedure achieves increasingly precise data for a more accurate result. This approach was followed in the development of the Model and is explained in the *Fuel LCA Model Methodology*.
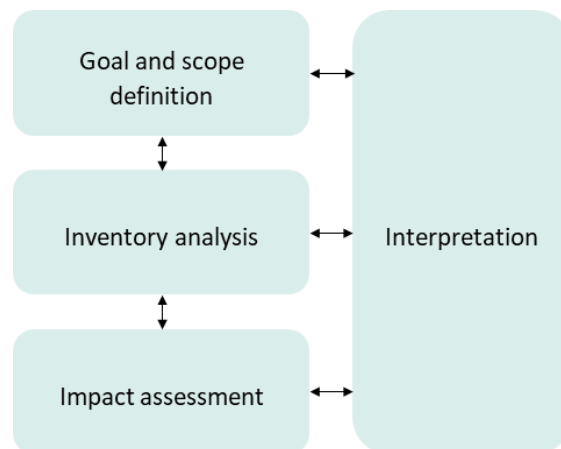
Figure 16: The four phases of an LCA study. Adapted from ISO 14040

The Model has been designed such that users follow a similar modelling procedure as ISO 14040 to calculate the CI of a fuel:

- The **Goal and Scope** are defined by the specifications of the program that uses the Model. If using the Model for more general purposes, then the goal and scope will be defined by the user, provided that they fall within the limitations and boundaries of the Fuel LCA Model described in the Fuel LCA Model Methodology.
- The Model Database is used to create the **life cycle inventory (LCI)** of the user's fuel.
- The **life cycle impact assessment (LCIA)** method included in the Model Database is then used to calculate the CI results from the LCI.
- The **interpretation** consists of reviewing the results to ensure their robustness and exactitude and preparing a CI report. **Chapter 5.3** contains information pertaining to analysis of results.

6.2 Fuel LCA modelling concepts

The following concepts are important to understand when using the Model:

- **Unit process**: Smallest divisible activity of a life cycle. It transforms quantities of inputs into quantities of outputs. It can use modeling parameters and background data.
- **Flow**: Material or energy stream entering (input) or leaving (output) a unit process. "Elementary flows" refer to exchanges between a unit process and the environment (i.e. extractions and emissions) while "intermediate flows" refer to exchanges between unit processes (e.g., electricity).
- **Life cycle stage**: Specific part of a life cycle (e.g., feedstock production). Life cycle stages are modelled by a collection of unit processes.
- **Fuel pathway**: Collection of unit processes, modeling parameters, and background data which represents the life cycle of a fuel from a given feedstock. In general LCA vocabulary, a fuel pathway is called a product system.
- **System process**: Process that contains the LCI of a group of unit processes.

These concepts and their mutual relationships are illustrated in **Figure 17**. Each concept is also further explored with examples in this chapter.
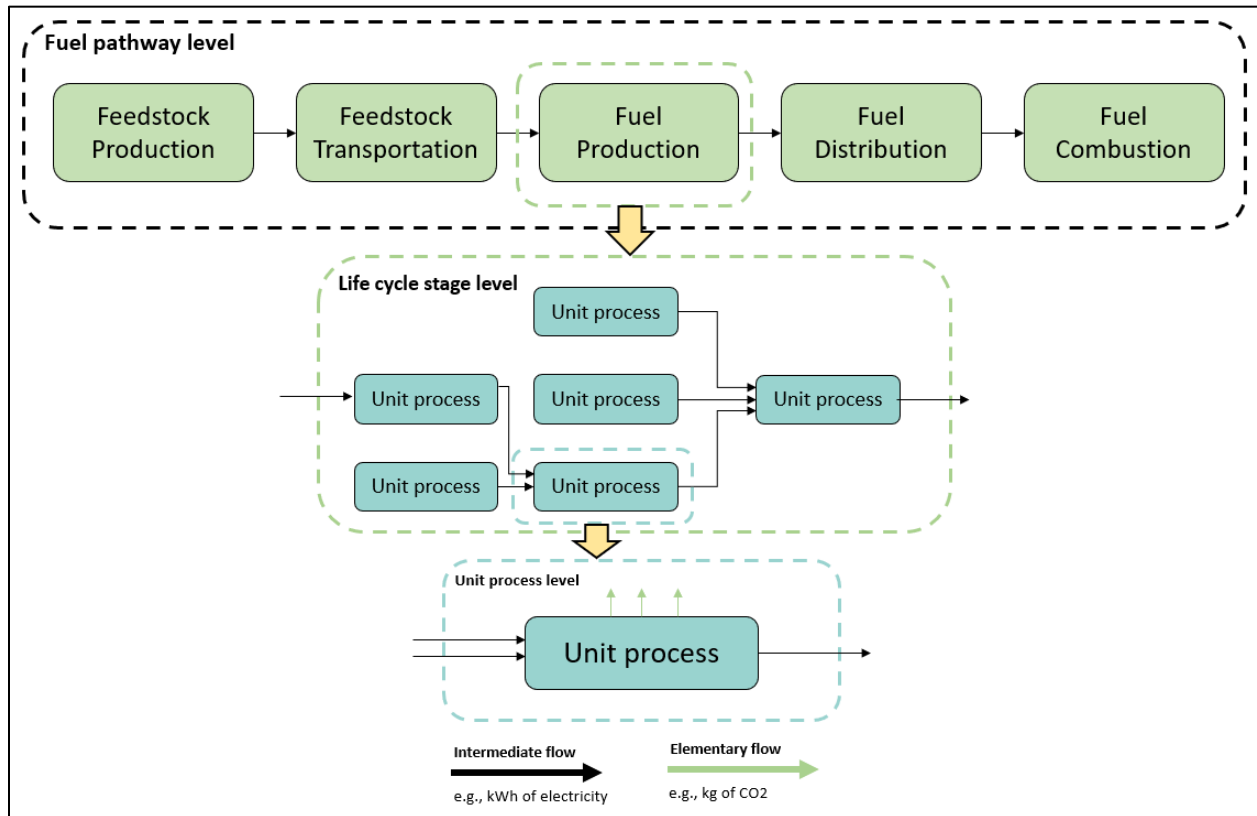
Figure 17: Levels involved in an LCA

6.2.1 Unit processes and flows

Unit processes

Unit processes represent the smallest divisible process of a life cycle. They transform quantities of inputs into quantities of outputs. They can use modeling parameters and background data. When performing LCA, most of the work is done at the unit process level. Life cycle stages are modelled by a collection of unit processes.

A unit process has two main functions:

Unit processes represent **independent** mass/energy balances. This means that while the quantities within a unit process must be balanced, the inputs and outputs of the unit process do not rely on the quantity of inputs and outputs of other unit processes.

1. Transform inputs into an outputs
2. Serve as an independent mass/energy balance

55

Flows

Flows represent material or energy streams entering (input) or leaving (output) a unit process. There are two types of flows: elementary flows and intermediate flows.

Elementary flows

Elementary flows refer to exchanges between a unit process and the environment. They can either **enter** the unit process (e.g., a mineral extracted, land used, etc.) or be **released** from it (e.g., a gas emitted into the air, a liquid released in a river, etc.). As the Model focuses on CI, relevant elementary flows as outputs are GHG emissions, and the uptake of CO_2 from air is a relevant example of an elementary flow as an input.

Intermediate flows

Intermediate flows refer to exchanges between two unit processes. All unit processes produce at least one intermediate flow (called reference flow) which is then used as an input to other unit processes.[2] For example, a corn cultivation unit process produces "Corn, at farm" as an intermediate flow. Subsequently, a truck transportation unit process can be used to transform the "Corn, at farm" flow to a "Corn, at bioethanol plant" flow.

Waste flows are a type of intermediate flow used in the openLCA software. They represent intermediate flows that are not used to create value, i.e. waste material sent to end-of-life treatment or waste energy that is not recovered.

Interaction between unit processes and flows

Unit processes are tightly linked to the flows they receive and produce. Using a process flow diagram, as illustrated in **Figure 18**, unit processes are represented by boxes, while flows are represented by arrows. Intermediate flows connect boxes, while elementary flows are only connected to a box at one end. Waste flows also connect boxes. The receiving box of a waste flow represents a waste treatment process.
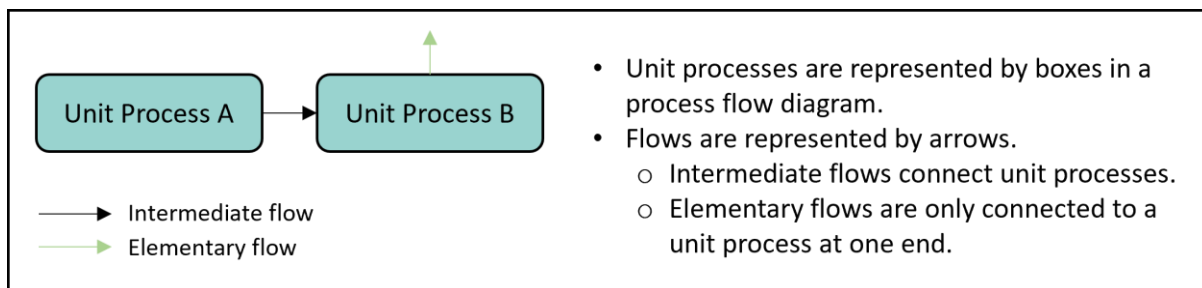
Figure 18: Interactions between unit processes and flows in a process flow diagram.

---

[2] If more than one intermediate flow is produced by a unit process, an allocation rule is required to attribute the environmental burdens of the unit process to each of the intermediate flows it produces (see **Chapter 6.2.7** on allocation).

Providers

OpenLCA uses the concept of **providers** when linking unit processes with intermediate flows. Providers link unit processes between each other to create a model. This ultimately allows the inclusion of the environmental burdens from all processes for the calculation.

In the case of the Model, the names of the providers are identical to the names of the reference flows. The case of several intermediate flows produced by a unit process is an exception to this rule, in which each intermediate output flow name could be different from the unit process name.

Figure 19 illustrates an example of inputs/outputs tabs with providers in openLCA. Note that waste flows require a provider when they are in the outputs table.

Figure 19: Providers as used in openLCA

Example: Bioethanol production from corn: unit process and flows

An example of a bioethanol production unit process is displayed in **Figure 20**. In the figure, the input and output flows are displayed, as well as the amounts and units for each flow. Note that the amounts and flows for this example are for illustrative purposes only. The inputs table of the production process contains electricity, chemicals, natural gas combustion, and corn as input flows. Each input flow comes from its own unit process with its own modelling. For example, this means that there is another process called "Electricity, from grid" (whose name matches the flow name in the input table) that models the emissions produced in electricity generation from the grid. The purpose of the "Electricity, at grid" flow inside the "Bioethanol production" process is to quantify the amount of electricity used to produce bioethanol.

The outputs table includes the bioethanol produced as an output (reference product, highlighted in **bold**) and animal feed co-product. The outputs table also includes an elementary flow, biogenic carbon

dioxide[3], which represents the process emissions associated with the conversion process. Process emissions only include those associated with the reactions involved in making the bioethanol. For example, the emissions associated with combusting the natural gas input are already accounted for in the natural gas combustion unit process. This properly accounts for GHG emissions for each unit process and avoid double counting.

If another process requires the use of bioethanol as an input, then the "Bioethanol production, from corn" flow will appear in the inputs table of the process, thus linking the process with the "Bioethanol production, from corn" process.

Since each unit process is a mass-energy balance, the quantities of input and output flows are quantified relative to the quantity of the main output (reference flow) of the unit process, which in the example below is the produced bioethanol. For example, this process requires 26 000 MWh of electricity per 5900 TJ bioethanol (i.e. 0.044 kWh/MJ bioethanol). This allows for proper scaling if the bioethanol is used by another process.

Bioethanol production, from corn

Intermediate flows (blue) come from other unit or system processes

Inputs

Flow	Amount	Unit
Electricity, from grid	26,000	MWh
Chemical use per MJ of conventional ethanol	5,900	TJ
Natural gas combustion	1,700	TJ
Corn, at bioethanol plant	587,000	t

The amounts of each flow (both inputs and outputs) are relative to the reference flow

Outputs

Flow	Amount	Unit
Bioethanol production, from corn	5,900	TJ
DDGS from corn	3,400	TJ
Carbon dioxide, biogenic	7,730	t

Elementary flows (green) are exchanged with the environment

Figure 20: An example of a bioethanol production unit process

---

[3] The Model sets biogenic emissions to zero because the impact factor is zero for this type of emission. This flow is included only for the benefit of this example.

6.2.2 Fuel pathway

The Model contains several fuel pathways intended to be representative of common LCIF life cycles. Each fuel pathway represents the life cycle of an LCIF, usually starting at the feedstock production and ending at fuel combustion. A fuel pathway is modelled using a collection of unit processes, modelling parameters, and background data allowing the determination of the CI. Users are able to complete these pathways using their data and building blocks known as system processes, which are stored in the Data Library of the Model Database. An example of a fuel pathway is provided below.

Example: Corn-based bioethanol fuel pathway

As shown in **Figure 21**, the corn-based bioethanol pathway includes corn cultivation, corn transportation, bioethanol production from corn, bioethanol distribution and bioethanol combustion. Each of these stages considers the material and energy inputs required, the process emissions, as well as potential co-products.

Bioethanol from corn fuel pathway

Corn cultivation → Corn transportation → Bioethanol production → Bioethanol distribution → Bioethanol combustion
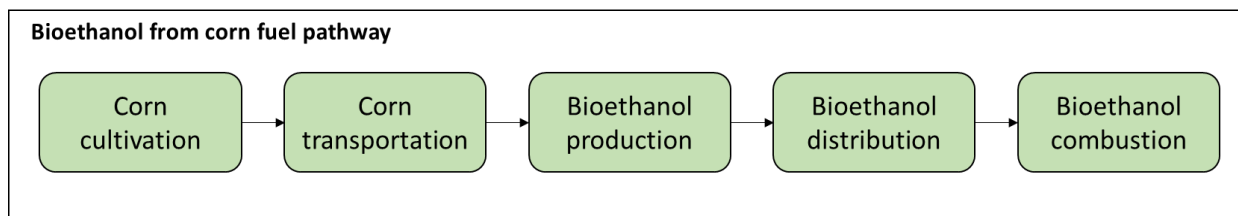
Figure 21: Example of a fuel pathway in the Model

Product system

A product system is, according to ISO 14040, a collection of unit processes that altogether represent the entire life cycle of a product (in the case of the Model: a fuel). The Model uses the term Fuel Pathway to cover this definition. In the case of openLCA, product systems are created from processes and used to calculate CIs. See **Chapter 5.2.9** for more details on product systems in openLCA.

6.2.3 Life cycle stage

In the Model, each fuel pathway is comprised of up to five life cycle stages, which are outlined in **Figure 22**. This provides a structure that regroups all activities and emissions that are part of the fuel pathway. Life cycle stages are connected by the product (or service) that flows between them. For example, in the case of the bioethanol from corn pathway presented in **Figure 23** corn is going from the corn cultivation stage to the corn transportation stage, and from the corn transportation stage to the bioethanol production stage, while bioethanol is going from the bioethanol production stage to the bioethanol distribution stage, and from the bioethanol distribution stage to the bioethanol combustion stage.

Feedstock Production: resource acquisition (e.g., natural gas extraction, or soybean cultivation, etc.) and transformation (e.g., natural gas upgrading, or soybean oil extraction, etc.) into substances ready for transport to the fuel production plant.

Feedstock Transportation: transportation of feedstock from its last transformation activity to the fuel producer.

Fuel Production: (or fuel conversion) conversion of feedstock into fuel, including potential pre-processing of feedstock, and post-processing and upgrading of fuel to final fuel product.

Fuel Distribution: storage and handling of fuel, transport of finished fuel product to storage and to final user.

Fuel Combustion: combustion of the final fuel product by the end user.
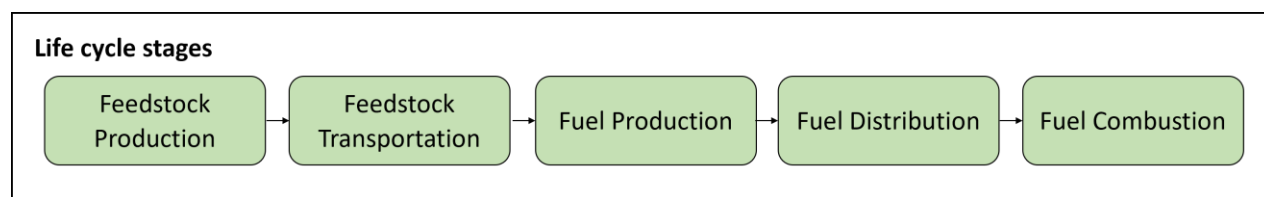
Figure 22: The five life cycle stages of LCIF in the Model

Functionally, life cycle stages are collections of unit processes connected by a network of flows, which are explained in the following sections.

Example: Bioethanol production life cycle stage

Figure 23 below shows some of the unit processes involved in the fuel production life cycle stage of bioethanol from corn. Each box in the figure represents a different unit process of the life cycle stage. The output of the life cycle stage is the produced bioethanol. All unit processes within the life cycle stage are needed to model the bioethanol production. Note that the "Corn, at bioethanol plant" unit process is part of the feedstock transportation life cycle stage. Similarly, the "Bioethanol production" unit process is then connected to the next life cycle stage: fuel distribution.
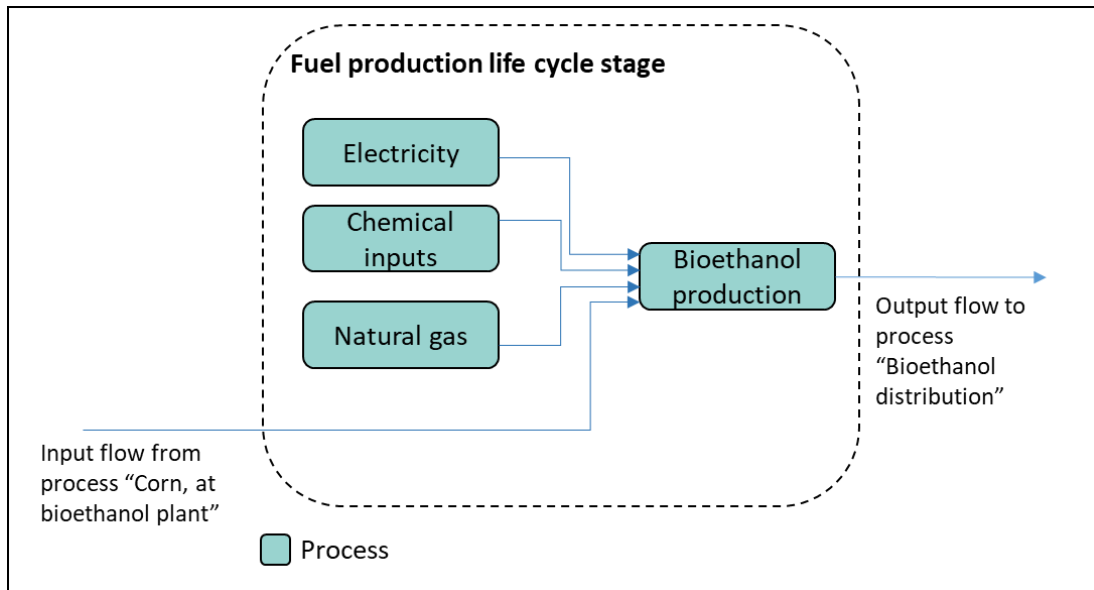
Figure 23: Simplified example showing some of the processes involved in a bioethanol production life cycle stage

6.2.4 System processes

System processes are a type of process used in openLCA. While a unit process represents the smallest divisible process for which inputs and outputs are quantified, a system process is an aggregation of unit processes that sums all elementary flows involved. Unlike a unit process, a system process has no intermediate flows in the inputs or outputs, except for the flow of the reference product. It only contains cumulative elementary flows representing the LCI of the aggregated unit processes.

Example: illustrating the concept of a system process

Figure 24 illustrates an example of a collection of unit processes and their corresponding system process. Their functional unit (see the following section) is the same: 1 MJ of "Z". The elementary flows (GHG emissions) of the system process are equivalent to the sum of the elementary flows involved in the collection of unit processes it represents (e.g., 1 kg of CO_2 from process X + 2 kg of CO_2 from process Y + 5 kg of CO_2 from process Z = 8 kg of CO_2). In other words, the inventory of the system process is identical to the inventory the collection of unit processes it represents. Therefore, their CI is also identical.
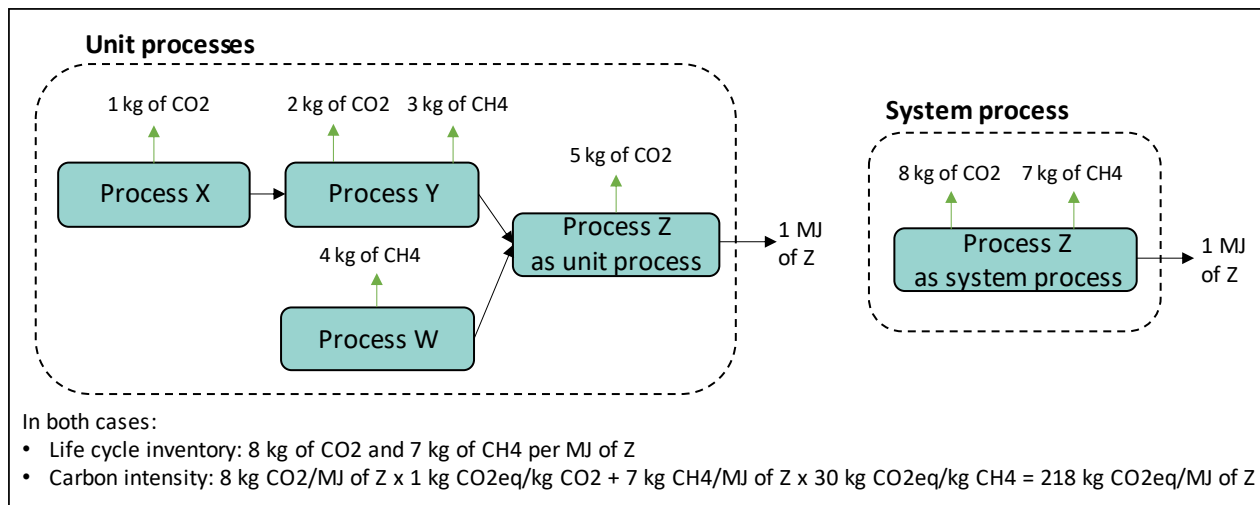
Figure 24: Illustration of the concept of system process and the differences between a collection of unit processes and a system process.

System processes in the Data Library

As seen in **Chapter 4.2.1**, the Model Database includes a Data Library of system processes that can be used when modelling a fuel pathway. These system processes represent an aggregation of all the unit processes required to model the life cycle of the product or the function they represent.

In developing the Model Data Library, ECCC created hundreds of unit processes modelling several aspects in the Canadian industries (as well as some international unit processes). ECCC used these unit processes to create the system processes available in the Data Library. The development of the unit processes is explained in the *Fuel LCA Model Methodology*. **Figure 25** illustrates how unit processes are aggregated into the system processes found in the Data Library. This is further explained in the *Fuel LCA Model Methodology*.
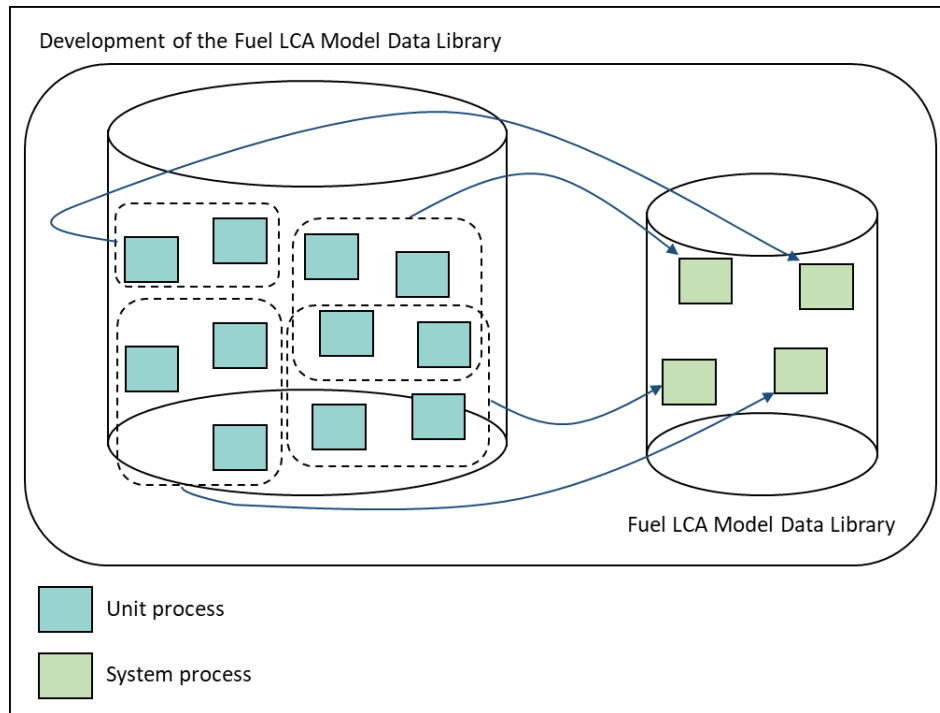
Figure 25: Visualization of the development of the Model Data Library

6.2.5 Functional unit

According to ISO 14040, the **functional unit** defines the quantification of the function (performance characteristic) of the product (flow). It is defined to allow a fair comparison in LCAs. The system boundaries and the scaling of an LCA model are then built in accordance with the functional unit.

In the case of the Model, the main functional unit is "1 MJ of energy content based on the HHV delivered to the end user and used for its energy content". As a consequence, the reference flow of the unit processes used to create product systems is usually 1 MJ HHV of an LCIF. **Chapter 4.5.1** shows where the reference flow is documented in the metadata of each process of the Data Library.

6.2.6 Life cycle impact assessment (LCIA) method

An LCIA method is composed of one or several sets of characterization factors, each set representing an environmental issue. While LCIA methods can include a wide range of environmental issues such as acidification, toxicity or water scarcity, the LCIA of the Model is limited to a CI.

A characterization factor is an impact intensity value associated to a specific elementary flow and a specific environmental issue. It allows the translation of the LCI into potential impacts. The characterization factors used in the Model are the GWP for a 100-year time horizon based of IPCC's Fifth Assessment Report (AR5) and Sixth Assessment Report (AR6). The results calculated from the Model are expressed in grams of CO_2e per MJ of HHV energy delivered to the user.

6.2.7 Allocation

In most cases, unit processes are monofunctional. This means they are associated with one single product (i.e. one single intermediate output flow) called the reference product. However, some unit processes are multifunctional (i.e. have several products) and in certain cases data is not available to

separate and attribute the environmental burdens (in the case of the Model, GHG emissions) of the process to each of its products. Those environmental burdens are associated to elementary and intermediate input flows as well as elementary output flows (i.e. all non-product flows).

In those cases, **allocation** is used to partition the non-product flows using a set of allocation factors. ISO 14040 defines the requirements to perform allocation, namely that allocation should be avoided if data can be collected individually for each product or the system boundary can be expanded to encompass them. Otherwise, an allocation procedure must be applied.

The following sub-sections explain the main types of allocation found in the Model. For instructions on when to use each type of allocation, please refer to program specifications when applicable. Otherwise, ISO 14044 contains guidance for determining the appropriate allocation method.

Energy-based allocation

The default allocation method for most unit processes in the Model is the energy basis. In the case of a production process which generates a fuel (reference product) and co-products, the non-product flows are partitioned between the fuel and co-products based on the energy content (HHV) of these outputs. Energy-based allocation is part of the "physical" allocation methods in openLCA.

Figure 26 illustrates how energy allocation divides the environmental burdens based on the energy content of each product. In the figure, the unit process generates two products: 0.25 MJ of product 1 and 1 MJ of product 2. As a consequence, energy allocation attributes 20% (=0.25 MJ/(0.25 MJ + 1 MJ)) of all non-product flows to product 1 and 80% (=1 MJ/(0.25 MJ + 1 MJ)) of all non-product flows to product 2.
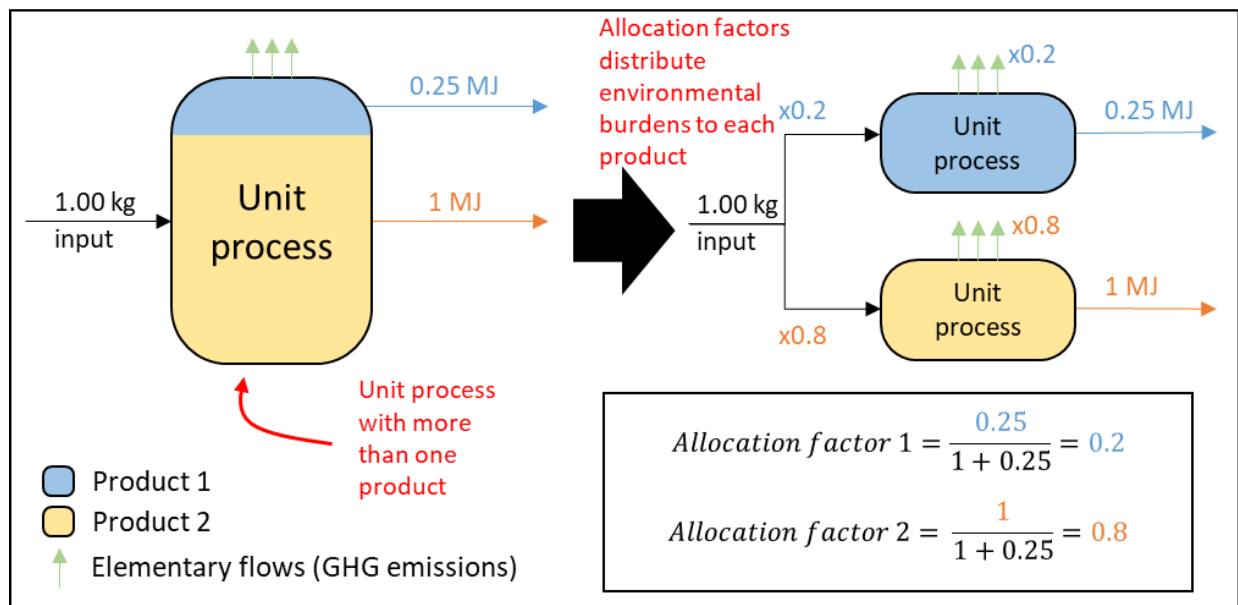

Figure 26: Example: energy allocation partitions the environmental burdens of a multifunctional unit process through the use of allocation factors that are based on the energy content of each product.

Mass-based allocation

Mass-based allocation attributes environmental burdens based on the mass of each product. The Model uses mass allocation for wood fibre, animal fats, and oilseed feedstock production processes. An example for mass-based allocation would be similar to the one in **Figure 26**, but the amounts of products would be expressed in terms of mass (e.g., kg) instead of energy. Mass allocation must be based on the dry mass (wet mass from which the water content is removed). Mass-based allocation is part of the "physical" allocation methods in openLCA.

Cut-off allocation

The Model applies the "cut-off" allocation approach to waste recycling, except if the use of the waste material for LCIF production results in significant and real methane reductions (see section on avoided product allocation below). Under the "cut-off" allocation approach, if a waste material (first life) is used for another purpose (second life) instead of disposal, the producer of the waste material is not attributed any burdens for disposal, and the user of the waste material is not attributed any burdens for the upstream production of the material. Consequently, waste products used as feedstock are represented in the Model by empty processes (and associated with zero carbon intensity).

Avoided product allocation (system expansion)

System expansion is not allocation, but rather a method to avoid allocation, as described by ISO 14044. System expansion considers that the products of multifunctional processes will be used by other processes, instead of the products they usually use. Those products usually used are called "avoided products" or "displaced products". To evaluate the environmental burdens associated with one of the products of a multifunctional process, the environmental burdens of the products displaced by the other products of this multifunctional process are subtracted from those of the multifunctional process (e.g., using the example from **Figure 26**: CI of product 1 = CI of entire multifunctional process – CI of product(s) avoided by product 2).

In an LCA software, avoided products can be computed in the form of input flows with negative value amounts. In the Model, system expansion is used for excess electricity and steam produced at fuel production facility, where the produced electricity and steam alongside a product can be used to heat other processes within another product system. It is also applied when some waste material is used as feedstock for LCIF production and results in real methane reductions. In this case, the system boundary around the waste material for fuel production is expanded to include the emission differential between using the waste material for fuel production and a baseline scenario that would have occurred if the waste material were not used for fuel production[4]. With system expansion, the emissions associated with the avoided electricity and steam production or the baseline scenario for waste management are subtracted from the CI value of the fuel.

Causal allocation

Causal allocation is a term used in openLCA to represent a custom allocation approach that can be used to model specific allocation scenarios, including the energy and mass-based allocations. Allocation factors are entered manually for each flow.

---

[4] Refer to the Fuel LCA Model Methodology for more information on the methodology, data sources, and modelling assumptions for allocation waste materials used as feedstock.

Appendix A: Example of using the *Fuel LCA Model* to calculate a CI

This appendix presents a hypothetical example of using the Model to calculate a carbon intensity. The example goes through how to use the Data Library and fuel pathways to complete a product system for a specific scenario, but the general modelling concepts are applicable to a wide variety of scenarios and situations.

A.1 Example: Bioethanol produced from corn

The example, as described in **Figure 27**, models the production of bioethanol produced from corn following the life cycle stages defined in **Chapter 0**. In this example corn is produced and then transported to the bioethanol plant. Then, energy and chemical inputs of natural gas and hydrogen respectively, alongside the corn, are used to produce the bioethanol. This generates process emissions and produces an animal feed coproduct. Finally, the produced bioethanol is distributed to end users, where it is combusted in an internal combustion engine. The quantities associated with the example are shown in **Figure 27**.
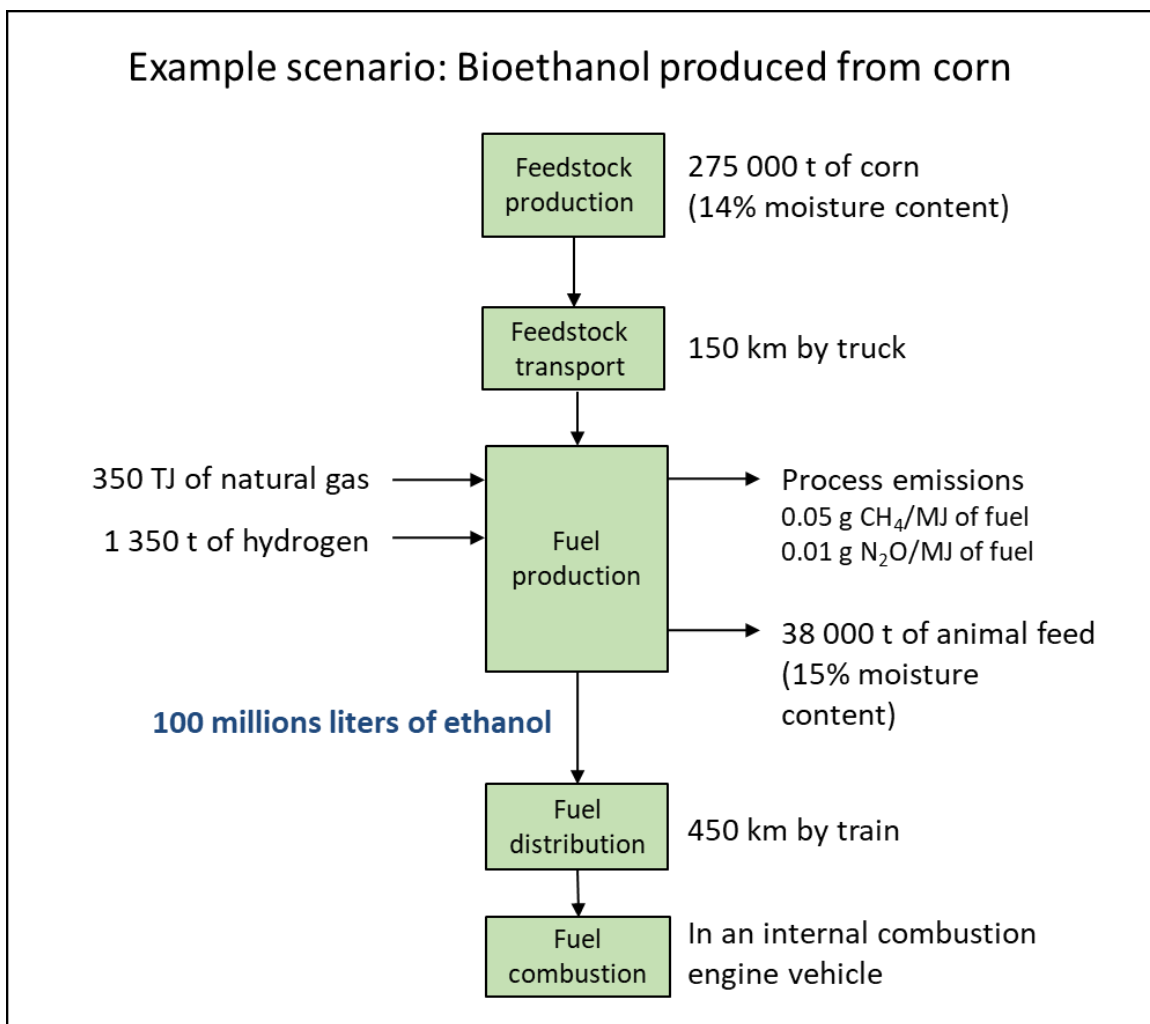

Figure 27: Product system of bioethanol being produced from corn

Figure 28 shows how the system will be modelled in openLCA. The example will walk through each of the life cycle stages, from feedstock production to fuel combustion, as they are modelled with the Model. It will also show how to use each unit process within the pathway. As explained in **Chapter 4.2.2**, the fuel pathways in the Model Database are not structured in the same way as what would occur in reality. This structure enables simpler modelling of a highly customizable systems that can be used for a variety of scenarios.
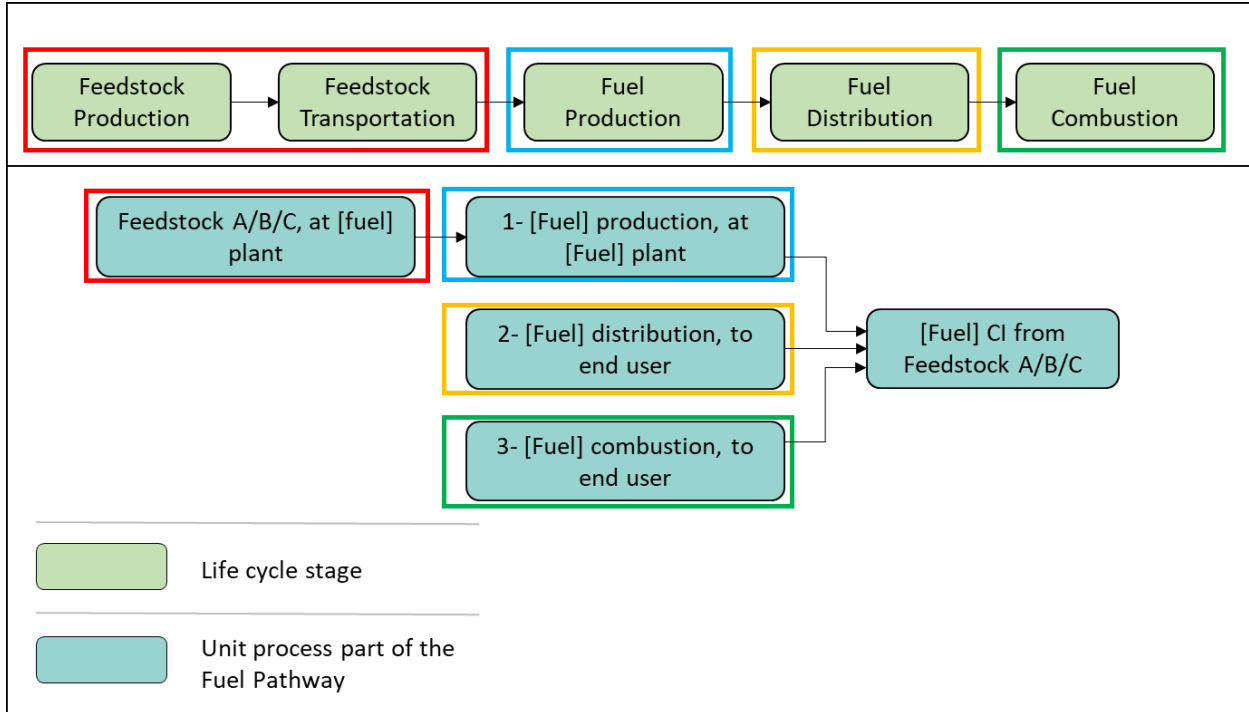
Figure 28: Comparison of life cycle stages with unit processes in openLCA

Figure 29 shows a screenshot of the model graph of the completed pathway in openLCA (see **Chapter 5.3.1**). The model graph shows the processes and linkages involved in modelling the bioethanol product system. Note that the example is theoretical and different systems may use different building blocks from the Data Library. System processes from the Data Library will be used alongside the data provided in the example to model the bioethanol life cycle.

Figure 29: Model graph of the bioethanol life cycle example

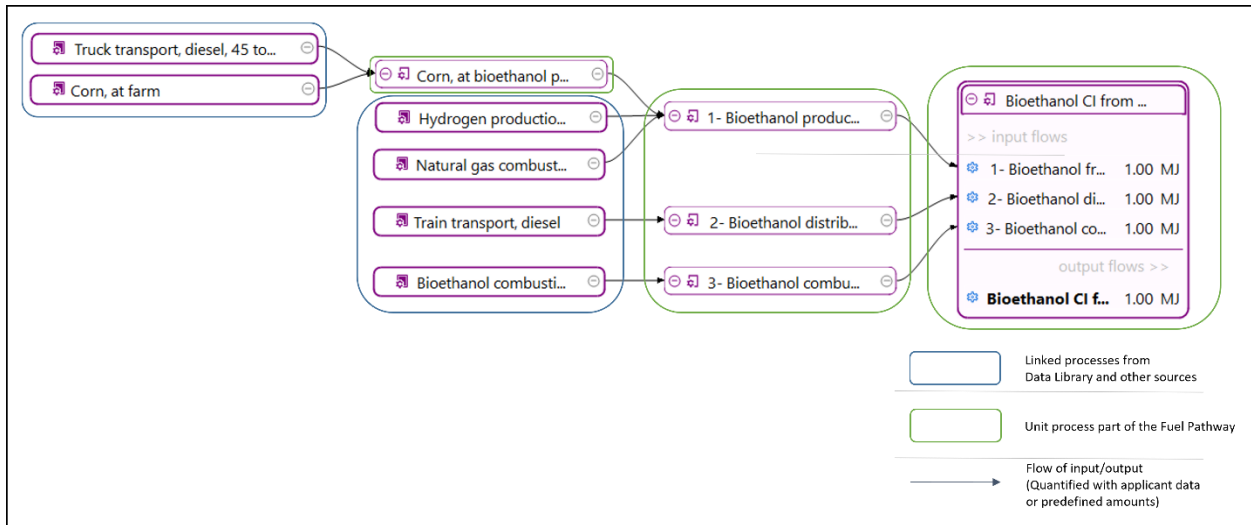The following sections show the steps to model each of the life cycle stages in openLCA and how to calculate the CI of the completed system.

A.1.1 Feedstock production and transport

The information needed to model the feedstock production and transport is shown in **Figure 30**. This figure shows how corn production and transportation are modelled in the first unit process of the bioethanol fuel pathway, including necessary calculations and conversions.

$$\text{Transport amount (kg. km)} = \text{distance (km) x } \frac{1 \text{ kg}_{dry-basis}}{1 - \text{Moisture content}_{feedstock}}$$
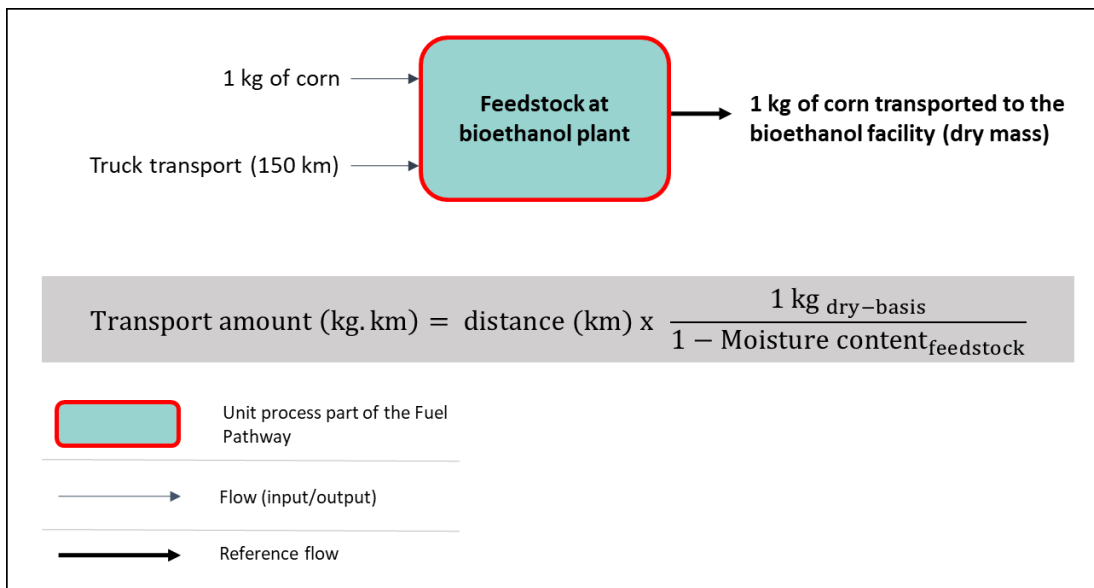
Figure 30: Corn production and transportation modelling

To model the feedstock production and transport, follow these steps:

a) In the Processes folder of the database, go to the folder Fuel Pathways/Bioethanol pathway/Feedstock at bioethanol plant, and open the process "Feedstock A, at bioethanol plant" by double-clicking it. As a reminder, the processes are found in the navigation pane on the left side of the screen (**Chapter 4.1**).

b) Go to the Inputs/Outputs tab of the unit process (**Chapter 4.2.3** shows the layout of a process in openLCA).

- As you notice, the unit processes in the fuel pathways are often empty. These unit processes are templates and must be filled with user data to calculate a custom CI.

With the "Feedstock A, at bioethanol plant" process, corn and truck transportation are needed as inputs, as shown above. These inputs will be taken from the Data Library. To add the processes as inputs, follow these steps:

c) Click-and-drag the processes (called Corn, at farm and Truck transport, diesel, 45 tonnes) from the navigation pane and into the Inputs Table of the process (see **Chapter 5.2.3**).

Figure 31 shows what the Inputs Table should look like after dragging the processes into the "Feedstock A, at bioethanol plant process".

Figure 31: Screenshot of the Inputs Table of the "Feedstock A, at bioethanol plant" process after dragging the processes

The next step is to determine the quantity and units to enter in the unit process. As explained in **Chapter 6.2.1**, the unit process is an independent mass balance that transforms inputs into outputs. In this process, 1 kg of corn is transported to the bioethanol production facility. Therefore, the input for Corn, at farm is balanced with the output, 1 kg of feedstock A, at bioethanol plant. The functional unit (**Chapter 6.2.5**) of the unit process is on a dry basis, as is the corn flow.[5] However, the corn is transported to the bioethanol production facility on a wet basis. Therefore, the transportation flow

---

[5] The information about the functional unit for the corn process can be found in its **metadata** in the Data Library. Open the system process for Corn, at farm to view the metadata. The contents of the metadata in the Data Library are explained in **Chapter 4.5.1**.

needs to account for the entire mass of corn being transported, and not just the dry mass. An example of this conversion is as follows:

$$\text{transportation amount (kg*km)} = \text{distance (km)} * (1\ kg_{dry\ basis})/(1\text{-moisture content}_{feedstock})$$

$$\text{transportation amount} = 150\ km * (1\ kg_{dry\ basis})/(1\text{-}0.14)$$

$$\text{transportation amount} = 174.4186\ kg*km$$

Follow these steps to add the transportation amount and finish modelling the unit process:

d) Enter the transportation amount into the "amount" column in the Inputs Table either as a formula or as the calculated value (see **Chapter 5.2.2**).

e) After the amount is entered, select the unit of "kg*km" by opening the drop-down menu for the unit column.

f) Rename it to represent the customization that has been made. Rename the unit process by going to the General information tab (see **Chapter 5.2.6**).

g) Rename the reference flow to match the new unit process name.

h) Save the unit process.

Figure 32 shows a screenshot of the unit process with the updated values and names.

Figure 32: Screenshot showing the updated values and names of the original "Feedstock A, at bioethanol plant" process

Now that the feedstock and transportation life cycle stages have been modelled in openLCA, the bioethanol production life cycle stage can be modelled.

A.1.2 Fuel production

The details of the example used to model the fuel production life cycle stage are shown in **Figure 33** below. This diagram shows how bioethanol production is modelled in the second unit process of the bioethanol fuel pathway, including necessary calculations and conversions.
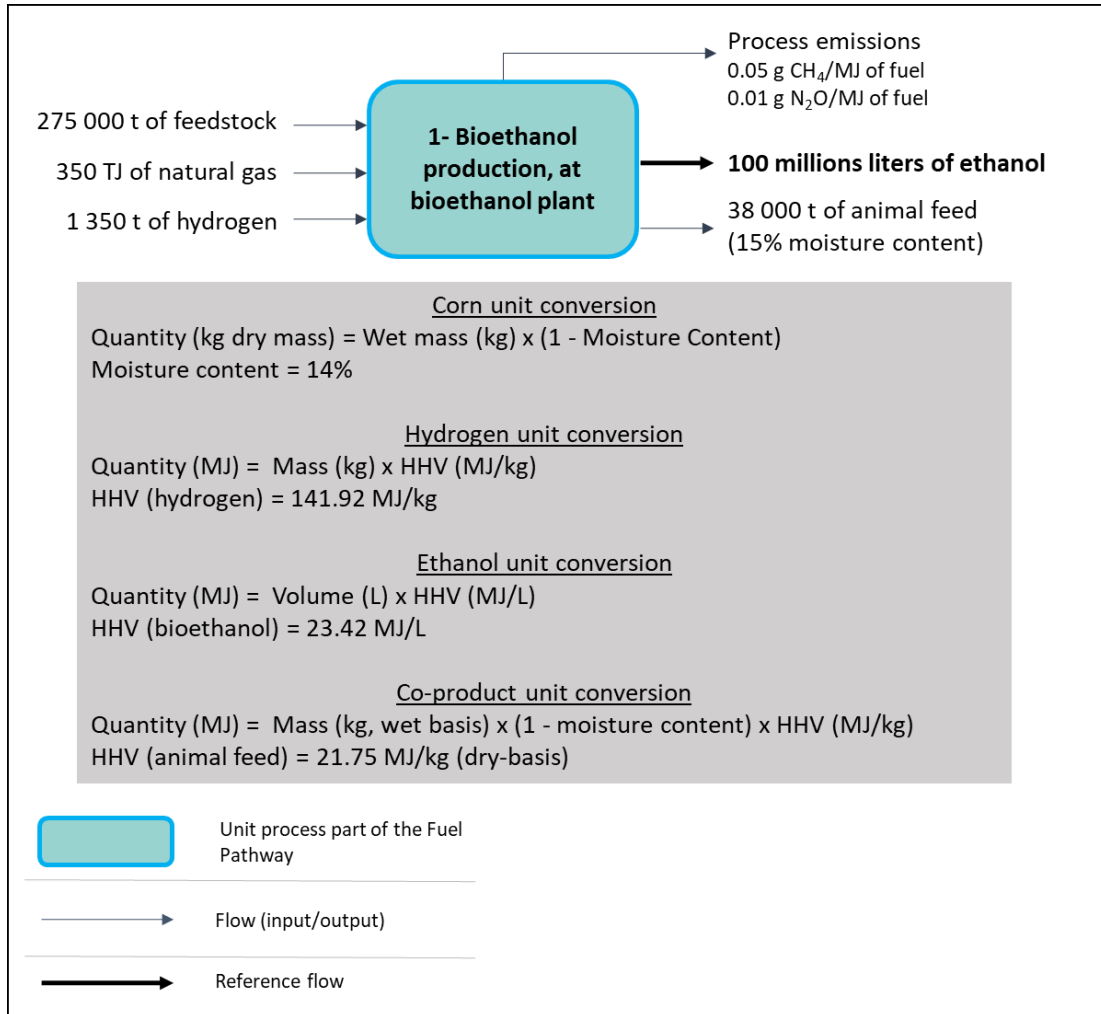

Figure 33: Bioethanol production modelling

The transported corn is used alongside natural gas and hydrogen to produce bioethanol and a coproduct. Like the feedstock production unit process, this process is empty and can be filled in with building blocks from the Data Library. In addition, the process will be linked to the previous life cycle stage (the feedstock production and transportation processes from the previous section).

Corn input

First, follow the step below to link the two life cycle stages:

   a) Open the "1- Bioethanol production, at bioethanol plant" unit process in the fuel pathways section of the database.

b) Click-and-drag the feedstock process from the navigation pane and into the Inputs Table of the bioethanol production process. (This is the same way as adding the inputs from the previous section.)

For the quantity of corn, there are 275 000 t being used to produce the 100 million litres of ethanol. This mass is a wet basis mass, so it must be converted to dry mass to properly represent the functional unit of the corn process. The conversion is as follows:

$$\text{Quantity (kg}_{dry\ basis}) = \text{mass (kg}_{wet\ basis})*(1\text{-moisture content})$$

Quantity = 275000 t*(1000 kg/1 t)*(1-0.14)

Quantity = 2.365E8 kg

c) Enter the calculated quantity or the formula into openLCA. Since the quantity has been converted to kg, there is no need to change the unit for the input. Also note that openLCA allows the use of scientific notation.

Unit processes are **independent** mass/energy balances. So while the quantities within a unit process must be balanced, this means that the inputs and outputs of the unit process do not rely on the quantity of inputs and outputs of other unit processes.

Hydrogen input

In this example scenario, hydrogen is used as a chemical input in the production of bioethanol. Therefore, it can be found in the chemical inputs folder of the Data Library.

a) Click-and-drag the hydrogen process into the Inputs Table. As explained in **Chapter 5.2.3**, you can also search for flows in the database using the green plus button at the top-right of the Inputs Table.

After adding the input you will realize that the flow property (i.e. the types of units, **Chapter 4.5.2**) is in units of energy. Since we have the mass of hydrogen used, we must convert the amount of hydrogen using hydrogen's HHV. The conversion is shown below.

$$\text{Quantity (MJ)} = \text{mass*HHV(MJ/kg)}$$

Quantity (MJ) = 1350 t*(1000 kg/1 t)*141.92 MJ/kg

Quantity = 1.91592E8 MJ

b) Enter either the formula or the calculated value into the Inputs Table. Keep the unit as the default MJ.

Natural gas input

The natural gas input can be found in the fossil fuels folder of the Data Library.

a) Since natural gas is being burned, use the "Natural gas combustion" process as the input.

Once added into the Inputs Table, you will notice that the flow is in MJ. Since our quantity of natural gas is already reported in units of energy, there are only two things to do:

b) Select TJ from the unit field.

c) Enter the amount of natural gas used (350 TJ) directly into the amount field.

Ethanol output

The ethanol output represents the functional unit of the process (the **bolded** flow in the output table). All inputs and other outputs must be balanced to this quantity. Since the functional unit here is in MJ, a conversion must be performed. The conversion, which uses the HHV of ethanol, is shown below.

$$\text{Quantity (MJ)} = \text{volume (L)} * \text{HHV (MJ/L)}$$

$$\text{Quantity (MJ)} = 100E6 \text{ L} * 23.42 \text{ MJ/L}$$

$$\text{Quantity} = 2.342E9 \text{ MJ}$$

a) Enter the quantity or formula into openLCA. In this case it may be better to leave the value as a formula because this formula can be copied and pasted for the upcoming section, Process emissions output.
b) Open the bioethanol flow (double-click the flow from the output table) and rename the flow to "1- Bioethanol from corn".

Note that the process has other outputs called "Bioethanol from Feedstock B" and "Bioethanol from Feedstock C". These are available to use if there is more than one feedstock, but since this is out of scope for this example the two flows can stay at 0.

Animal feed coproduct output

The production of bioethanol also results in a coproduct that can be used as animal feed. Unlike the inputs, which come from other **processes**, the outputs are **flows** and are therefore found in the flows folder of the database (see **Chapter 6.2.1** for a description of processes and flows, and **Chapter 4.3.2** for the organization of the intermediate flows). The "Fuel pathways" section of the "Flows" folder contains common coproducts for each fuel pathway. In this case, follow the step below to add the coproduct:

a) Add the "Animal feed from Feedstock A" flow from the Flows folder to use as the output. Click-and-drag the flow into the outputs table.

The coproduct is reported in mass, but must modelled in energy. The HHV and the moisture content of the animal feed are used to convert the amount to MJ as shown below.

$$\text{Quantity (MJ)} = \text{mass (kg}_{\text{wet basis}}) * (1\text{-moisture content}) * \text{HHV (MJ/kg)}$$

$$\text{Quantity (MJ)} = 38000 \text{ t} * (1000 \text{ kg/1 t}) * (1\text{-}0.15) * 21.75 \text{ MJ/kg}$$

$$\text{Quantity} = 7.02525E8 \text{ MJ}$$

b) Enter the quantity or formula into openLCA. The unit is already in MJ.
c) Open the animal feed flow (double-click the flow from the output table) and rename the flow to "Animal feed from Corn".

Process emissions output

The final flows to consider are those for the process emissions. Note that these process emissions only correspond to emissions that directly result in the chemical production of bioethanol. Emissions from other sources, such as through the combustion of natural gas, have already been accounted for in their

respective processes. The flows themselves are elementary flows, i.e. GHGs. They can be found in the "Elementary flows" section of the "Flows" folder in the database (see **Chapter 4.3.1**).

a) Click-and-drag the "Methane (CH4), biogenic" and "Nitrous oxide (N2O)" flows from the "Elementary flows" section of the database into the outputs table of the unit process.

Like the inputs and other outputs, the process emissions must be balanced with the amount of ethanol being produced, which is the reference flow of the process. The process emissions have been reported in g/MJ. Therefore, we need to convert the values to grams per [amount of MJ of ethanol being produced]. To do so, simply copy the formula of energy (MJ) we previously calculated (in step (a) in the section Ethanol output) and apply it to the amount of emissions being released. This is shown below for each elementary flow.

$$\text{Quantity } CH_4 \text{ (g) = amount released (g/MJ)*amount of ethanol produced (MJ)}$$

$$\text{Quantity } CH_4 \text{ (g) = 0.05 g/MJ*100E6 L*23.42 MJ/L}$$

$$\text{Quantity } CH_4 = 1.171E8 \text{ g}$$

$$\text{Quantity } N_2O \text{ (g) = amount released (g/MJ)*amount of ethanol produced (MJ)}$$

$$\text{Quantity } N_2O \text{ (g) = 0.01 g/MJ*100E6 L*23.42 MJ/L}$$

$$\text{Quantity } N_2O = 2.342E7 \text{ g}$$

b) Enter the results in openLCA as formulas or quantities. For each flow, change the unit from "kg" to "g".
c) Save the unit process.

Figure 34 presents a screenshot of the process including all of the inputs and outputs.

Figure 34: Screenshot of the "1-Bioethanol production, at bioethanol plant" process including all of the inputs and outputs

Allocation of bioethanol production

Before moving on to the next life cycle stage, allocation needs to be performed on the outputs of the unit process. Follow the steps below to perform allocation on the process.

a) Before doing allocation, ensure all values are correctly input and save any changes.
b) Go to the allocation tab of unit process. Select "Physical allocation" (see method 2 in **Chapter 5.2.8**) and then click "Calculate factors".
c) Save the process.

A screenshot of the completed physical allocation table is shown in **Figure 35**.
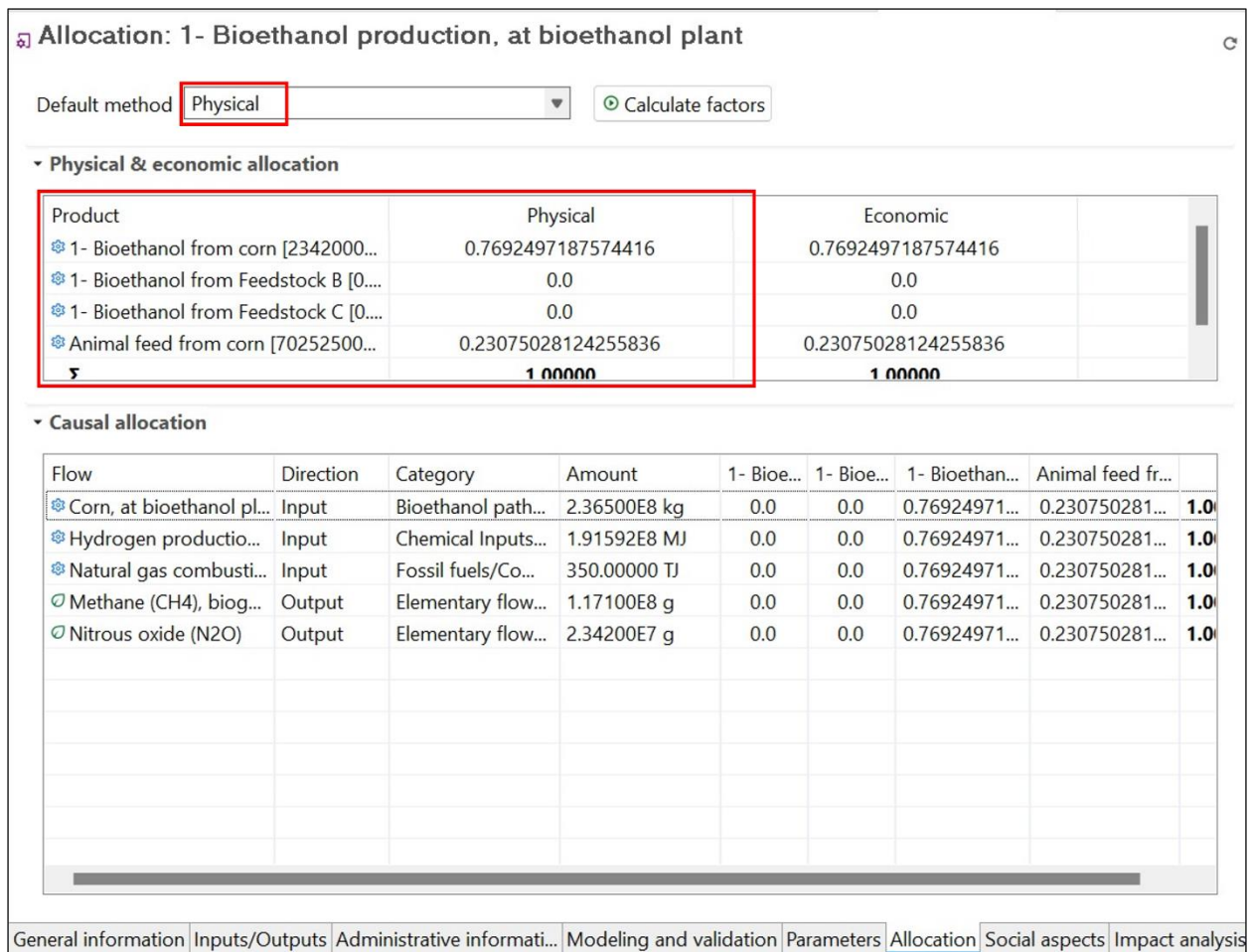
Figure 35: Screenshot of the completed physical allocation table

A.1.3 Bioethanol distribution

After the bioethanol has been produced, it needs to be transported to the end user. This life cycle stage can be modelled using a single unit process, called "2- Bioethanol distribution, to end-user". The bioethanol is transported via train, so the "Train transport, diesel" system process from the Data Library can be dragged into the Inputs Table. This is visualized in **Figure 36**. This figure shows how bioethanol distribution is modelled in the third unit process of the bioethanol fuel pathway, including necessary calculations and conversions.
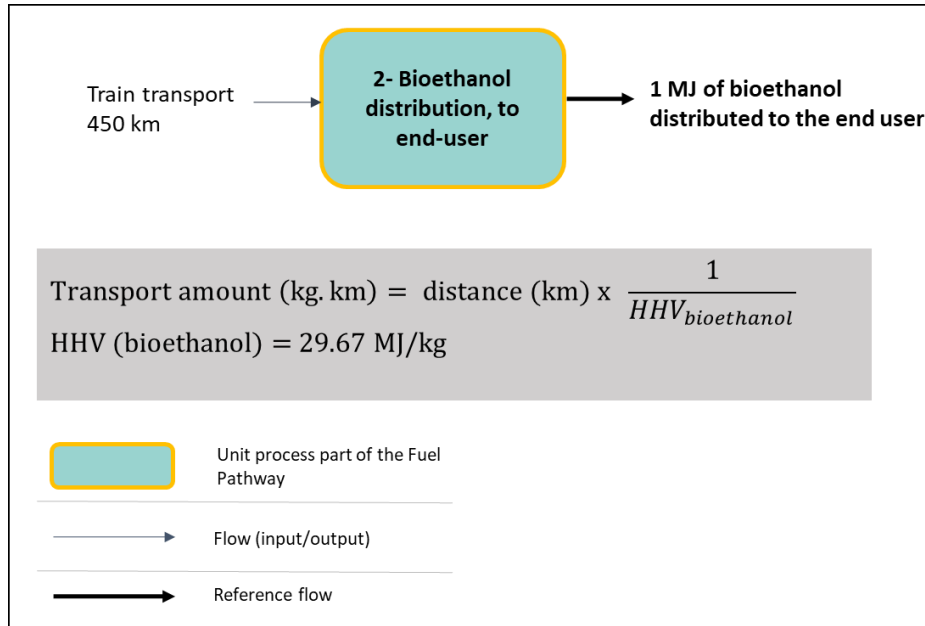
Figure 36: Bioethanol distribution modelling

Follow the steps below:

a) Open the "2- Bioethanol distribution, to end-user" process.
b) Add the "Train transport, diesel" system process from the Data Library into the Inputs Table.

Once the train transport flow has been added to the Inputs Table, its amount and unit can be determined. Just like with feedstock transportation, the distribution distance needs to be entered in units of mass*distance. Note that the functional unit of the process is 1 MJ bioethanol being distributed. Therefore, we can use the HHV of bioethanol and a basis of 1 MJ to convert our amount of ethanol to kg. The calculation is shown below.

$$\text{Quantity (kg*km)} = \text{mass (kg)*distance (km)}$$

$$\text{Quantity (kg*km)} = (1 \text{ MJ})*(1 \text{ kg})/(29.67 \text{ MJ})*450 \text{ km}$$

$$\text{Quantity} = 15.167 \text{ kg*km}$$

c) Enter the value or formula in openLCA. Select the unit "kg*km".
d) Save the process.

A screenshot of the completed distribution process is shown in **Figure 37**.

Figure 37: Screenshot of the completed distribution process

A.1.4 Bioethanol combustion

The final life cycle stage to model is fuel combustion. A visualization of the combustion stage is shown in **Figure 38**. This figure shows how bioethanol combustion is modelled in the fourth unit process of the bioethanol fuel pathway, including necessary calculations and conversions.
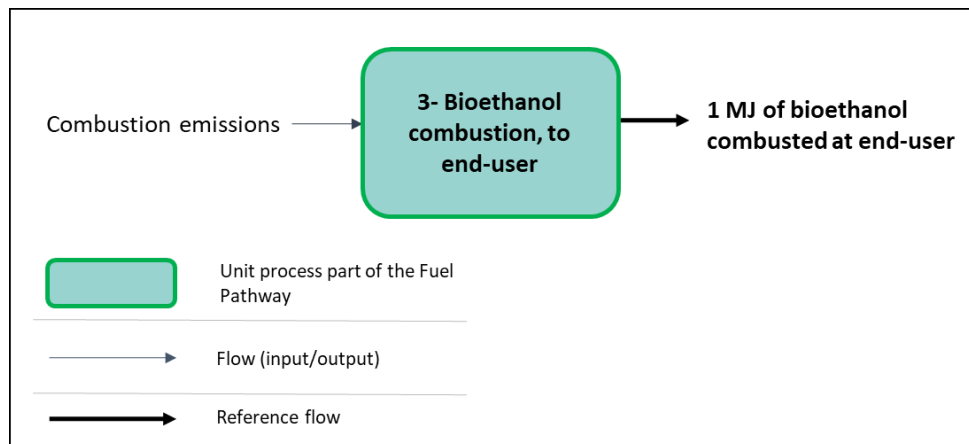

Figure 38: Bioethanol combustion modelling

To model combustion emissions, we can again use the Data Library for the input. Bioethanol combustion emissions are included in the "Combustion emission factors" section of the Data Library. Note that while the rest of the Data Library contains system processes that model the life cycle emissions of certain activities, the processes in the combustion emission factors **only** model the emissions for combustion. This avoids double counting and allows us to use the process to model combustion emissions for the fuel pathways.

a) Open the "3- Bioethanol combustion, at end-user" unit process.
b) Click-and-drag the "Bioethanol combustion" process into the Inputs Table.
c) Save the process.

No other calculations are needed here as the unit process is balanced. A screenshot of the completed process is shown below.

Figure 39: Screenshot of the completed process of combustion

A.1.5 Bioethanol CI calculation

Now that all of the life cycle stages have been modelled, the bioethanol CI can be calculated.

a) Open the "Bioethanol CI from Feedstock A" unit process.
b) Change the names of the unit process and reference flow to "Bioethanol CI from corn" to better represent the system.
c) Follow the steps outlined in **Chapter 5.2.9** to create a product system.
d) Once the product system has been created, go to the "Model graph" tab (see **Chapter 5.3.1**). Here, you will see the exact same model graph as the one presented in **Figure 29** in the beginning of this example.
e) To calculate the CI, go back to the "General information" tab and use the calculate button according to instructions stated in **Chapter 5.2.10.**

The CI calculated is 43.54029 g CO2e/ 1 MJ bioethanol combusted, HHV, viewable in the "Impact analysis" tab of the calculated results. Note that the value of the CI may be slightly different depending on the version of the Model being used. This intensity represents the customized product system that includes all of the life cycle stages modelled in the previous section. A screenshot of the "Impact Analysis" tab is shown in **Figure 40**.
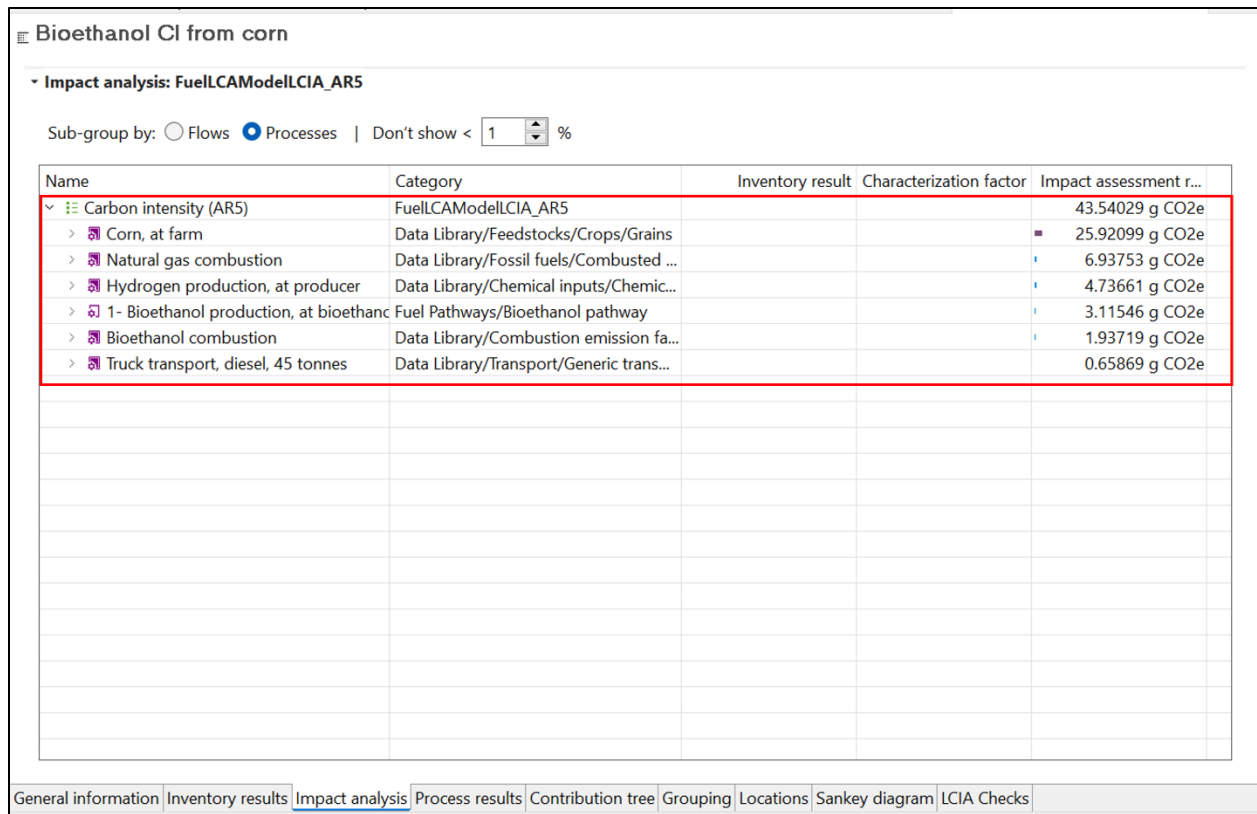
Figure 40: Screenshot of the Impact Analysis tab

Additional openLCA analysis tools are explained in **Chapter 5.3**.

A.2 Additional modelling scenarios

The following examples show how to model certain scenarios when using the Model outside of a specific program.

A.2.1 Inputs from outside the Data Library

While the example only includes information from the Data Library, the Model allows the creation of unit processes to model any activity using other sources of data. This unit process can then be added to a fuel pathway, just like adding anything from the Data Library. For example, if you have data to model a feedstock that is not in the Data Library, follow the steps in **Chapter 5.2.4**, then, follow the principles outlined in the example, such as ensuring a proper mass-energy balance for the new unit process.

When creating a new unit process, just like the fuel pathway processes, any input from the Data Library can be added as an input to model certain activities. Any elementary flows can also be added in the outputs to model process emissions.

Once the new unit process has been created, it can be added as an input to the fuel pathway in the same way as described in the example.

A.2.2 Avoided emissions or waste feedstock processes

A simple way to model avoided emissions is to use a negative elementary flow in the output of a process. Any of the elementary flows can be added to process output as described in **Chapter 5.2.3**.

Similarly, a waste feedstock that results in avoided emissions can be modelled in the same way. A process can be created as described in **Chapter 5.2.4**, and then the elementary flow corresponding to the avoided emissions can be added to the process outputs as a negative value.