

LCA Data Collection & Modeling Guide

For the Replit Development Agent

Version: 1.1
Date: 2025-07-20
Author: Replit Coach Too
Status: Draft

1. Objective & Scope

This document provides a comprehensive technical guide for the Replit Agent to build the data collection and processing engine for conducting a detailed "cradle-to-gate" Life Cycle Assessment (LCA). It uses the provided "Avallen LCA Feb 2021.pdf" as a template.

Scope Clarification: For the MVP, this guide focuses exclusively on the workflow for **drinks brands that manufacture/distill their own products in-house**. The workflow for brands that use contract manufacturers is deferred to a future version.

2. Part 1: Database Schema Expansion

To capture the required level of detail, we must enhance our database schema.

2.1. New Table: lca_questionnaires

This table will store the complete, structured dataset for a single product's LCA, mirroring the stages in the Avallen LCA.

Column Name	Data Type	Constraints	Description
id	UUID	PRIMARY KEY	Unique identifier for the questionnaire submission.
product_id	UUID	NOT NULL, FOREIGN KEY (products.id)	Links to the product SKU this data is for.
reporting_period	DATERANGE	NOT NULL	The reporting period this data applies to.
lca_data	JSONB	NOT NULL	Stores the entire structured questionnaire data. See structure below.

status	VARCHAR(50)	NOT NULL, DEFAULT 'incomplete'	e.g., 'incomplete', 'complete', 'processing', 'calculated'.
--------	-------------	-----------------------------------	--

Structure for lca_data JSONB field:

This JSON object will be structured to hold all the necessary data points, organized by life cycle stage.

```
{
  "agriculture": {
    "main_crop_type": "Apples",
    "yield_ton_per_hectare": 31.5,
    "diesel_l_per_hectare": 15,
    "sequestration_ton_co2_per_ton_crop": 0.40
  },
  "inbound_transport": {
    "distance_km": 10,
    "mode": "Lorry >32t"
  },
  "processing": {
    "water_m3_per_ton_crop": 0.056,
    "electricity_kwh_per_ton_crop": 10.21,
    "juice_l_per_ton_crop": 650,
    "pulp_kg_per_ton_crop": 300,
    "cider_l_per_l_spirit": 13,
    "lpg_kg_per_l_alcohol": 0.16,
    "net_water_use_l_per_bottle": 1.23,
    "spirit_yield_l_per_ton_crop": 50,
    "angels_share_percentage": 2
  },
  "packaging": [
    {
      "component": "Bottle",
      "material": "Glass, white",
      "weight_grams": 540
    },
    {
      "component": "Label",
      "material": "Paper, from apple pulp",

```

```

    "weight_grams": 0.824
  },
  {
    "component": "Stopper",
    "material": "Cork, natural",
    "weight_grams": 2.6
  },
  {
    "component": "Secondary Packaging",
    "material": "Cardboard, fluting medium",
    "weight_grams": 38.16
  }
]
}

```

3. Part 2: Expanded User Onboarding Flow (For Producers)

This section details the in-depth data collection interface for brands that produce their own products. This will replace the simplified "Level 3: Product Footprint" in our App Flow Doc.

Interface: A multi-tabbed or multi-page form within the onboarding wizard.

Tab 1: Agriculture

- **Headline:** "Tell us about your core ingredient."
- **Fields:**
 - main_crop_type: (Dropdown: Apples, Grapes, Barley, Maize, Sugar Cane, etc.)
 - yield_ton_per_hectare: (Numeric Input)
 - diesel_l_per_hectare: (Numeric Input, for harvesting machinery)
 - sequestration_ton_co2_per_ton_crop: (Optional Numeric Input, with a tooltip explaining this is an advanced metric and can be left blank).

Tab 2: Inbound Transport

- **Headline:** "How do your ingredients get to your facility?"
- **Fields:**
 - distance_km: (Numeric Input: "Average distance from farm/source to your facility")
 - mode: (Dropdown: Lorry >32t, Van, Rail, etc.)

Tab 3: Processing

- **Headline:** "Let's look at your production process."
- **Fields (Dynamic based on product_type from Level 1):**
 - **For all types:**
 - net_water_use_l_per_bottle: (Numeric Input: "Net water usage in the facility per final bottle")
 - **For Spirits/Cider:**
 - water_m3_per_ton_crop: (Numeric Input: "Water for washing")
 - electricity_kwh_per_ton_crop: (Numeric Input: "Electricity for crushing/pressing")
 - cider_l_per_l_spirit: (Numeric Input: "Cider/Wash required to produce 1L of spirit")
 - lpg_kg_per_l_alcohol: (Numeric Input: "LPG or other fuel for distillation")
 - **For Aged Spirits:**
 - angels_share_percentage: (Numeric Input: "Annual evaporation loss during aging")

Tab 4: Packaging

- **Headline:** "Detail every piece of your product's packaging."
- **Interface:** A repeatable list component. The user clicks "Add Component" to add a new piece of packaging.
- **Fields for each component:**
 - component: (Dropdown: Bottle, Can, Label, Stopper, Secondary Packaging, etc.)
 - material: (A searchable dropdown populated with common materials from the ecoinvent database, e.g., "Glass, white", "Paper, kraft", "Aluminum").
 - weight_grams: (Numeric Input)

4. Part 3: Backend Modeling Instructions for Replit Agent

This section provides the detailed logic for the Celery worker that will process the lca_questionnaires data for a self-producing brand.

1. **Initiate Worker:** The worker is triggered when a user completes the questionnaire and the lca_questionnaires.status is set to complete.
2. **Fetch Data:** The worker retrieves the lca_data JSON object from the database.
3. **Create Product System in OpenLCA:** Create a new product system named after the product SKU. The reference process will be the final "Packaging" step, with a target amount of 1 bottle.
4. **Model the Process Chain (in reverse order):**
 - **Packaging Process:**

- Create a "Packaging" process.
- For each item in the packaging array, find the corresponding material flow in ecoinvent (e.g., market for packaging glass, white) and add it as an input to this process with the specified weight_grams.
- **Processing Process:**
 - Create a "Processing" process (e.g., "Distillation").
 - Add energy and water flows as inputs based on the processing data.
 - The primary output of this process (the liquid) will be an input to the "Packaging" process.
- **Transport Process:**
 - Create a "Transport" process using the inbound_transport data.
 - The output of this process is an input to the "Processing" process.
- **Agriculture Process:**
 - Create an "Agriculture" process (e.g., "Apple growing").
 - Add diesel and other inputs from the agriculture data.
 - If the user provided a sequestration value, add a negative CO2 emission flow to this process.
 - The output of this process (the crop) is an input to the "Transport" process.
- 5. **Connect the Chain:** Use the OpenLCA API to link the output of each process as an input to the next, creating a complete life cycle chain from farm to finished product.
- 6. **Calculate & Store Results:**
 - Run the calculation on the final product system for the required impact methods (GWP100, Water Footprint).
 - Parse the results.
 - Save the final numbers into the reports table in the report_data_json field.
 - Update the reports.status to draft and notify the user.