version 1.3

LCA Collaboration Server

openLCA

LCA Collaboration Server v.1.3

User Manual and Introduction

Version: 1.0.1

Authors: Julia Cilleruelo Palomero, Sebastian Greve, Andreas Ciroth

Date: 08 February 2022

GreenDelta GmbH

Kaiserdamm 13
14057 Berlin, Germany
www.greendelta.com

Tel. +49 30 48 496 – 030
Fax +49 30 48 496 – 991
gd@greendelta.com

GreenDelta openLCA

Contents

List of Figures

Glossary

Commit - A *commit* refers to contributing a package of data sets from a user's local database in openLCA to a repository on the LCA Collaboration Server.

Fetch - To *fetch* refers to updating a user's local database in openLCA with changes from a repository on the LCA Collaboration Server.

Repository - A *repository* it the equivalent to a database in openLCA and consists of grouped data sets (projects, product systems, processes, flows, indicators and parameters and background data). A *repository* mirrors the local databases of the working group's users.

Group - A *group* contains one or more repositories. *Users* and *teams* can be added to a *group* to gain access to the repositories that are assigned to a specific *group*.

Team - A *team* is an aggregation of users. *Teams* can be added to *groups* to gain access to one or more specific repositories.

Checkout - A *checkout* is the recovery of a previous commit from a repository. During a *checkout*, the local database in openLCA is replaced with a selected version of the repository from the LCA Collaboration Server.

Abbreviations

GLAD - *Global LCA Data Access network*

LCA - *Life Cycle Assessment*

LCA CS - *LCA Collaboration Server*

LCT - *Life Cycle Thinking*

PEF - *Product Environmental Footprint*

USDA - *United States Department of Agriculture*

Has your team ever faced limitations in conducting an LCA study because only one user can work on an LCA model at a time?

The LCA Collaboration Server facilitates next-level collaborative Life Cycle Assessment modelling!

1. Introduction

With the general advancement of Life Cycle Assessment (LCA) and the spreading of Life Cycle Thinking (LCT), collaborative work on LCA studies becomes increasingly common e.g. within a company or team at the same location, within a company or team at different locations within multilateral projects (e.g. in research) on an international level and under participation of various entities such as companies, universities or consultancies as well as co-development of an LCA among executing contractors and clients. Moreover, it is becoming increasingly commonplace to use reference data for background processes, elementary flows, impact assessment methods, and other elements of existing LCA models. Often sharing of reference data for updating databases and distribution to users who are (also) distributed comes along with technical issues. In addition, quality assurance and review of LCA models through an external reviewer is of interest for anybody who wants to create a consistent LCA database.

The LCA Collaboration Server, now in version 1.3, is a server application that complements openLCA (the LCA desktop application). It is available for free, and support is available on demand.

Since version 1.7, openLCA accommodates full integration of the LCA Collaboration Server and facilitates the development of LCA studies in distributed teams. The LCA Collaboration Server is not only a milestone for openLCA but offers innovative features and opportunities for LCA practitioners that are unparalleled by any other LCA software.[1] openLCA users are now able to work simultaneously on an LCA study while tracking each other's changes along synchronised databases.

1.1 How can the LCA Collaboration Server be for free?

The LCA Collaboration Server was developed by GreenDelta with support from the United States Department of Agriculture (USDA), creator of the LCA Digital Commons[2]. Without support from the USDA, GreenDelta would not be able to offer the LCA Collaboration Server for free and GreenDelta regards this is a great chance to return a favour to the LCA community and support Life Cycle Thinking (LCT). Check-out conclusion and outlook on section 5 to see what we have in mind.

1.2 What are use cases of the LCA Collaboration Server?

The following are some of the use cases covered by the LCA Collaboration Server:

1) A team working on the same database/LCA model, and wanting to synchronise their data.

2) A single LCA practitioner working on a database/LCA model from different devices, and wanting to synchronise their data.

[1] Software solutions that offer a functionality close to that of the LCA Collaboration Server may be *SimaPro's* multi-user feature from *PRé Sustainability* and *Soda4LCA* from the *Karlsruhe Institute of Technology*

[2] https://www.lcacommons.gov/

3) A personal documentation backup system with change control, where one can comment blocks of changes to document your work.

4) Straightforward sharing of data sets, LCA models and entire databases and their updates without losing a database's integrity and enabling the track of minor changes.

5) Publication of data, where others can view or download publicly available data on the LCA Collaboration Server and import them directly into openLCA without registration.

6) Creation of consistent and verified databases with the option to check previous version.

2. First steps

The LCA Collaboration Server (LCA CS) is an open-source tool available for free. For that reason, it can be downloaded, installed, and maintained by the user alone, see section 2.1. However, not all users have the IT infrastructure to host their own LCA Collaboration Server or do not want to bother with its installation and operation. For that case, GreenDelta also offers hosting solutions and support, see section 2.2.

This section also explains how to make the link between the openLCA in your machine and the LCA CS, the dashboard of the LCA CS, and the types of users and roles available for this tool.

2.1 Install the LCA Collaboration Server on your server

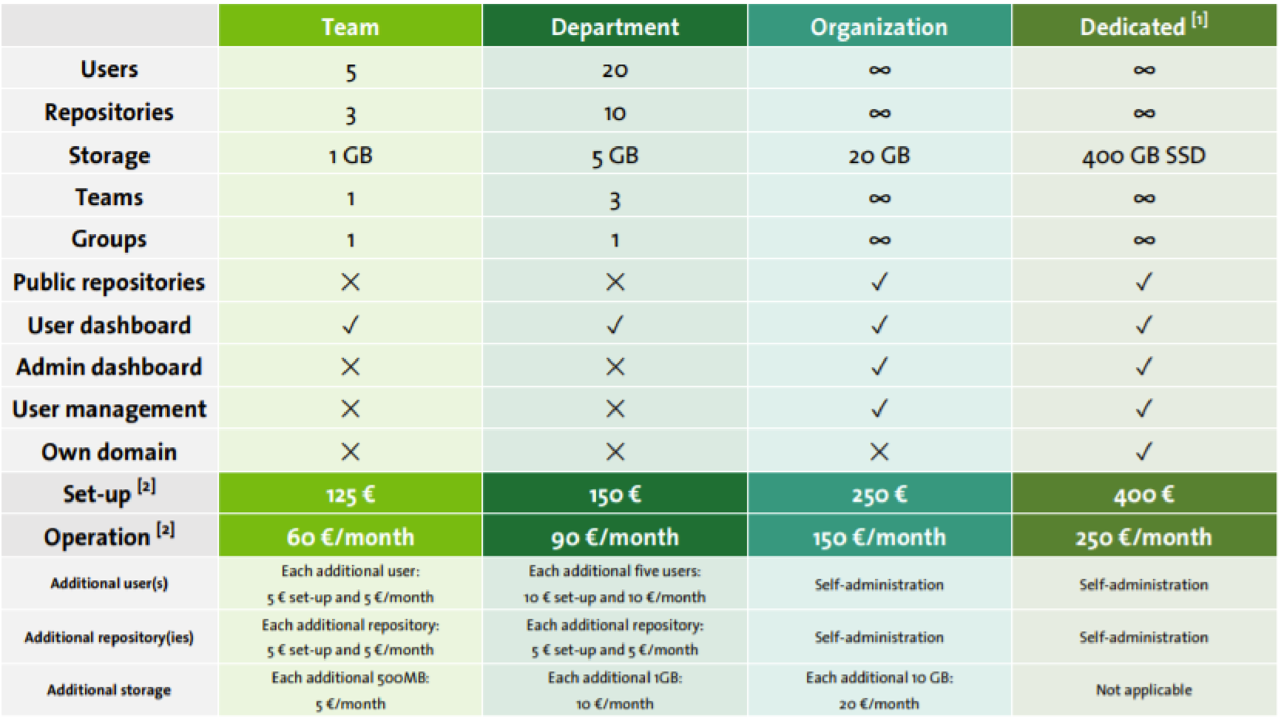
With this option the user can use the LCA CS for free, and has to install and maintain it themselves. Here are the steps:

1. Download the LCA Collaboration Server from https://www.openlca.org/download/
2. Install a local instance of the LCA Collaboration Server on your server
    - Hardware requirements for the LCA Collaboration Server (p.36 in the Appendix)
    - Installation - follow: https://www.openlca.org/lca-collaboration-server-1-3-0-installation-guide

2.2 Hosting and further support

If the user wants to start using the tool directly without having to instal or maintain it, GreenDelta can host the user's server. The hosting plans are described in Figure *2-1*, and it can be purchased in openLCA Nexus: https://nexus.openlca.org/service/LCA%20Collaboration%20Server%20Hosting.

For users and administrators of the LCA Collaboration Server who wish to receive guaranteed and prioritised professional support in handling their LCA Collaboration Server, GreenDelta offers paid support via the GreenDelta HelpDesk: https://nexus.openlca.org/service/openLCA%20Support%20(help%20desk). Public user-to-user support is available via https://ask.openlca.org/.



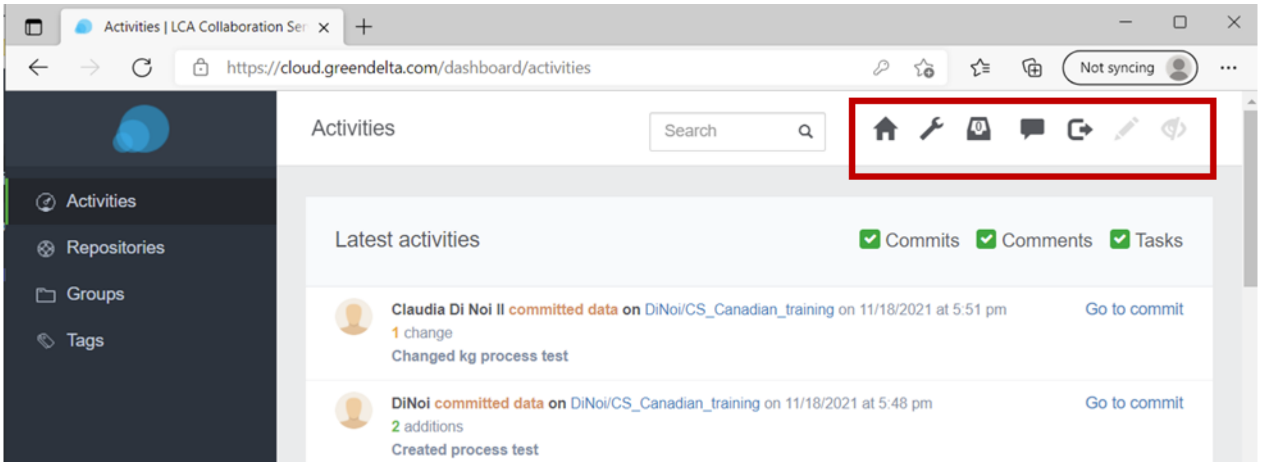
	Team	Department	Organization	Dedicated [1]
Users	5	20	∞	∞
Repositories	3	10	∞	∞
Storage	1 GB	5 GB	20 GB	400 GB SSD
Teams	1	3	∞	∞
Groups	1	1	∞	∞
Public repositories	✗	✗	✓	✓
User dashboard	✓	✓	✓	✓
Admin dashboard	✗	✗	✓	✓
User management	✗	✗	✓	✓
Own domain	✗	✗	✗	✓
Set-up [2]	125 €	150 €	250 €	400 €
Operation [2]	60 €/month	90 €/month	150 €/month	250 €/month
Additional user(s)	Each additional user: 5 € set-up and 5 €/month	Each additional five users: 10 € set-up and 10 €/month	Self-administration	Self-administration
Additional repository(ies)	Each additional repository: 5 € set-up and 5 €/month	Each additional repository: 5 € set-up and 5 €/month	Self-administration	Self-administration
Additional storage	Each additional 500MB: 5 €/month	Each additional 1GB: 10 €/month	Each additional 10 GB: 20 €/month	Not applicable

Figure 2-1 Hosting options offered for the LCA Collaboration Server[3]

2.3 Getting familiar with the LCA Collaboration Server dashboard

Figure 2-2 shows the dashboard of the LCA Collaboration Server, this is the first page after login. The icons in the top right corner give are explained in Table *2-1*.

[3] https://www.openlca.org/wp-content/uploads/2019/11/GreenDelta___Hosting_and_Services.pdf



Claudia Di Noi II committed data on DiNoi/CS_Canadian_training on 11/18/2021 at 5:51 pm — Go to commit
1 change
Changed kg process test

DiNoi committed data on DiNoi/CS_Canadian_training on 11/18/2021 at 5:48 pm — Go to commit
2 additions
Created process test

Figure 2-2 The LCA Collaboration Server dashboard

Table 2-1 Features accessible via the LCA Collaboration Server dashboard

Icon	Feature
🏠	Dashboard, takes user back to home page
🔧	Admin area (creating users, teams, etc.), only for admin users.
0	Task log, brings back to task overview (optional feature)
💬	Messaging, start conversations with your team members
⇥	Log-out
✏	Review mode (*on/off*), enables to add comments on datasets, ready for review
👁	Debugging (*on/off*) , show detailed error stacktraces, only for admin users

To the left of the dashboard we have a menu:

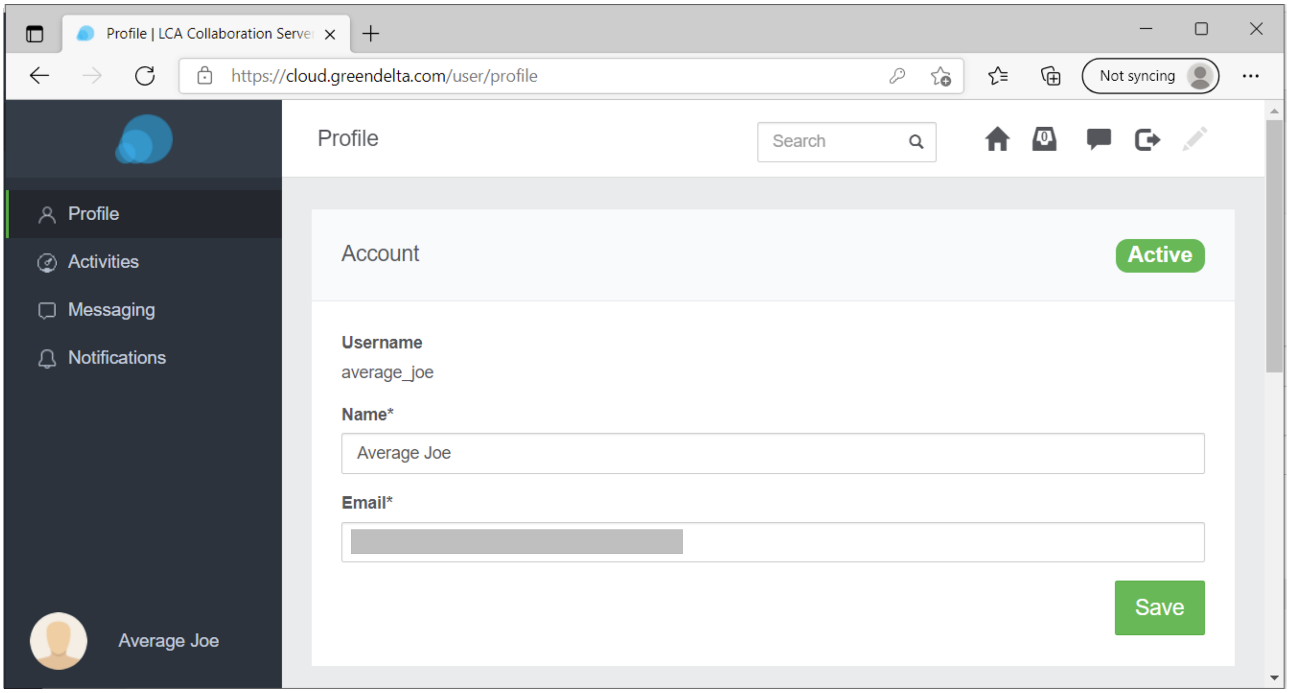
Activities tab: displays the latest activities relating to the repositories the user is involved with. It is a log of the commits, tasks and comments.

Repositories tab: is the central place for the repositories[4] the user is involved with. Repositories can be managed from here, including creating a new repository, or importing one.

Groups tab: holds the groups that the user is part of. A group is made up of a team of users that works on a number of repositories. For example, if a new member joins an LCA team working with 5 repositories in the LCA Collaboration Server, this new user can join the "group" containing those 5 repositories and have access to all of them at once instead of having to be added to each of the repositories individually. Once a group is entered, the user can add *tags*, or add and edit the members, depending on their user role (section 2.3.3).

Tags: this tab brings the user to the tag cloud (experimental feature).

Clicking on the user's name in the bottom left corner of the LCA Collaboration Server dashboard opens the **user dashboard**, see Figure 2-3 . This gives access to account properties, messaging and notification settings. The dashboard may look slightly different for members with different user rights.



Account

Username
average_joe

Name*
Average Joe

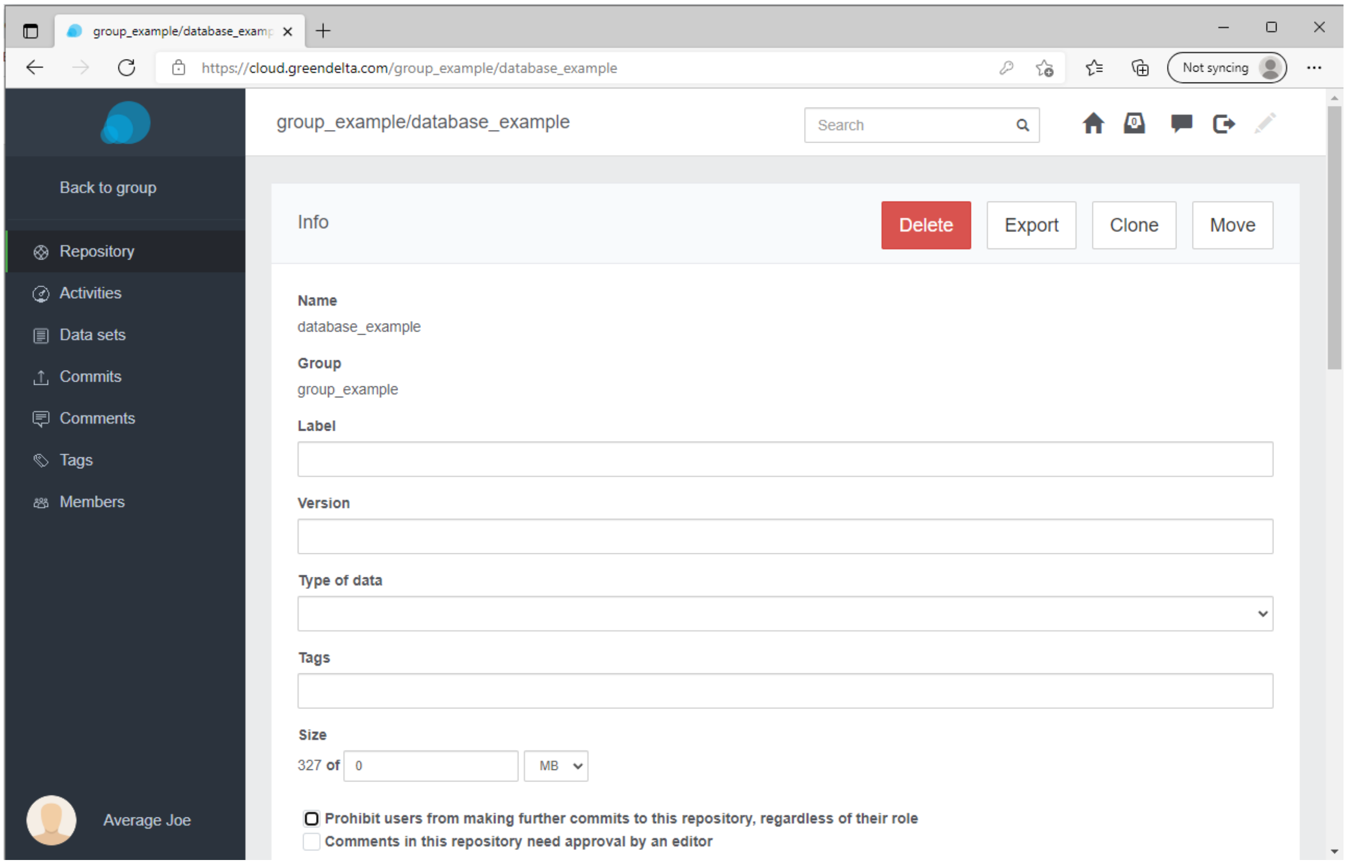
Email*
[redacted]

Figure 2-3 User's dashboard

2.3.1 Expanding on the repository dashboard

After clicking on a repository in the repository library (Figure *2-2* The LCA Collaboration Server dashboard), the following dashboard is displayed:

[4] A repository is the equivalent to a database in openLCA and consists of grouped data sets (projects, product systems, processes, flows, indicators and parameters and background data). Basically, a repository mirrors the local openLCA databases of the working group's users.



Name
database_example

Group
group_example

Label

Version

Type of data

Tags

Size
327 of 0 MB

□ Prohibit users from making further commits to this repository, regardless of their role
□ Comments in this repository need approval by an editor

Figure 2-4 Repository dashboard, of which Average Joe has an "owner" role

Repository

The first tab, **Repository**, shows the information of the repository. Further down, there is the "Metadata" section, a great opportunity to document the database. The **repository ID** is the **"groupname/repositoryname"**. The repository name can be changed following:

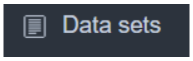
Move -> Name

Caution, this will also change the ID of the repository. To change the name without changing the ID, add a "Label", below "Group" in Figure 2-4, to override the repository name on the repository library, without changing the ID.

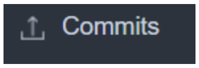
The user can add **tags** which further classify repositories through this tab. They can look for repositories with specific tags through the search function.

Activities

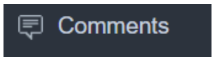
Activities logs the activities of the repository (i.e. commits or comments to the repository), which are also displayed in the "Activities" of the main menu.



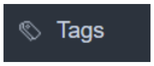
The **Data sets** tab allows to see all of the data in the repository, and also to download and export repositories and datasets through the download feature shown in Figure 2-5. It allows to download individual versions (commits) of a repository as JSON-LD or ILCD.



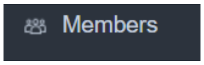
The **Commits** tab logs all of the commits to the repository, with the option to download the commit history too. The data committed can always be commented too.



The **Comments** tab holds all of the comment history of comments to committed processes.

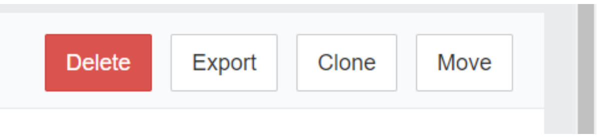


Adding *tags* to repositories can be done in *Repository tab*. individual datasets can also be tagged, as well as full repositories.

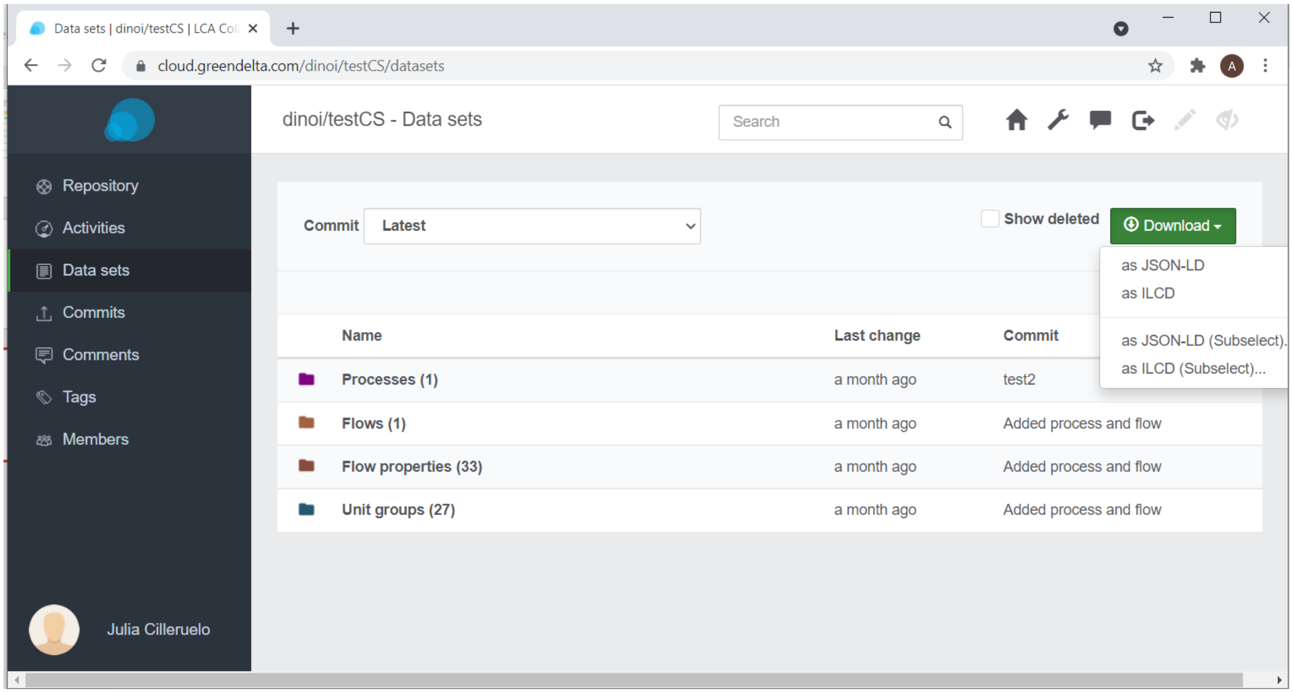


The **Members** tab holds a record of both the members of the repository, and the members of the repository through a group. Members can be added and deleted here, and also the user roles described in Section 2.3.3 can be changed.

Furthermore, the “owner” of the repository will be able to see the following managing tools for their repository:



More on user roles will be explained in section 2.3.3, and these buttons will also be explained in section 4.2.1.



dinoi/testCS - Data sets

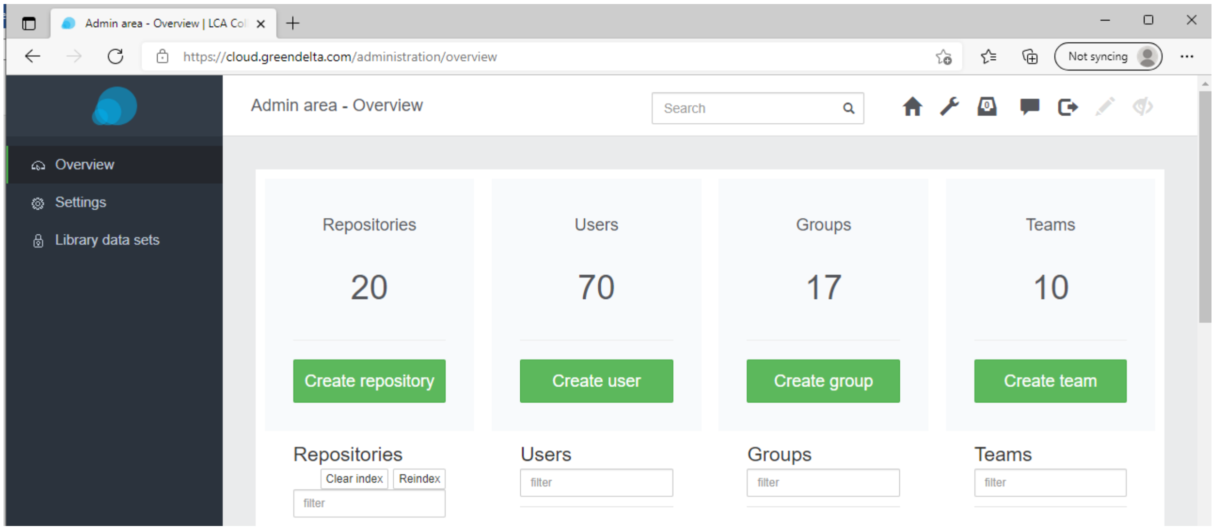
Commit: Latest

Name	Last change	Commit
Processes (1)	a month ago	test2
Flows (1)	a month ago	Added process and flow
Flow properties (33)	a month ago	Added process and flow
Unit groups (27)	a month ago	Added process and flow

Figure 2-5 Download and export of data sets

2.3.2 Expanding on the admin area dashboard

One can access the admin area by clicking on the 🔧 icon. The icon can only be seen by data managers, user managers, or server administrator. The regular user cannot see it. From the admin area, one can create, delete or manage repositories, users, groups of repositories and teams of users. In Settings the admin user can configure the Collaboration Server and enable or disable features.



Admin area - Overview

Repositories	Users	Groups	Teams
20	70	17	10

Figure 2-6 Admin area in the LCA Collaboration Server

2.3.3 Groups and user roles

The group feature allows to create *working groups* to which a repository or several repositories (see below) can be assigned. A working group can be managed by assigning different user roles to its members (Table 2-2)[5]. A summary of the differences is resumed in Table *2-2*.

Table 2-2 User rights

Rights	Reader	Contributor	Reviewer	Editor	Owner
Read repositories and fetch contents	X	X	X	X	X
Commit data to repositories		X	X	X	X
Comment specific fields of data sets			X	X	X
Review comments			X	X	X
Manage comments				X	X
Manage reviews				X	X
Create repositories					X
Edit repository members					X
Adjust settings					X
Move repositories					X
Delete repositories					X
Generate change log					X

[5] Data manager (administrate libraries) and user manager (create users and teams) are other additional roles; p.27

3 How-To: Basic Workflow

3.1 Repositories - Create a new repository for your working group

Go to *Repository* in the main menu. Click + ▾ to create or import a new repository. There are several ways to upload the database the user wants to work with to the repository in the Collaboration Server:

1. Create new repository and commit the database from openLCA (section 3.3)
2. Import JSON-LD file from a previously exported JSON-LD file from the desired database in openLCA.
3. Import repository – this option imports a previously exported repository zip file (e.g. from another CS, or maybe a backup file)

It is very important that all of the users that will be using the repository have the same version of the database that is uploaded to the repository. Users can also create a new database in their openLCA and '*fetch'* the database in the repository.

When creating a repository, it is possible to limit the repository's maximum size on the server (*0 = unlimited*). For more about the repository board, see section 2.3.1.

3.2 How to connect and disconnect the LCA CS to your openLCA and database

Connecting openLCA to the LCA CS

The user must have openLCA[6] installed in their computer, and a server holding the LCA Collaboration Server assigned.

On your openLCA, to open the window shown Figure 3-1, click

file > settings > collaboration,

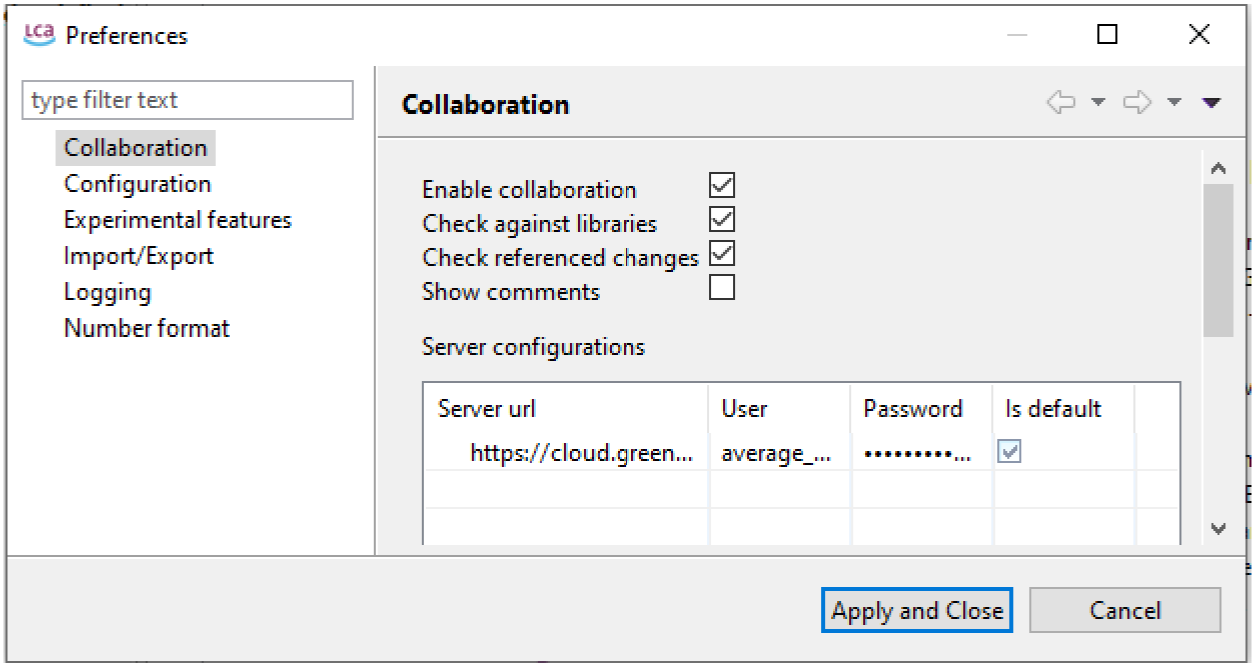
Check *enable collaboration*. Right click on the table in the same window and click *create new*. Please enter your server url, your user name, password, and click *is default* to proceed. Click *Apply and Close*.

Tip: if it doesn't work, double check you have the correct url, and you haven't accidentally included any extra spaces or similar. The right url should be displayed after hovering over the logo in your LCA CS in the web.

Checked against libraries – checks if the user can commit specific data to that library. The server admin, or another high user role, can decide to choose certain data in the repository that should not be committed further.

[6] To download openLCA visit: https://www.openlca.org/download/

Check referenced changes – suggests new commits to the library that might otherwise cause an error. For example, the user has added a new flow to an existing process, where the flow is not yet in the LCA CS repository. When the process is committed, the flow should also be committed. This is reminded by the software, if this option is checked.



Collaboration

Enable collaboration ☑
Check against libraries ☑
Check referenced changes ☑
Show comments □

Server configurations

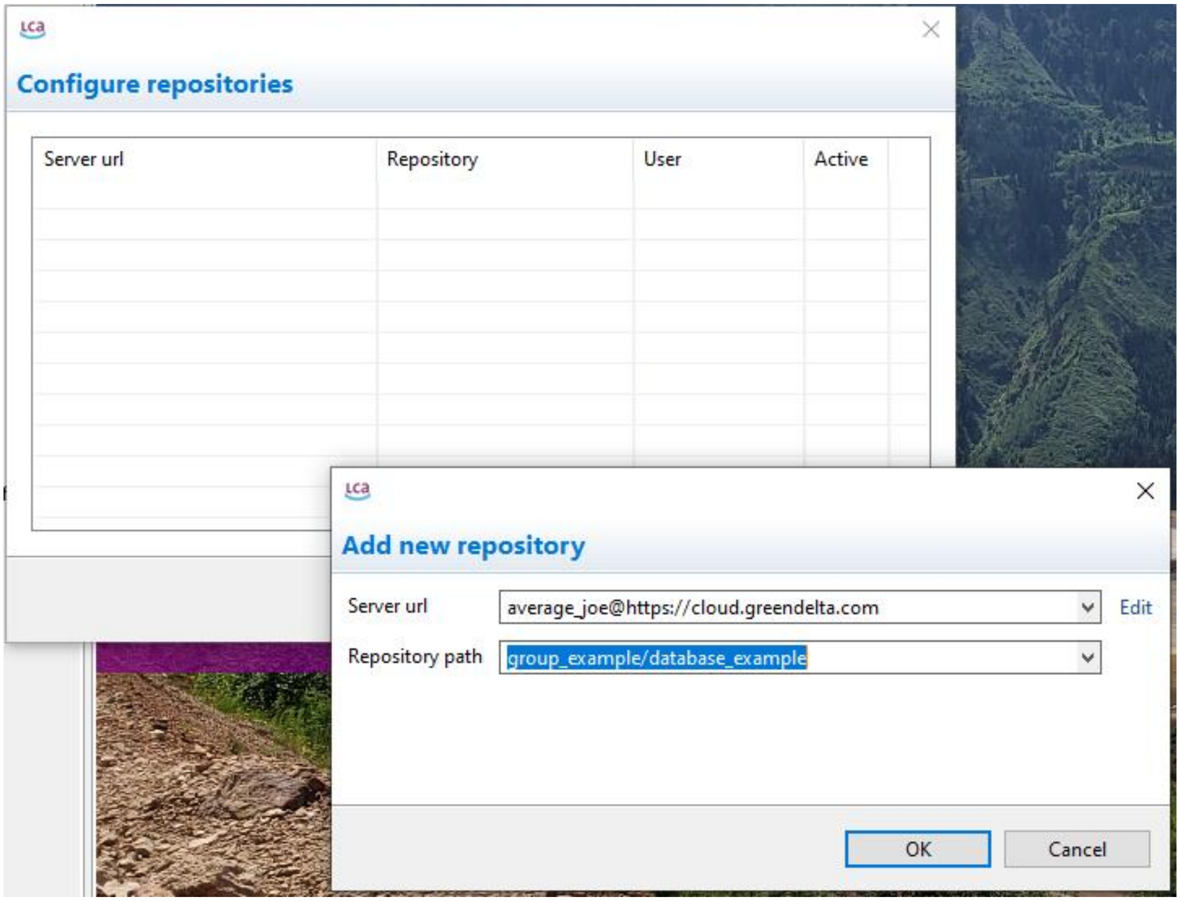
Server url	User	Password	Is default
https://cloud.green...	average_...	••••••••••...	☑

Figure 3-1 Preferences window in openLCA

Connecting a database in openLCA to the LCA CS

Once openLCA is connected to the LCA CS, the user can connect the database to a specific repository by right clicking on the database in openLCA > *repository* > *#configure...*

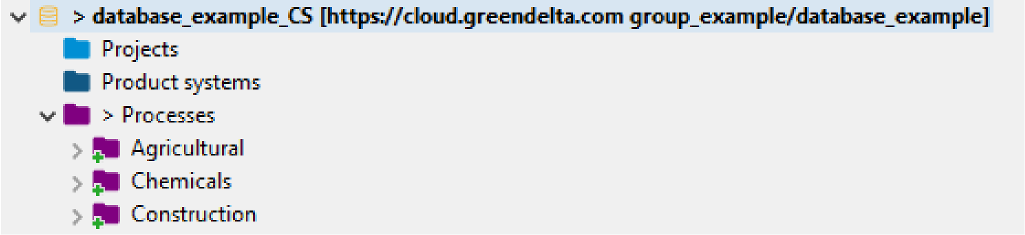
Right click on the empty table, and click *add* to add the repository (Figure 3-2). Then, click *OK* and set the database as *Active*. After clicking OK one last time, openLCA connects to the repository and rebuilds the index of the selected database. A connected database looks like the one shown in Figure *3-3*, where the URL is shown.



Server url: average_joe@https://cloud.greendelta.com

Repository path: group_example/database_example

Figure 3-2 add repository from openLCA to the LCA CS



> database_example_CS [https://cloud.greendelta.com group_example/database_example]
- Projects
- Product systems
- > Processes
  - Agricultural
  - Chemicals
  - Construction

Figure 3-3 Connected database

To disconnect from a repository right-click on the respective database and select *repository > disconnect from repository*.

3.3 Uploading the database to the LCA collaboration Server for the first time

In the case that the user has created a new (empty) repository in the Collaboration Server, the database is added to the LCA CS by right clicking on the database name in openLCA > *Repository > Commit.* **The user should always include a commit message** to be able to click *commit*.

Uploading the whole database (first commit) to the LCA CS may take around **15 minutes**. openLCA may '*not respond*' anymore. This is only temporary and may last for several minutes. The commit is still ongoing (see network activity). It is also possible to select only specific data sets for a commit via right-click + CTRL.

3.4 Committing data sets to the connected repository

Upon successfully connecting a database to a repository, a right-click on any item in the list underneath the database's name displays the new menu entry *Repository* (Figure 3-4).

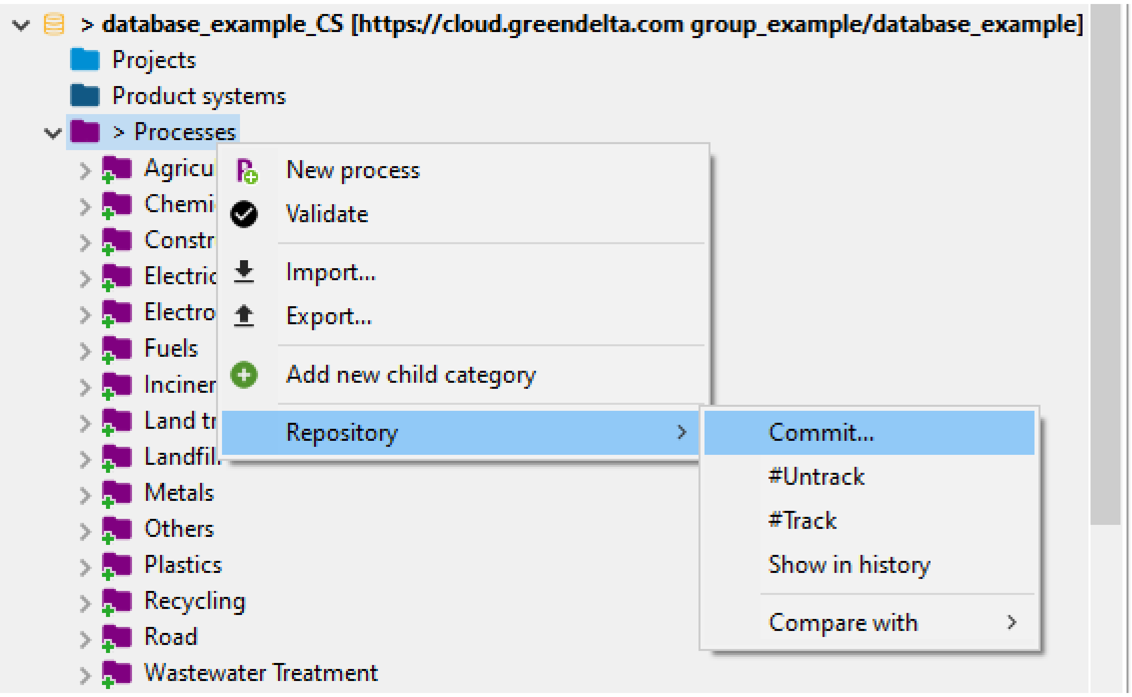
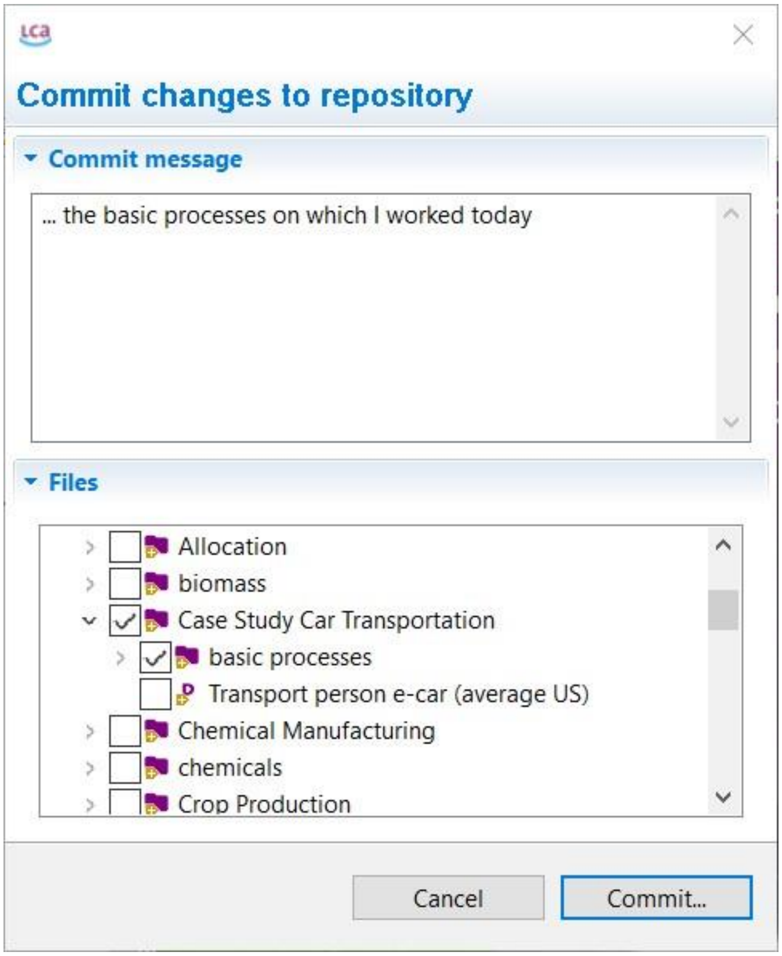


Figure 3-4 Right-click on any item in the database to initiate a commit

Click on commit to initiate committing changes to the repository. The user will be prompted to select which data sets are to be committed. A commit message is mandatory and must be added to each commit. The speed of the commit depends on network connection and the size of the database.



... the basic processes on which I worked today

- □ Allocation
- □ biomass
- ☑ Case Study Car Transportation
  - ☑ basic processes
  - □ Transport person e-car (average US)
- □ Chemical Manufacturing
- □ chemicals
- □ Crop Production

Figure 3-5 Committing changes to a repository requires a 'commit message'

Hint I: To maintain transparency and clarity, it is recommended to regularly commit small data sets rather than to occasionally commit large data sets. This help steam members to trace and understand changes, guarantees clean versioning of a repository and helps to avoid pitfalls (see section 3.8).

Hint II: If commits are large (e.g. entire databases) openLCA may '*not respond*' anymore. This is only temporary and may last for several minutes. The commit is still ongoing (see network activity). It is also possible to select only specific data sets for a commit via right-click + CTRL.

3.5 Fetching data sets from the repository

To fetch data sets from a repository, right-click on a local database (make sure you're connected to the repository) and select *repository > fetch* (Figure 3-6).

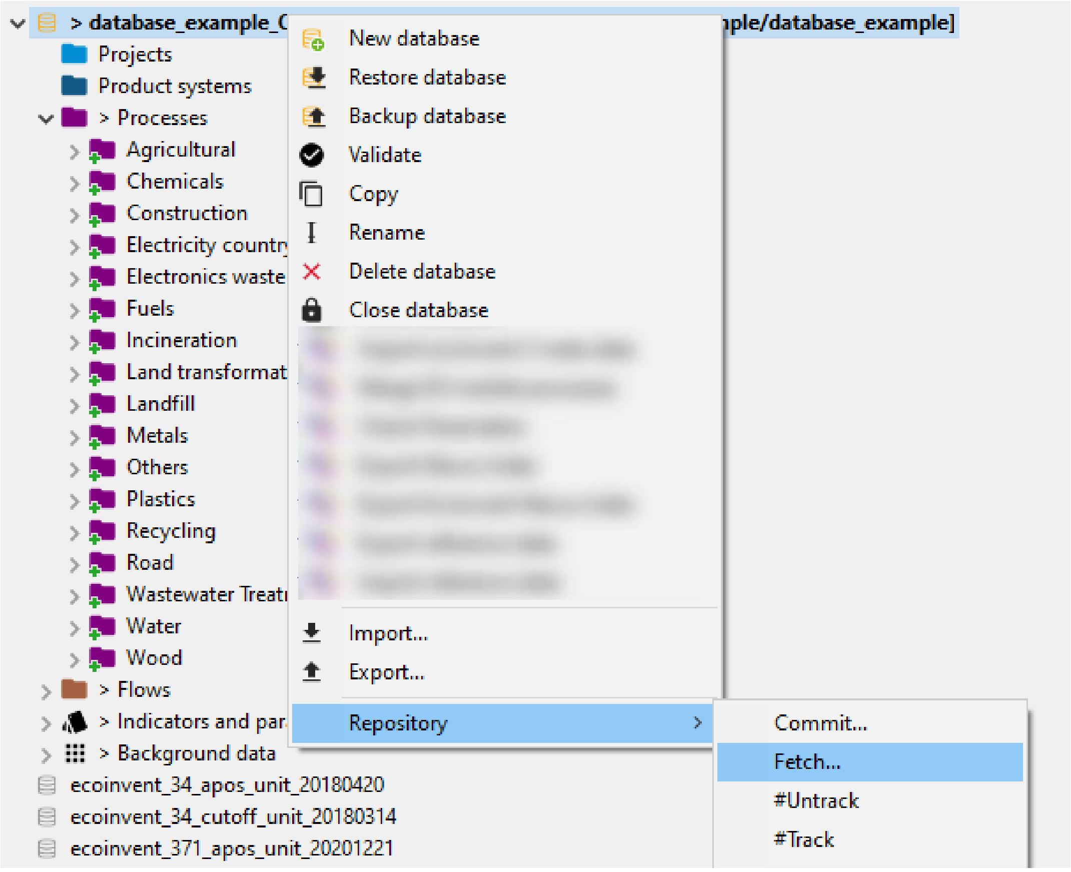
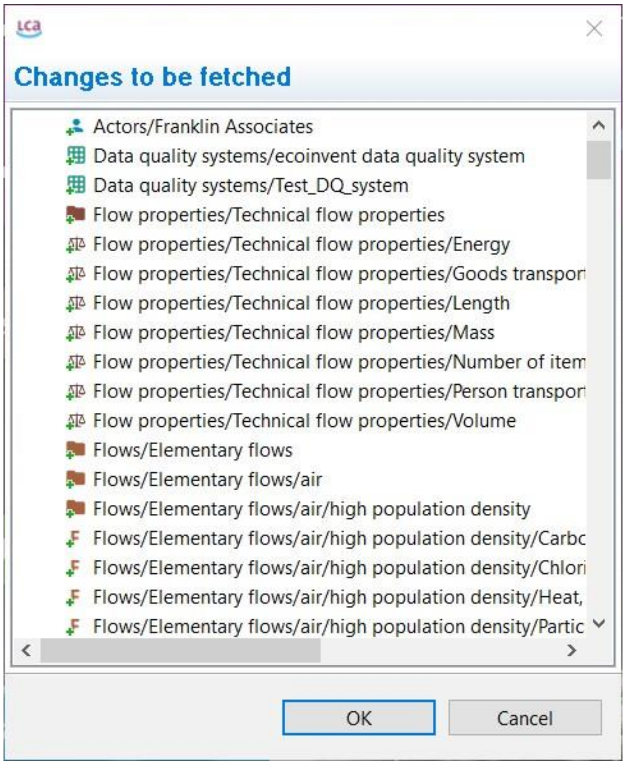


Figure 3-6 Fetching a repository from the LCA Collaboration Server in openLCA

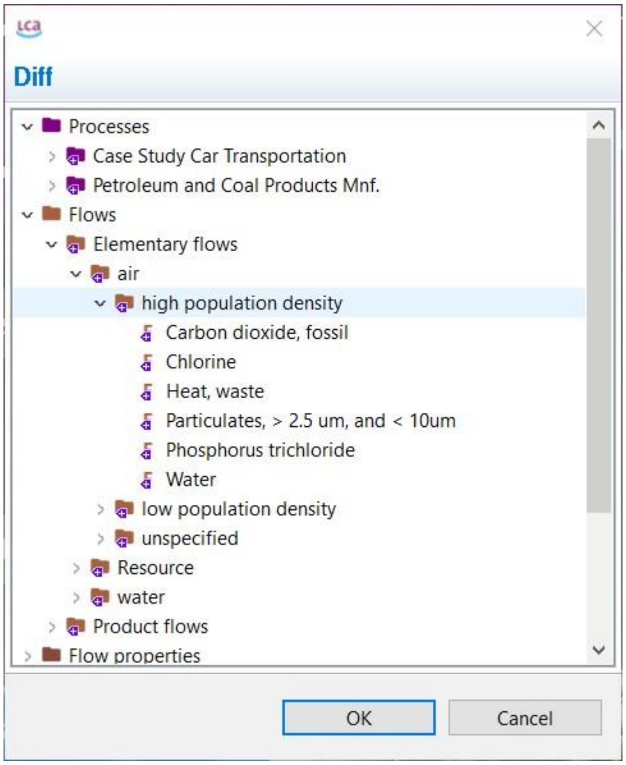
Two windows will pop up: The first one lists all changes of each individual commit since the last fetch (Figure 3-7). The second one lists all commits since the last fetch in a merged and structured list (Figure 3-8).



Changes to be fetched

- Actors/Franklin Associates
- Data quality systems/ecoinvent data quality system
- Data quality systems/Test_DQ_system
- Flow properties/Technical flow properties
- Flow properties/Technical flow properties/Energy
- Flow properties/Technical flow properties/Goods transpor
- Flow properties/Technical flow properties/Length
- Flow properties/Technical flow properties/Mass
- Flow properties/Technical flow properties/Number of item
- Flow properties/Technical flow properties/Person transpor
- Flow properties/Technical flow properties/Volume
- Flows/Elementary flows
- Flows/Elementary flows/air
- Flows/Elementary flows/air/high population density
- Flows/Elementary flows/air/high population density/Carbc
- Flows/Elementary flows/air/high population density/Chlori
- Flows/Elementary flows/air/high population density/Heat,
- Flows/Elementary flows/air/high population density/Partic

Figure 3-7 Simple list of changes that need to be fetched



Diff

```
Processes
  Case Study Car Transportation
  Petroleum and Coal Products Mnf.
Flows
  Elementary flows
    air
      high population density
        Carbon dioxide, fossil
        Chlorine
        Heat, waste
        Particulates, > 2.5 um, and < 10um
        Phosphorus trichloride
        Water
      low population density
      unspecified
    Resource
    water
  Product flows
Flow properties
```

Figure 3-8 Structured list of all changes that need to be fetched

Hint: To maintain transparency and clarity, it is recommended to fetch regularly. This enables the user to trace and understand changes easily and helps to avoid conflicts (see section 3.8).

3.6 The workflow is always linear

To understand how exactly the LCA Collaboration Server works, it is important to note that the workflow of *commits* and *fetches* is always linear. In practice, this means for instance that users cannot commit data sets to the server indiscriminately. In fact, **before each commit, the user has to fetch the current state of the repository** from the LCA Collaboration Server. (Figure 3-9).

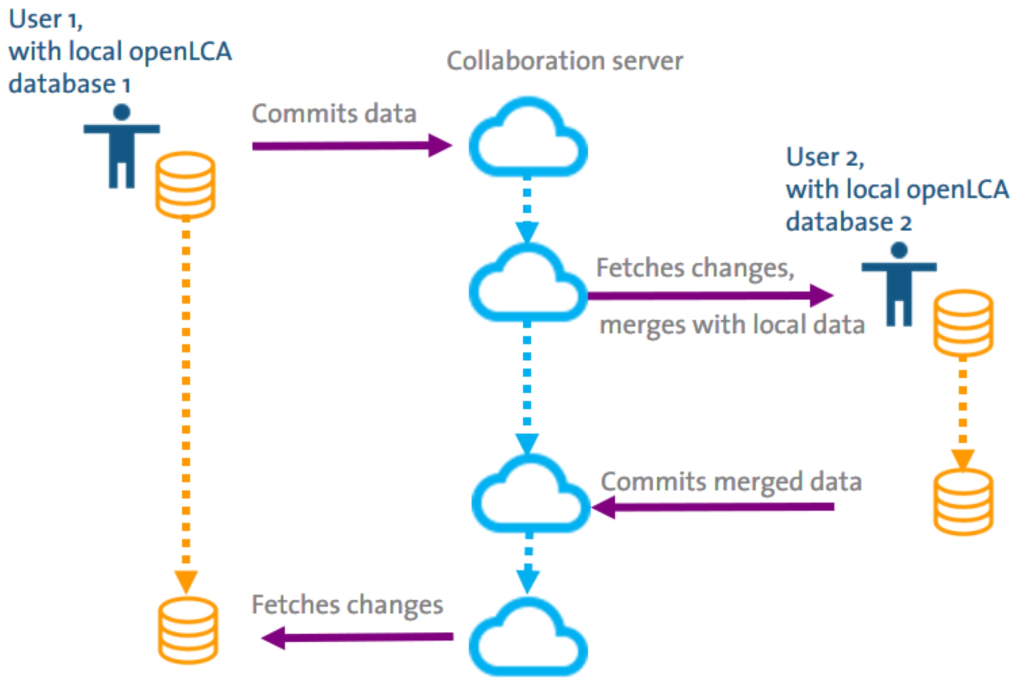


Figure 3-9 The workflow of a commit is always linear

3.7 Conflicts

Conflicts may occur if two or more users work on the same database item or interlinked database items at the exact same time. This is best avoided by directly coordinating work within your team as well as to perform regular fetches and to commit regularly. The diff utility (see section 4.2.7) assists in detecting commits by other users and to identify potential conflicts (see section 3.8).

3.8 Pitfalls

Eventually, users are responsible for the integrity of a repository and must understand that a repository is a complex interlinked system. Interlinkages must be respected and a commit must always contain all interlinkages to any of the changes in the data set.

3.8.1 Validating databases

To assure that all interlinkages within a database are set correctly, databases should be validated regularly (right-click on a database and select *validate,* Figure 3-10, Figure 3-11). A validation identifies whether all linkages within a database are working.

If a validation fails, **missing interlinkages should be fixed before performing a commit**. If a user is unable to validate a database it is recommended to discard changes, fetch the repository again and to redo modelling steps. A user may fail to validate a database if corrupt data sets were fetched from the repository that have been committed by other users.

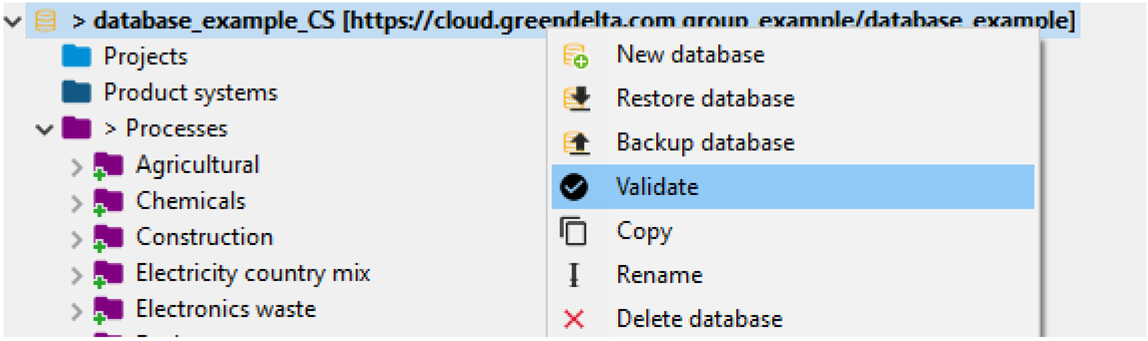
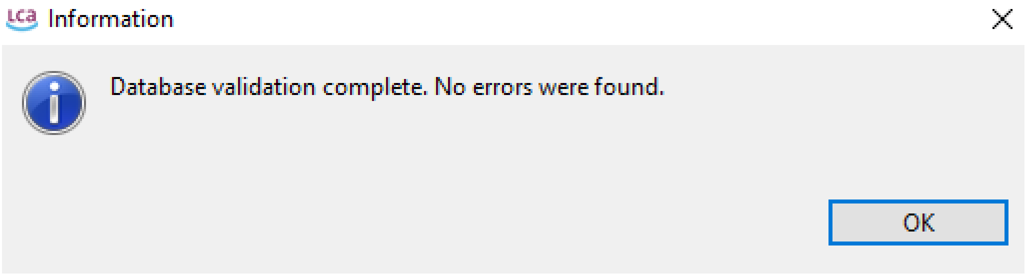


Figure 3-10 'Validate' allows to check whether all linkages within a database are working



Database validation complete. No errors were found.

Figure 3-11 Success message when validating the database

3.8.2 Example 1 - Missing interlinked flow

A user adds a flow to a local database and uses the flow as an input for a process. The user commits the new process to the repository but does not commit the underlying flow. The linked flow is missing in the repository and corrupts the database for other users.

3.8.3 Example 2 - Modified flow property

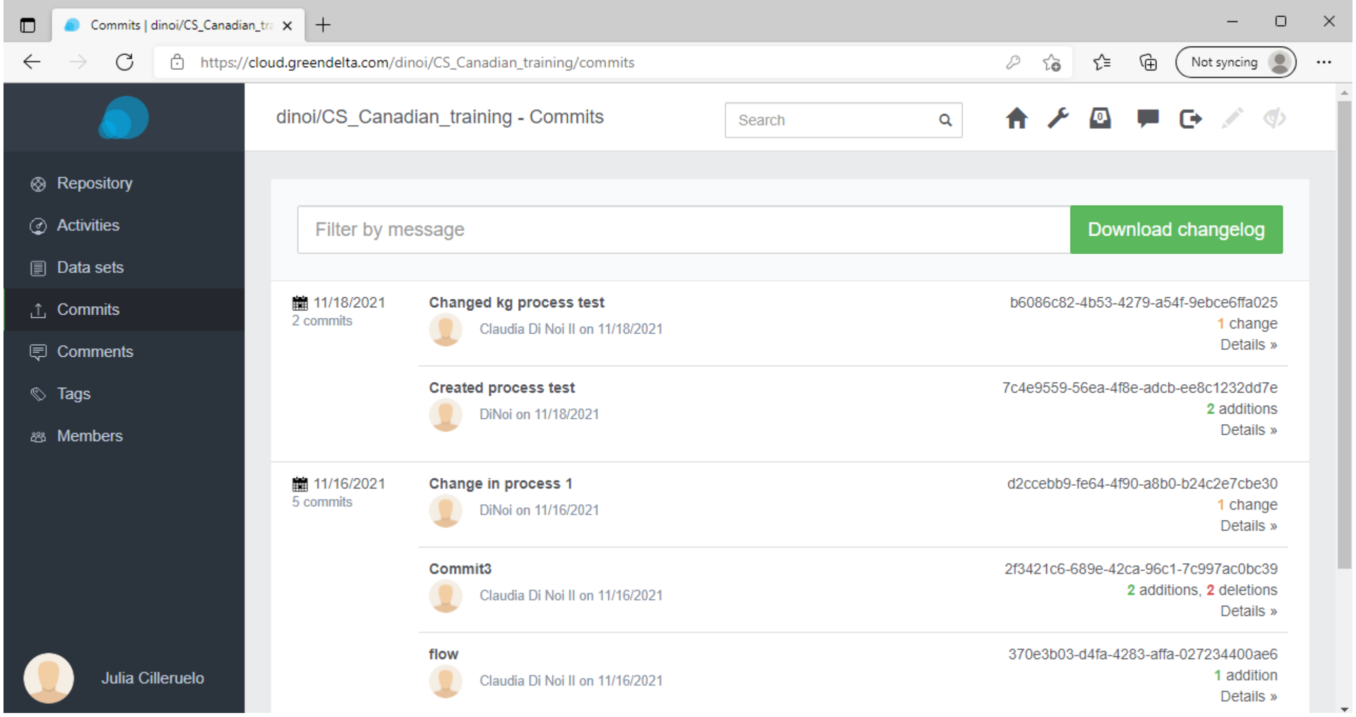
A user changes the flow property of an existing flow in a local openLCA database and uses the flow with the updated property in a process. The user commits the process to the repository but not the flow with the modified flow property. The modification of the flow is missing in the repository and corrupts the data set in the repository for other users.

4 Features

4.1 Basic features

4.1.1 Commit history

A history of all commits is available in openLCA via right-click on a database *Repository>Show in history* (Figure 4-2) and via the LCA Collaboration Server dashboard (Figure 4-1).



dinoi/CS_Canadian_training - Commits

11/18/2021
2 commits

Changed kg process test
Claudia Di Noi II on 11/18/2021
b6086c82-4b53-4279-a54f-9ebce6ffa025
1 change
Details »

Created process test
DiNoi on 11/18/2021
7c4e9559-56ea-4f8e-adcb-ee8c1232dd7e
2 additions
Details »

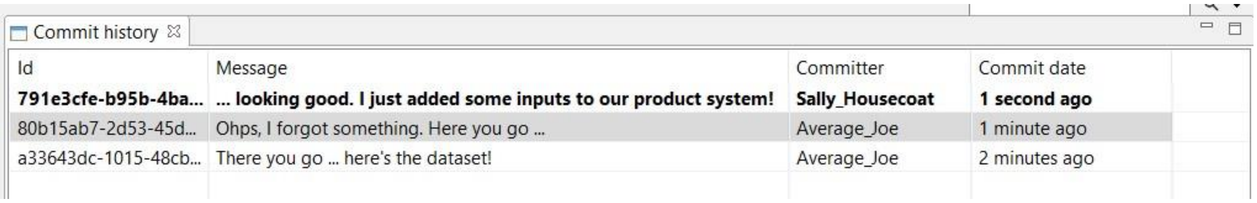
11/16/2021
5 commits

Change in process 1
DiNoi on 11/16/2021
d2ccebb9-fe64-4f90-a8b0-b24c2e7cbe30
1 change
Details »

Commit3
Claudia Di Noi II on 11/16/2021
2f3421c6-689e-42ca-96c1-7c997ac0bc39
2 additions, 2 deletions
Details »

flow
Claudia Di Noi II on 11/16/2021
370e3b03-d4fa-4283-affa-027234400ae6
1 addition
Details »

Figure 4-1 History of commits in the LCA Collaboration Server dashboard



Id	Message	Committer	Commit date
791e3cfe-b95b-4ba...	**... looking good. I just added some inputs to our product system!**	**Sally_Housecoat**	**1 second ago**
80b15ab7-2d53-45d...	Ohps, I forgot something. Here you go ...	Average_Joe	1 minute ago
a33643dc-1015-48cb...	There you go ... here's the dataset!	Average_Joe	2 minutes ago

Figure 4-2 History of commits in openLCA

4.1.2 Checkout commits

A *checkout* replaces the local openLCA database with a selected commit (version of the repository) from the LCA Collaboration Server. To checkout a commit, open the commit history via right-click *repository > show in history*. Right-click on a commit in the commit history in openLCA and select checkout (Figure 4-3). After the checkout, the database in openLCA is different from the repository. Before the next commit, the local database in openLCA would be replaced with the repository and thus the checkout reversed. However, users may want to check a commit out, disconnect from the repository and continue to work on the checked-out version of the database.

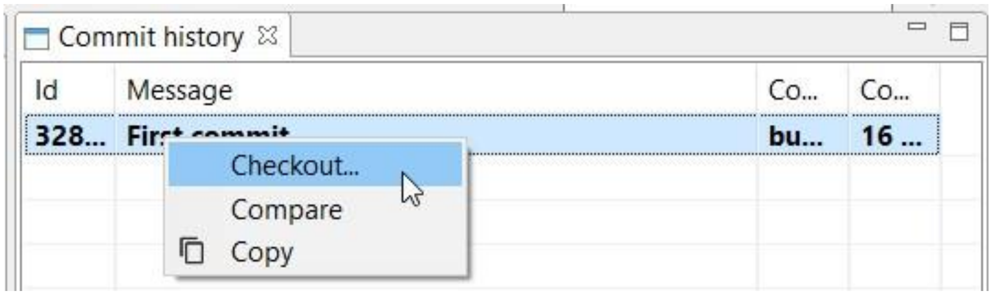
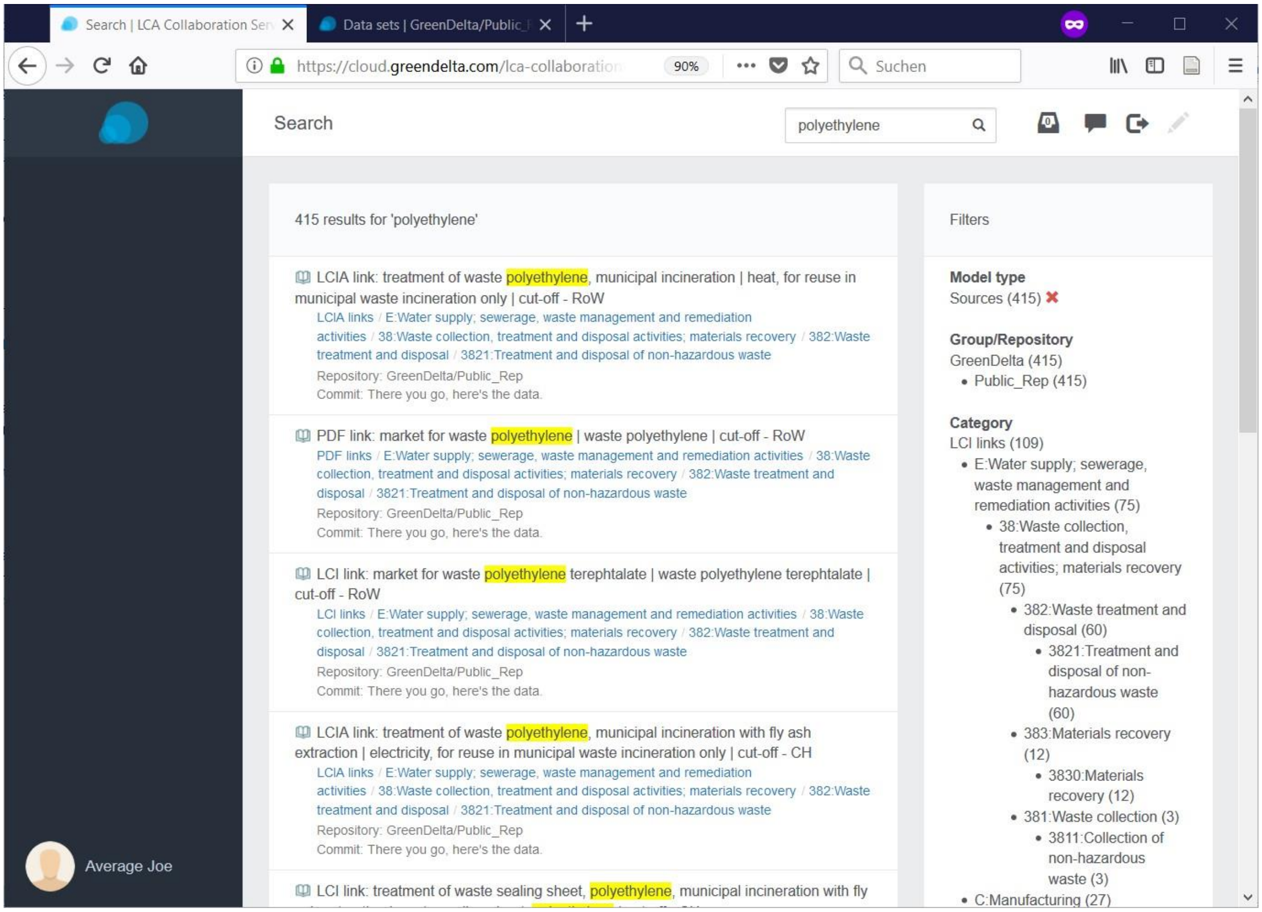


Figure 4-3 A checkout replaces the local openLCA database with a specific commit

4.1.3 Search function and filter (LCA Collaboration Server dashboard & openLCA)

A repository-overarching search function is available via the LCA Collaboration Server dashboard. Simply use the search form in the top bar. Filters allow enhanced navigation through the search results (Figure 4-4 and Figure 4-5). All search items are referenced and link to the respective items. The search function also allows unregistered users to search through a repository if the latter is a public repository.

From openLCA v1.11 onwards, the user will be allowed to search and download datasets from collaboration servers without needing to connect the database to the LCA CS.


Search | LCA Collaboration Ser
Data sets | GreenDelta/Public_
https://cloud.greendelta.com/lca-collaboration
90%
Suchen
Search
polyethylene
Average Joe

415 results for 'polyethylene'

LCIA link: treatment of waste polyethylene, municipal incineration | heat, for reuse in municipal waste incineration only | cut-off - RoW
LCIA links / E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 382:Waste treatment and disposal / 3821:Treatment and disposal of non-hazardous waste
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.

PDF link: market for waste polyethylene | waste polyethylene | cut-off - RoW
PDF links / E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 382:Waste treatment and disposal / 3821:Treatment and disposal of non-hazardous waste
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.

LCI link: market for waste polyethylene terephtalate | waste polyethylene terephtalate | cut-off - RoW
LCI links / E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 382:Waste treatment and disposal / 3821:Treatment and disposal of non-hazardous waste
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.

LCIA link: treatment of waste polyethylene, municipal incineration with fly ash extraction | electricity, for reuse in municipal waste incineration only | cut-off - CH
LCIA links / E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 382:Waste treatment and disposal / 3821:Treatment and disposal of non-hazardous waste
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.

LCI link: treatment of waste sealing sheet, polyethylene, municipal incineration with fly

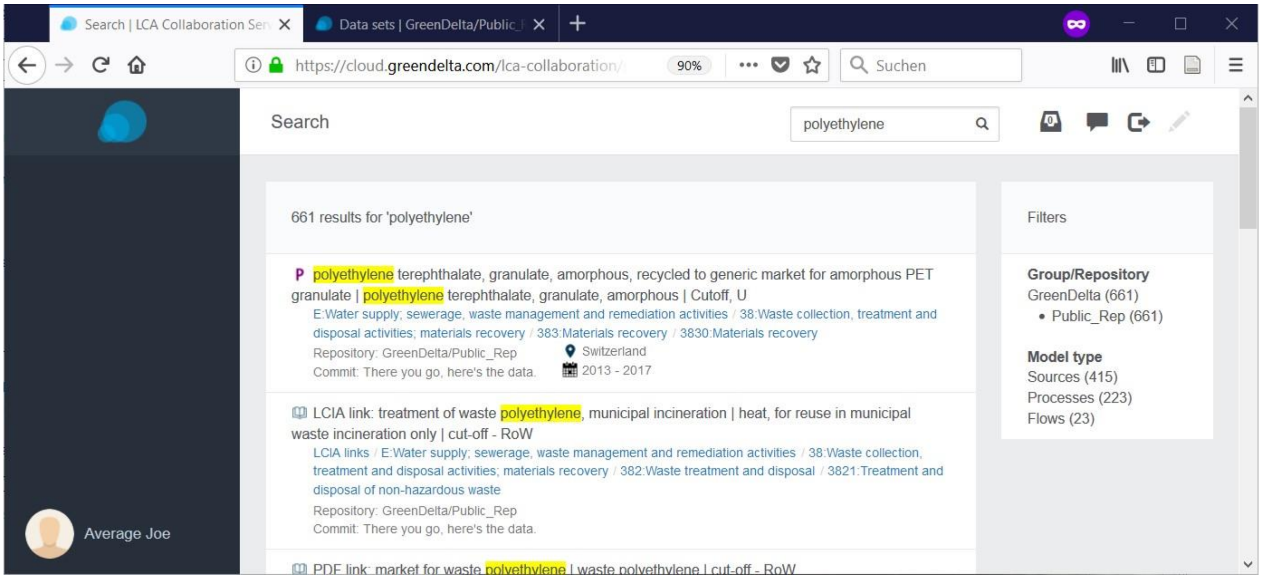
Filters

Model type
Sources (415) ✖

Group/Repository
GreenDelta (415)
- Public_Rep (415)

Category
LCI links (109)
- E:Water supply; sewerage, waste management and remediation activities (75)
  - 38:Waste collection, treatment and disposal activities; materials recovery (75)
    - 382:Waste treatment and disposal (60)
      - 3821:Treatment and disposal of non-hazardous waste (60)
    - 383:Materials recovery (12)
      - 3830:Materials recovery (12)
    - 381:Waste collection (3)
      - 3811:Collection of non-hazardous waste (3)
- C:Manufacturing (27)

Figure 4-4 Grouped search results


Search | LCA Collaboration Ser
Data sets | GreenDelta/Public_
https://cloud.greendelta.com/lca-collaboration/
90%
Suchen
Search
polyethylene
Average Joe

661 results for 'polyethylene'

P polyethylene terephthalate, granulate, amorphous, recycled to generic market for amorphous PET granulate | polyethylene terephthalate, granulate, amorphous | Cutoff, U
E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 383:Materials recovery / 3830:Materials recovery
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.
Switzerland
2013 - 2017

LCIA link: treatment of waste polyethylene, municipal incineration | heat, for reuse in municipal waste incineration only | cut-off - RoW
LCIA links / E:Water supply; sewerage, waste management and remediation activities / 38:Waste collection, treatment and disposal activities; materials recovery / 382:Waste treatment and disposal / 3821:Treatment and disposal of non-hazardous waste
Repository: GreenDelta/Public_Rep
Commit: There you go, here's the data.

PDF link: market for waste polyethylene | waste polyethylene | cut-off - RoW

Filters

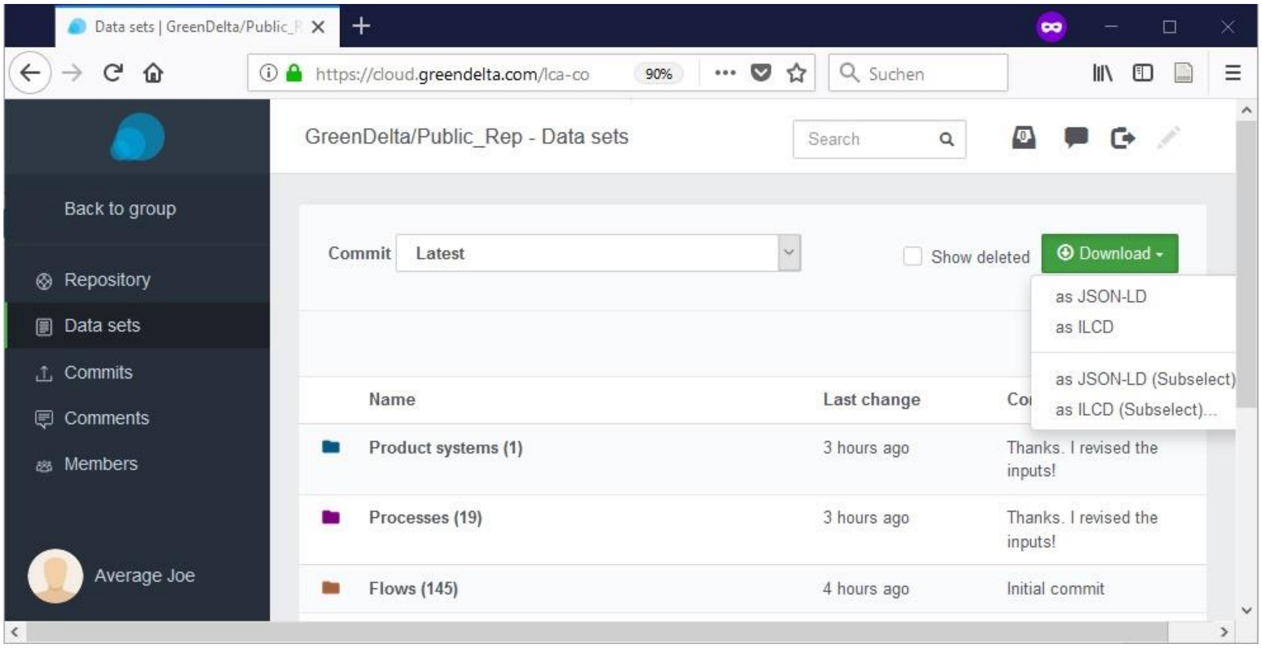
Group/Repository
GreenDelta (661)
- Public_Rep (661)

Model type
Sources (415)
Processes (223)
Flows (23)

Figure 4-5 Ungrouped search results

4.1.4 Download of whole repositories or selected datasets

The download feature allows to download individual versions (commits) of a repository as *JSON-LD* or *ILCD* (Figure 4-6) file. It is also possible to download individual datasets from the Collaboration Server.



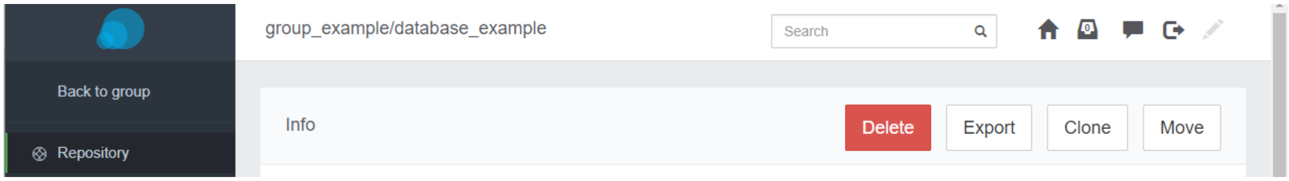
Name	Last change	Co
Product systems (1)	3 hours ago	Thanks. I revised the inputs!
Processes (19)	3 hours ago	Thanks. I revised the inputs!
Flows (145)	4 hours ago	Initial commit

Figure 4-6 Download and export of data sets

4.2 Advanced features

4.2.1 Managing repositories

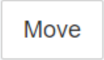
The "owner" of the repository will be able to see a set of buttons that can edit the entire repository.



Delete To delete the repository.

Export To export the repository as a zip file. The data format is specific to the LCA CS. This file can be uploaded again to the LCA CS.

Clone To make a copy of the repository in the LCA CS. It clones the whole content including metadata. One can select where you want to clone it to. You can clone over a certain point. This becomes handy when updating the database whilst maintaining the older version.



Move

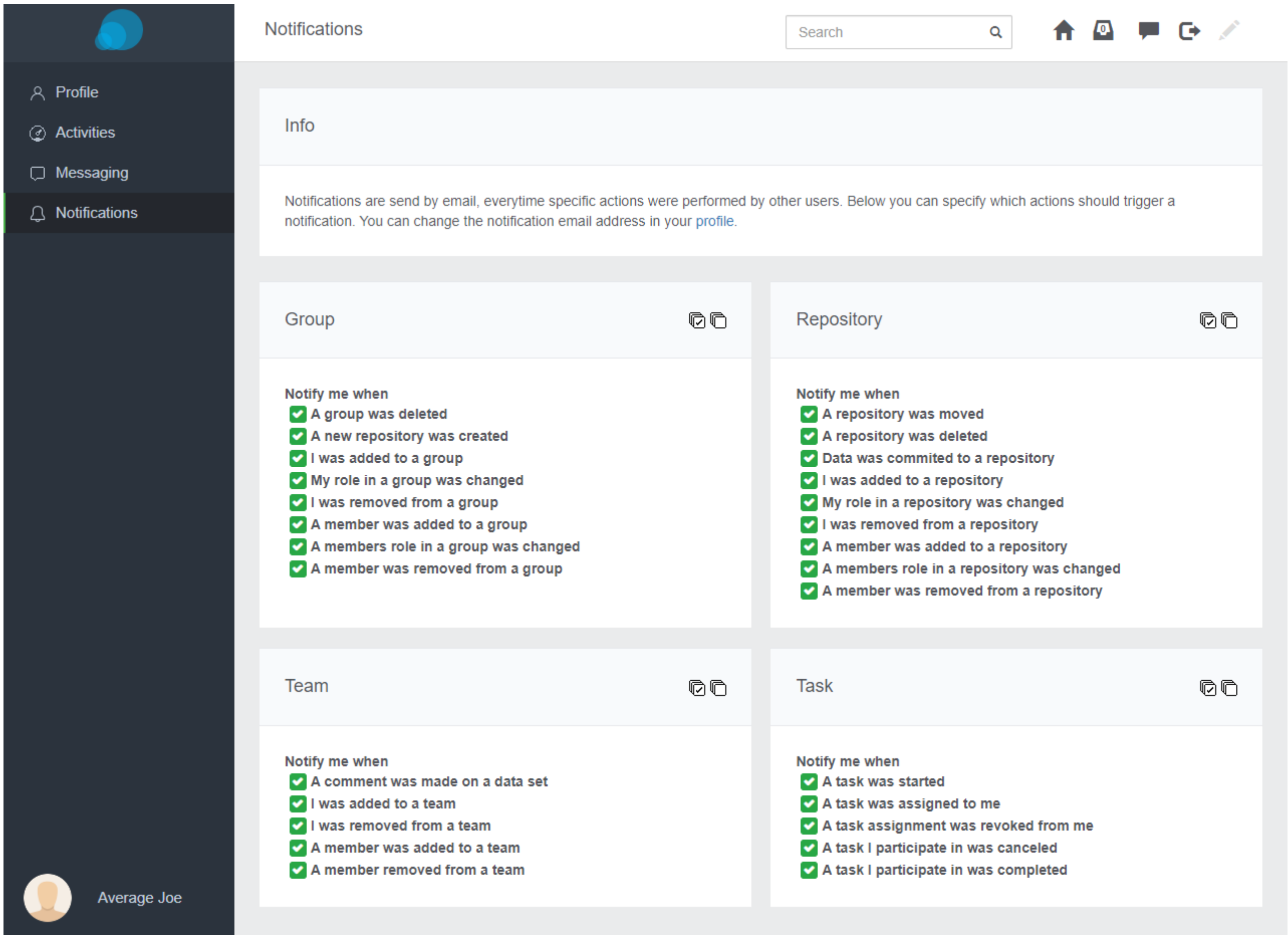
To change the name of a repository. Caution should be taken here, as this will also change the reference ID or repository path which will cause further problems with the connections to this repository. Every user should update the repository path.

4.2.2 Restoring previous versions of a dataset

There is no direct option to restore previous versions of a repository. However, a workaround does exist. Right click on a database in openLCA and select *Repository > compare with > HEAD revision*. Afterwards, right-click and select overwrite with local changes and commit.

4.2.3 Notifications

Notification settings can be adjusted upon clicking on the user's name in the bottom left corner of the LCA Collaboration Server dashboard. Afterwards, select *Notifications* from the entries in the menu on the left side (Figure 4-7).



Info

Notifications are send by email, everytime specific actions were performed by other users. Below you can specify which actions should trigger a notification. You can change the notification email address in your profile.

Group

Notify me when
☑ A group was deleted
☑ A new repository was created
☑ I was added to a group
☑ My role in a group was changed
☑ I was removed from a group
☑ A member was added to a group
☑ A members role in a group was changed
☑ A member was removed from a group

Repository

Notify me when
☑ A repository was moved
☑ A repository was deleted
☑ Data was commited to a repository
☑ I was added to a repository
☑ My role in a repository was changed
☑ I was removed from a repository
☑ A member was added to a repository
☑ A members role in a repository was changed
☑ A member was removed from a repository

Team

Notify me when
☑ A comment was made on a data set
☑ I was added to a team
☑ I was removed from a team
☑ A member was added to a team
☑ A member removed from a team

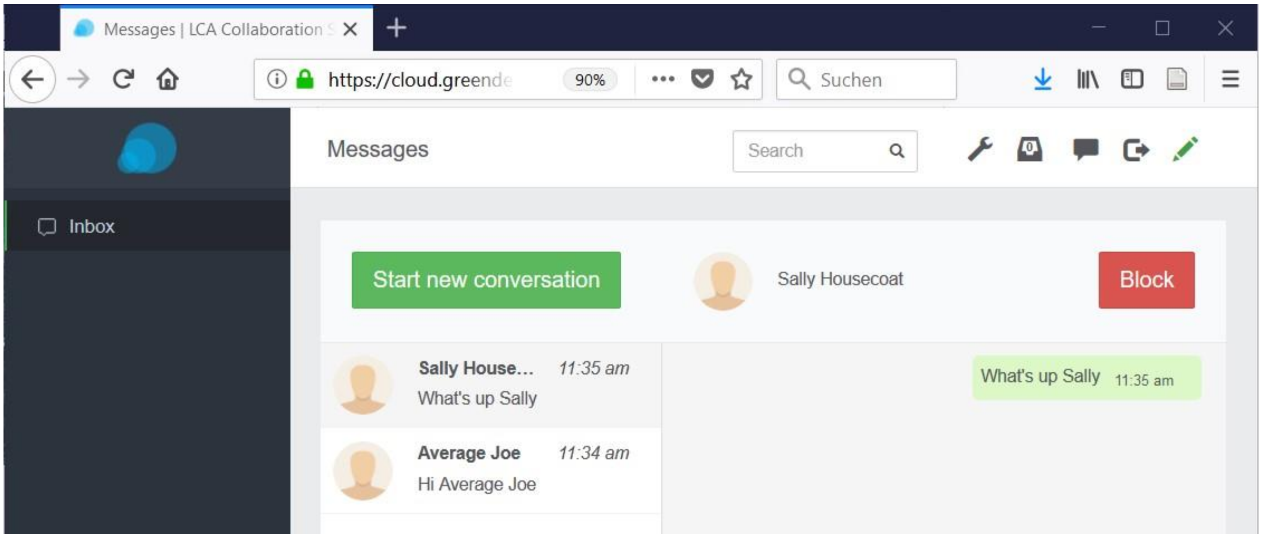
Task

Notify me when
☑ A task was started
☑ A task was assigned to me
☑ A task assignment was revoked from me
☑ A task I participate in was canceled
☑ A task I participate in was completed

Figure 4-7 Notifications settings

4.2.4 Messaging

A messaging feature for conversations among team members is available via the speech bubble in the top right-corner of the LCA Collaboration Server user dashboard (Figure 4-8). To enable the messaging feature click on your profile settings in the bottom-left corner of the LCA Collaboration Server dashboard and select Messaging in the left sidebar.



Sally House... 11:35 am
What's up Sally

Average Joe 11:34 am
Hi Average Joe

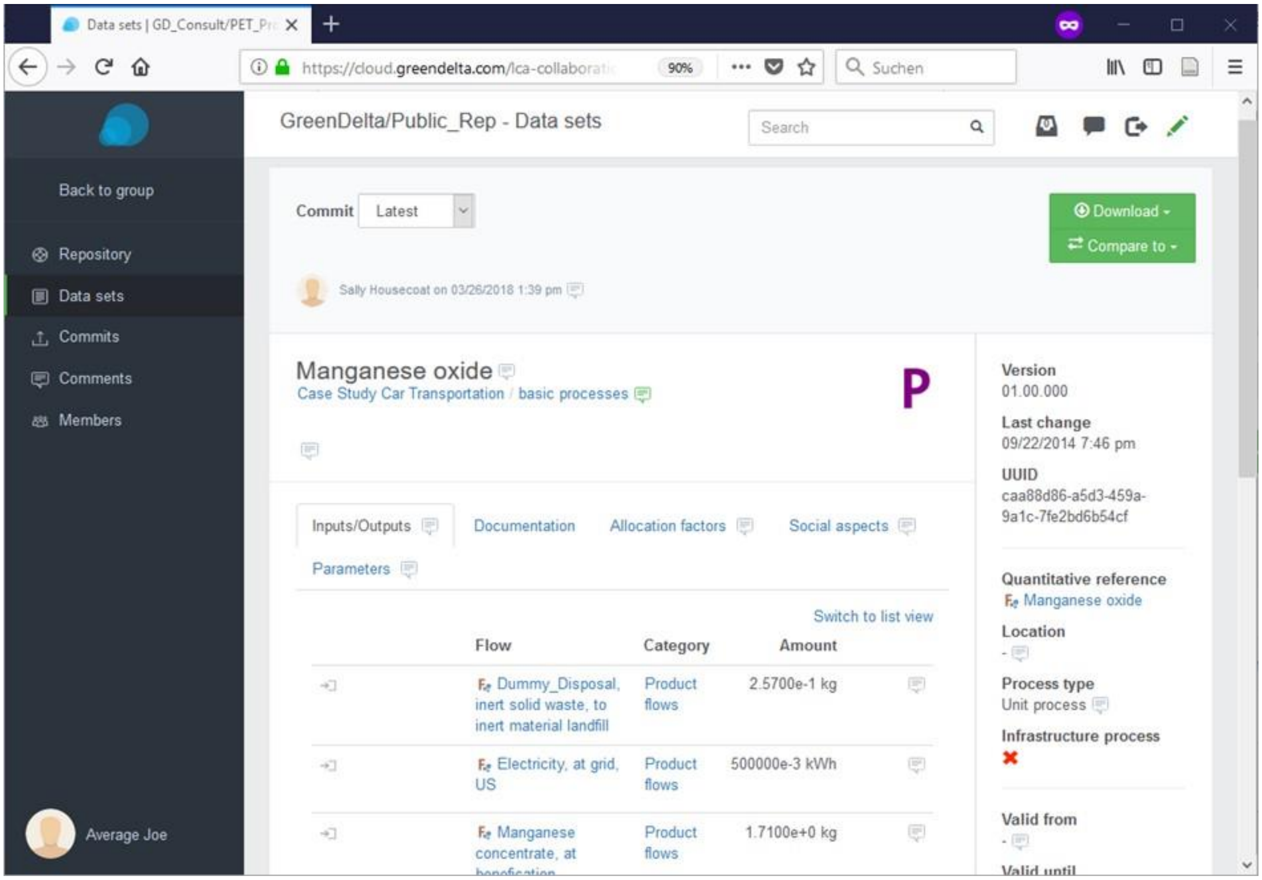
What's up Sally 11:35 am

Figure 4-8 The messaging feature can be accessed via the speech bubble in the top right corner

4.2.5 Comments and review mode

Specific fields of a data set as well as entire data sets as such can be commented. Comments can be added upon activating the review mode through a click on the pencil in the top-right corner of the LCA Collaboration Server dashboard. Once the review mode has been activated, speech bubbles show up for each item that is able to be commented (Figure 4-9). Regardless of whether the review mode is active or not, an existing comment is indicated through a green speech bubble (Figure 4-9).

Comments can be released immediately, added as a draft or added for a review by a moderator. An overview of all comments in the LCA Collaboration Server dashboard is available via the left sidebar (Figure 4-10). Comments can also be viewed in openLCA. Right-click on the respective database and navigate to *repository* > *show comments* (Figure 4-11). If an item has been commented, a 'Comments' tab shows up in the bottom right corner of the item's tab (Figure 4-12).



GreenDelta/Public_Rep - Data sets

Commit: Latest

Sally Housecoat on 03/26/2018 1:39 pm

Manganese oxide

Case Study Car Transportation / basic processes

P

	Flow	Category	Amount
	Dummy_Disposal, inert solid waste, to inert material landfill	Product flows	2.5700e-1 kg
	Electricity, at grid, US	Product flows	500000e-3 kWh
	Manganese concentrate, at beneficiation	Product flows	1.7100e+0 kg

Version: 01.00.000

Last change: 09/22/2014 7:46 pm

UUID: caa88d86-a5d3-459a-9a1c-7fe2bd6b54cf

Quantitative reference: Manganese oxide

Location: -

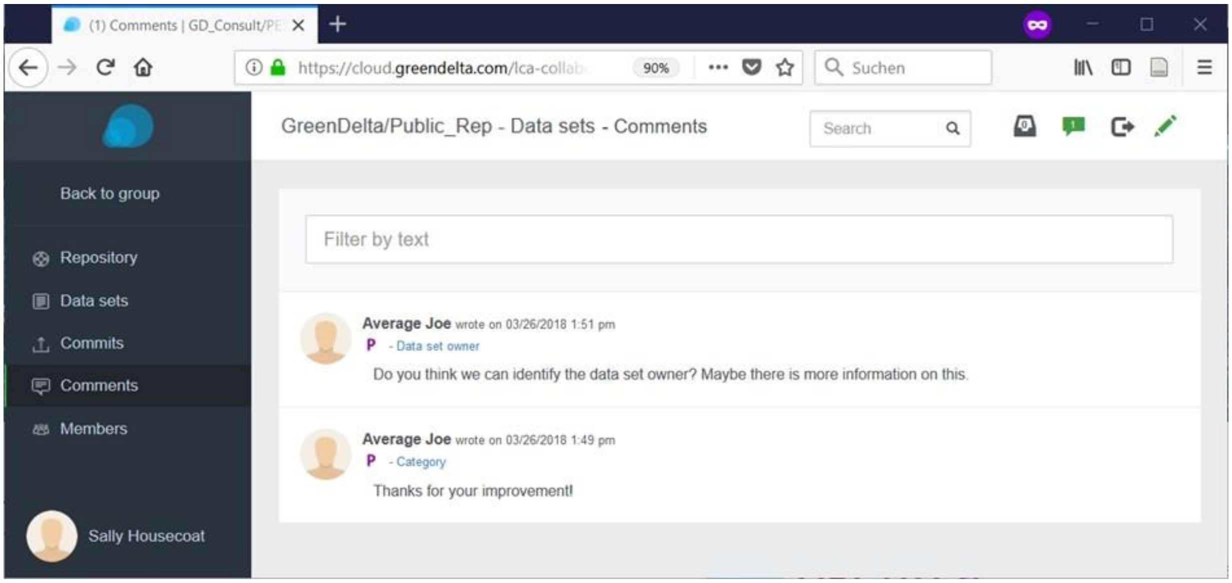
Process type: Unit process

Infrastructure process: ✗

Valid from: -

Valid until

Figure 4-9 Upon activating the review mode, comment bubbles can be seen in different locations. Regardless of whether the review mode is activated or not, commented items are indicated through a green speech bubble

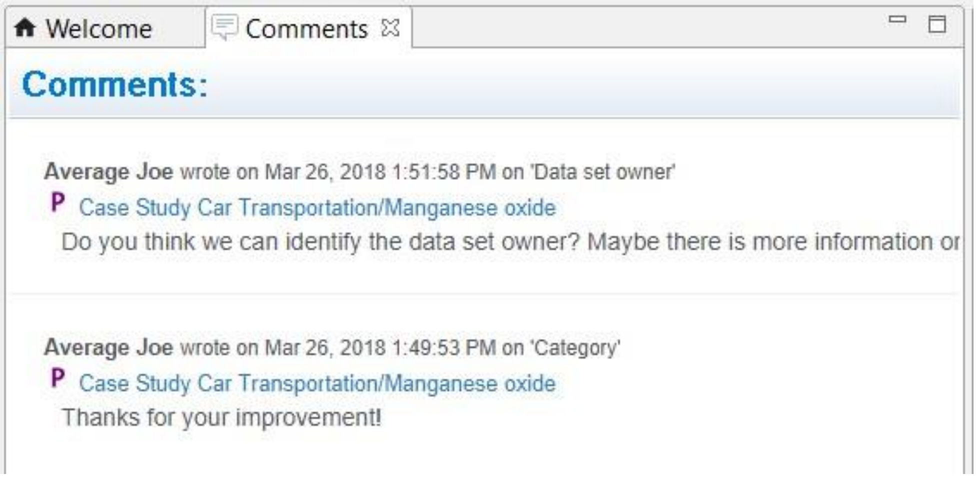


GreenDelta/Public_Rep - Data sets - Comments

Average Joe wrote on 03/26/2018 1:51 pm
P - Data set owner
Do you think we can identify the data set owner? Maybe there is more information on this.

Average Joe wrote on 03/26/2018 1:49 pm
P - Category
Thanks for your improvement!

Figure 4-10 Comment overview in the LCA Collaboration Server dashboard

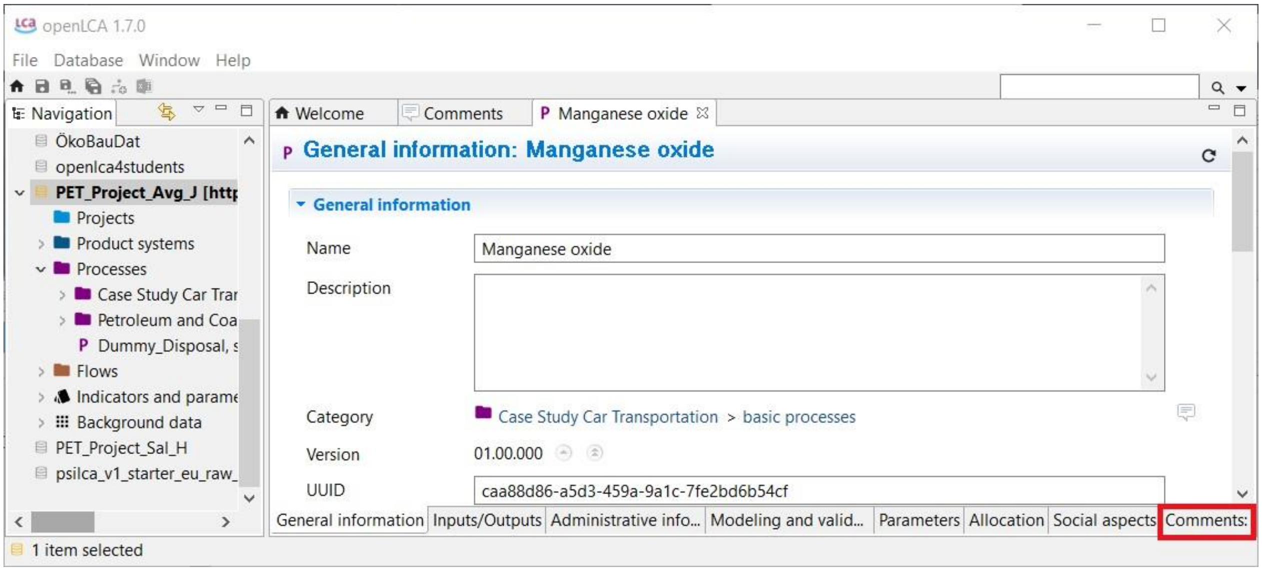


Comments:

Average Joe wrote on Mar 26, 2018 1:51:58 PM on 'Data set owner'
P Case Study Car Transportation/Manganese oxide
Do you think we can identify the data set owner? Maybe there is more information or

Average Joe wrote on Mar 26, 2018 1:49:53 PM on 'Category'
P Case Study Car Transportation/Manganese oxide
Thanks for your improvement!

Figure 4-11 Comment overview in openLCA



Name: Manganese oxide

Description:

Category: Case Study Car Transportation > basic processes

Version: 01.00.000

UUID: caa88d86-a5d3-459a-9a1c-7fe2bd6b54cf

Figure 4-12 If an item has been commented, a 'Comments' tab shows up in the bottom right corner of the item's tab

4.2.6 Tasks

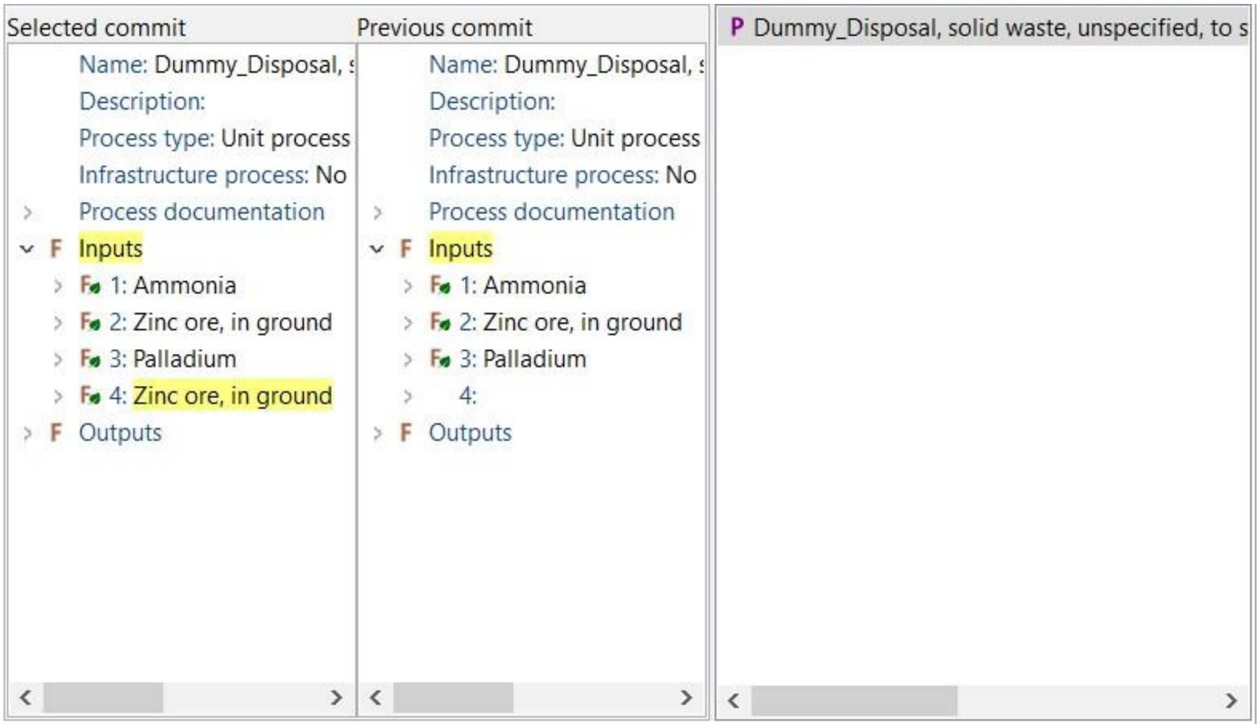
The task system can be accessed via the paper tray symbol in the top-right corner of the LCA Collaboration Server dashboard and allows creating tasks which can be assigned to *Reviewers* (Table 2-2). Proper use of the task system requires at least one user in the team to have *Reviewer* user rights (Table *2-2*).

Table 4-1 Tasks and their status

Status	Description
In progress	Awaiting release or moderation (depending on whether a user is *Editor* or *Reviewer*)
Completed	Released and/or reviewed
Cancelled	No further action is required

4.2.7 Diff utility

The 'diff utility' in openLCA is a data comparison tool to display differences between commits. It compares data sets in the local database with those in the repository of the LCA Collaboration Server. To open the diff utility right-click on a database and select *Repository > Show in history*. The diff utility supports solving conflicts by identifying differences between the local database and a version of the repository (Figure 4-13).



Selected commit	Previous commit
Name: Dummy_Disposal, s	Name: Dummy_Disposal, s
Description:	Description:
Process type: Unit process	Process type: Unit process
Infrastructure process: No	Infrastructure process: No
Process documentation	Process documentation
F Inputs	F Inputs
1: Ammonia	1: Ammonia
2: Zinc ore, in ground	2: Zinc ore, in ground
3: Palladium	3: Palladium
4: Zinc ore, in ground	4:
F Outputs	F Outputs

P Dummy_Disposal, solid waste, unspecified, to s

Figure 4-13 openLCA's Diff utility displays differences between the local version of a database and a version of the repository. In the Figure, the flow Zinc ore, in ground has been added to the process Dummy disposal, ….

4.3 Expert features

4.3.1 Additional roles

Additional roles are that of the Data Manager and User Manager. Both can be assigned as such by the **Server Administrator**.

- **Data Manager –** A data manger can manage libraries and push data to GLAD[7]
- **User Manager –** A user manager can manage users and teams

Libraries

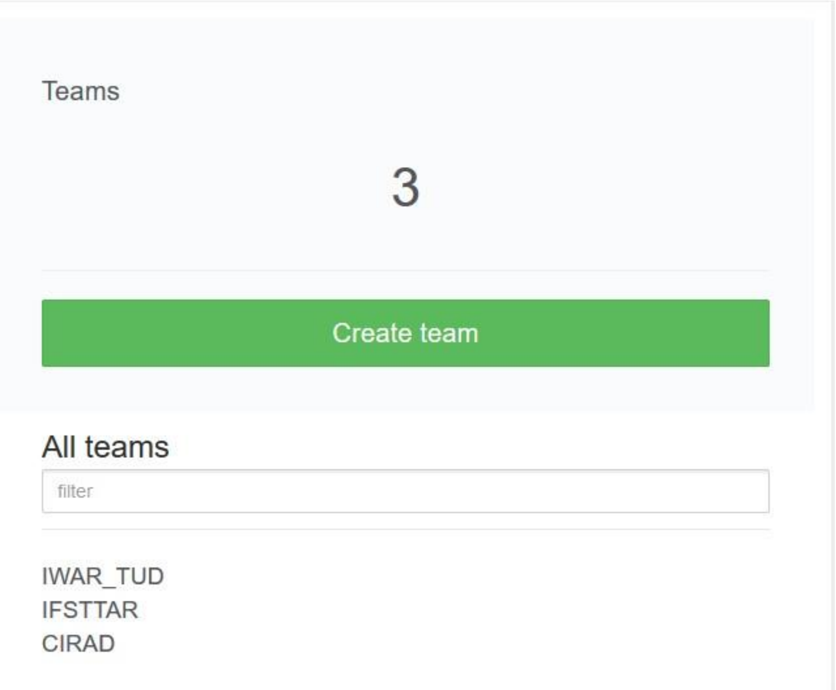
Libraries are protected data sets that require an additional confirmation to commit changes.

Create users

Users can be created by accessing the admin area via clicking on the wrench in the top-right corner of the LCA Collaboration Server dashboard.

Teams

Individual users can be combined to a team and teams can be added to groups (of repositories). Teams consist of users, and groups contain repositories. Teams can be added to a repository. Teams are created by accessing the admin area via clicking on the wrench in the top-right corner of the LCA Collaboration Server dashboard (Figure 4-14)[8].



IWAR_TUD
IFSTTAR
CIRAD

Figure 4-14 Teams are created by accessing the admin area via clicking on the wrench in the top-right corner of the LCA Collaboration Server dashboard

[7] Global LCA Data Access network (http://unep-glad.71.ecedi.fr/). Feature developed for the USDA.

[8] Please note that only admins can create teams

5 Conclusion and Outlook

The LCA Collaboration Server is an unprecedented software tool and likewise, offers unprecedented opportunities for LCA practitioners and the LCA community. We have two things in our mind and kindly ask for your support!

5.1 Building a community of best practice - A public repository for LCA models

We appreciate that, with the development of the LCA Collaboration Server, the USDA (section 1.1) gave GreenDelta the opportunity not only to develop an unprecedented piece of Software but also to enhance the capabilities and reach of LCA practitioners worldwide. This is something we appreciate, and we hope so do the LCA Collaboration Server users. We were wondering how we can pick this up and return something to the LCA community to further promote the application of LCA and Life Cycle Thinking. What GreenDelta came up with is the **vision of a public repository of freely available LCA models** under the Creative Commons license[9].

Often, it is challenging for LCA beginners to develop their first complex LCA model and to our knowledge, no global database for free LCA models exists. GreenDelta wants to address this. Therefore, **we encourage you to share your LCA model with us for use in a public open access repository**. Users will be able to access the repository directly from openLCA and import existing LCA models for educational purposes.

If this sounds interesting, please reach out: https://www.openlca.org/contact/

5.2 The LCA Collaboration Server as a nexus for LCA data sets

Currently, LCA practitioners must download LCA data sets for openLCA via the web browser[10] and import them into openLCA. Actually, this procedure is not too complicated but the LCA Collaboration Server offers an opportunity to render the import of LCA data sets into openLCA even more simple. In the long-term, GreenDelta would like to offer commercial and non-commercial LCA data sets directly via the LCA Collaboration Server.

[9] https://creativecommons.org/licenses/

[10] e.g. http://nexus.openlca.org/

6 Support and Contact

6.1 Prioritised support via the GreenDelta HelpDesk

For more information about GreenDelta's support services for openLCA and the LCA Collaboration Server, please refer to: https://www.openlca.org/helpdesk.

Support services include:

- Hosting of an LCA Collaboration Server (http://www.openlca.org/lca-collaboration-server-hosting-andservices/)
- Support in setting up an LCA Collaboration Server (contact us)
- Professional training on how to get the best out of the LCA Collaboration Server for your organisation (contact us)

6.2 Public support via ask.openLCA.org

ask.openlca.org is a question-and-answer (Q&A) website on Life Cycle Assessment and the support platform for openLCA, openLCA nexus and the LCA Collaboration Server. We encourage users to use ask.openlca.org for non-confidential support requests. However, with ask.openlca.org we are building a public knowledge platform for all LCA practitioners regardless of what software they use or LCA-related questions they may have. Don't stop contacting us via email for confidential support requests or resort to one of GreenDelta's service contracts for prioritised support via the GreenDelta HelpDesk.

6.3 Demonstration of the LCA Collaboration Server

Demonstration videos of the LCA Collaboration Server are available on the openLCA YouTube channel:

https://www.youtube.com/c/openLCA. New content is coming soon, with the release of this latest version.

6.4 Contact GreenDelta GmbH

For further inquiries please contact GreenDelta GmbH.

Tel. +49 30 48 496 – 030 Fax +49
30 48 496 – 991
gd@greendelta.com GreenDelta
GmbH

Kaiserdam 13
14057 Berlin
www.greendelta.com

Privacy

GreenDelta GmbH honours privacy and backs users in protecting confidential data. The LCA Collaboration Server is an independent application and decisions about where to host data as well as access to data are exclusively taken by its users. With the LCA Collaboration Server, users remain in full control of their data and commit to building their own sovereign IT-infrastructure for collaborative development of LCA studies in a distributed team.

Appendix

A.1 Known issues

A.1.1 Redirected requests

openLCA currently does not support handling redirected requests (e.g. reverse proxy)

A.1.2 Fetching deleted data sets

Fetching deleted data sets that were not used in the local database of the user who commits but that are used in the local database of the user who fetches will lead to broken references e.g.

1. *User A* deletes a flow that is not used in the user's local database and commits the deletion to the LCA Collaboration Server
2. *User B* added a new process in the user's local database and uses the flow as input
3. When *User B* then fetches the changes of User A. The flow is deleted locally, but the input in the process still refers to the (now) deleted flow

Workaround: If a user is aware that another user is going to fetch a deletion of a (locally) linked element. Modify any field in the element in question (e.g. add a letter in the name and remove it again and save). openLCA will track that the model has changed. When fetching the "*deletion*" a conflict will come up and the user can manually choose to keep the local element

A.1.3 Error reporting

The Collaboration Server uses *log4j2* for logging. In addition to writing server-side log files, *log4j2* can be configured to send error emails to a specific email account. These emails can either be sent from the local server directly or via an external email account. Admins will need to replace the file */WEB-INF/classes/log4j2.xml* in the built war-file. Below is an example configuration for an external email account. For more information see the *log4j2*.

- ***Note:*** If you want to directly report errors to us, please use error@greendelta.com as 'to' attribute and append your company or institution name to the 'subject' attribute.
- ***Important:*** The username and email address of the user encountering the error, will be added to the email. If you do not want this information to be send to us, please configure the logging to send the emails to your email address first and forward the email - cleared from personal information - to us.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Configuration status="WARN">
```

```
    <Appenders>
        <Console name="Console" target="SYSTEM_OUT">
  <PatternLayout pattern="%d{yy-MM-dd HH:mm:ss:SSS} %-5p %c:%L - %m%n" />
        </Console>
        <SMTP name="Mail" from="some-
account@your-company.com" to="error@your-
company.com" subject="Error on LCA Collaboration
Server" smtpPassword="the-password"
smtpUsername="some-account@your-company.com"
smtpProtocol="smtps" smtpPort="465"
smtpHost="your-email-host-address" />
</Appenders>
    <Loggers>
        <Root level="info">
            <AppenderRef ref="Console" level="info" />
            <AppenderRef ref="Mail" level="error"/>
        </Root>
    </Loggers>
</Configuration>
```

A.2 Installation of the LCA Collaboration Server

We have prepared a detailed installation guide on openLCA.org - https://www.openlca.org/lca-collaboration-server-1-3-0-installation-guide

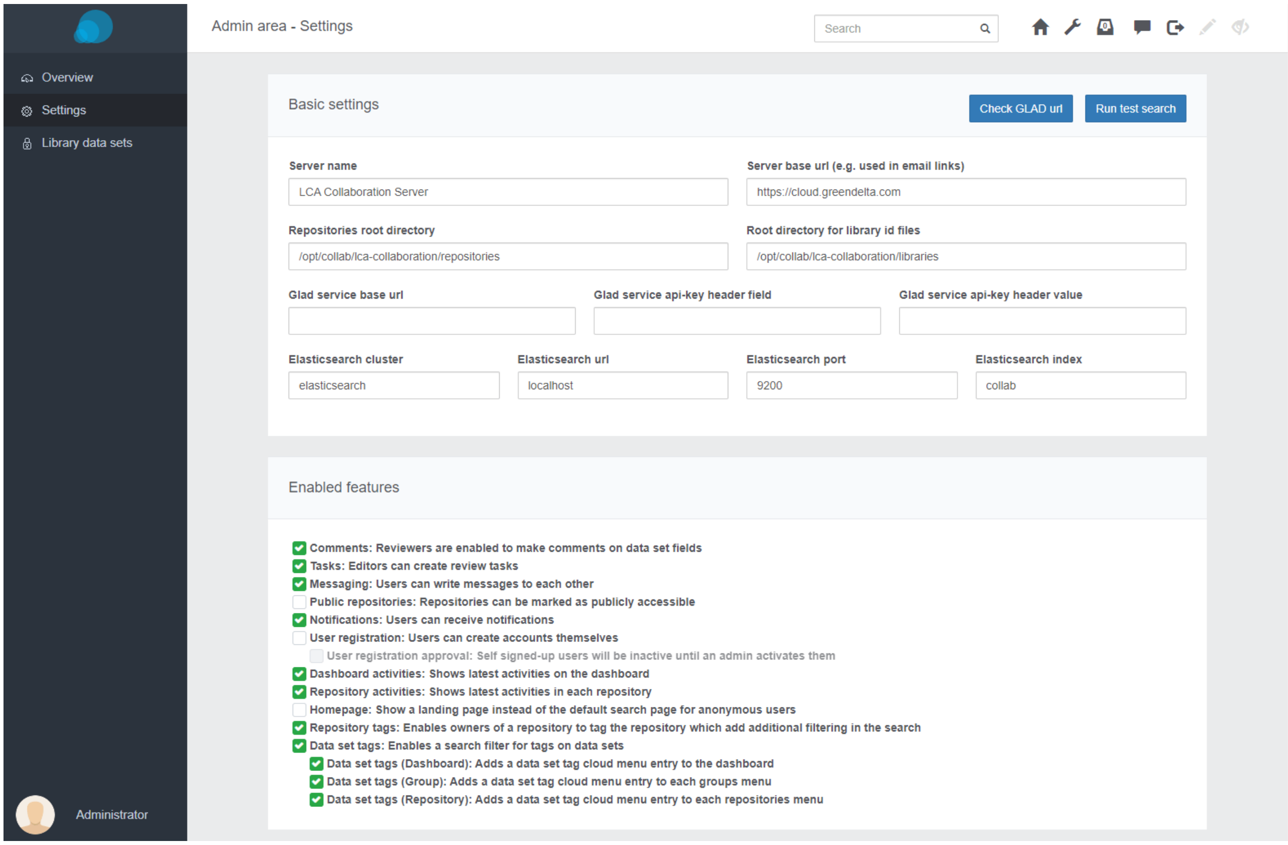
A.3 Configuration

If you want to run the application in the server root context or another context path then */lca-collaboration/*, you will need to adjust the *<basehref="/lca-collaboration/"/>* with your context path,e.g. *<basehref="/anotherpath/"/>* in the *index.html*, *login.html* and *imprint.html* files located in the root of the war file.

Before starting the application, you will need to configure the database directory, either as a system variable or as a context parameter, named app.database e.g. on Linux either set an environment variable *export app.database=/opt/collab/database* or in tomcat 8 server.xml (for each instance):

```
<Host        name="localhost"        appBase="webapps"
      unpackWARs="true" autoDeploy="true">

  <Valve className="org.apache.catalina.valves.AccessLogValve"
         directory="logs" prefix="localhost_access_log" suffix=".txt"
         pattern="%h %l %u %t &quot;%r&quot; %s %b"/>

  <Context path="/lca-collaboration">

    <Parameter name="app.database" value="/opt/collab/database"/>

  </Context>

</Host>
```

Changing settings in the web application



Basic settings

Server name: LCA Collaboration Server

Server base url (e.g. used in email links): https://cloud.greendelta.com

Repositories root directory: /opt/collab/lca-collaboration/repositories

Root directory for library id files: /opt/collab/lca-collaboration/libraries

Glad service base url:

Glad service api-key header field:

Glad service api-key header value:

Elasticsearch cluster: elasticsearch

Elasticsearch url: localhost

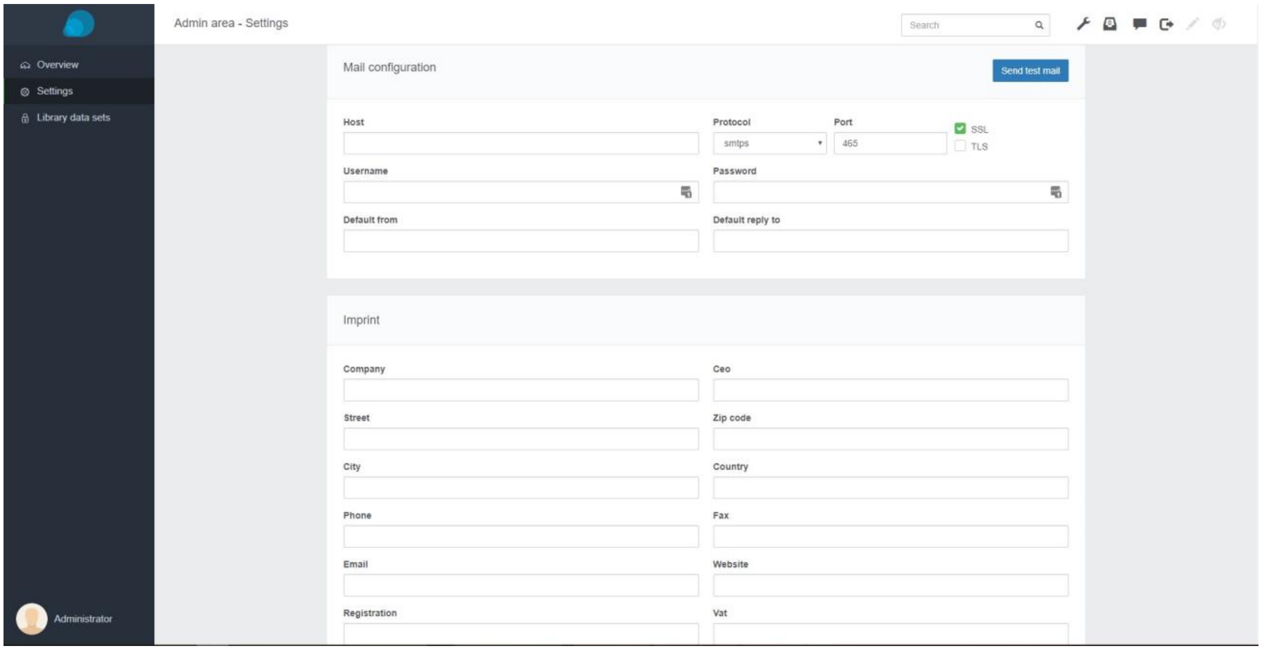
Elasticsearch port: 9200

Elasticsearch index: collab

Enabled features

- ☑ Comments: Reviewers are enabled to make comments on data set fields
- ☑ Tasks: Editors can create review tasks
- ☑ Messaging: Users can write messages to each other
- □ Public repositories: Repositories can be marked as publicly accessible
- ☑ Notifications: Users can receive notifications
- □ User registration: Users can create accounts themselves
  - □ User registration approval: Self signed-up users will be inactive until an admin activates them
- ☑ Dashboard activities: Shows latest activities on the dashboard
- ☑ Repository activities: Shows latest activities in each repository
- □ Homepage: Show a landing page instead of the default search page for anonymous users
- ☑ Repository tags: Enables owners of a repository to tag the repository which add additional filtering in the search
- ☑ Data set tags: Enables a search filter for tags on data sets
  - ☑ Data set tags (Dashboard): Adds a data set tag cloud menu entry to the dashboard
  - ☑ Data set tags (Group): Adds a data set tag cloud menu entry to each groups menu
  - ☑ Data set tags (Repository): Adds a data set tag cloud menu entry to each repositories menu

Figure A.1: Admin area basic settings and enabled features



Protocol: smtps

Port: 465

☑ SSL

□ TLS

Figure A.2: Admin area mail configuration and imprint

A.4.1 Basic settings

Servername: The Server name, used for two-factor-authentication (optional)

Server base url: The base url used when linking pages in the notification emails

Repositories root directory: In this directory, the repository data sets will be stored, this will need extended disk space, depending on the amount of data sets committed.

Root directory for library id files: In this directory, configured library data set ref ids will be stored

Glad service base url: The base url to an external GLAD service

Glad service api key: To push data set descriptors to the GLAD service, an API key is required, which can be set here.

Elasticsearch cluster: The cluster name of your elasticsearch installation (default: elasticsearch)

Elasticsearch server url: The host address of the elasticsearch installation (default: localhost)

Elasticsearch index name: The name of the elasicsearch index used for this collaboration server instance (default: lca-collaboration)

A.4.2 Enabled features

Some of the collaboration server's advanced features can be disabled, this includes: Comments, Tasks, Messaging, Public repositories, Notifications, Activities and Tags (Figure A.2 p.40).

A.4.3 Mail configuration

To be able to use notifications, you need to configure an email account to send from. You can use an existing smpt email account (Figure A.1 p.40).

A.5 Hardware and Software requirements

A.5.1 Requirements for openLCA

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

A.5.2 Requirements for the LCA Collaboration Server

Minimum:

CPU: DualCore CPU or better

RAM (Elasticsearch/LCA CS): 2GB/2GB[11]

Disk: HDD (the required disk space depends mostly on the amount of data you want to be able to host; the application itself only requires <100MB) **Recommended:**

CPU: QuadCore CPU or better

RAM (Elasticsearch/LCA CS): 4GB/8GB

Disk: SSD[4]

Software requirements

The LCA Collaboration Server is a web application based on Java Servlet. We recommend the use of Tomcat 8 in combination with Java 8. Most testing has been done under use of this configuration (for both Windows 10 and Ubuntu/Debian Linux).

The LCA Collaboration Server depends on Elasticsearch. Since version 6 it does not support JVM internal local nodes. Therefore, Elasticsearch must be installed and run separately.

[11] Each Elasticsearch and the LCA CS will need separate RAM. The requirements depend mostly on the number of concurrent requests for fetching and committing data.

GreenDelta GmbH

Kaiserdamm 13
14057 Berlin
Germany
www.greendelta.com

Tel. +49 30 48 496 – 030
Fax +49 30 48 496 – 991
gd@greendelta.com

GreenDelta
software / data / know-how