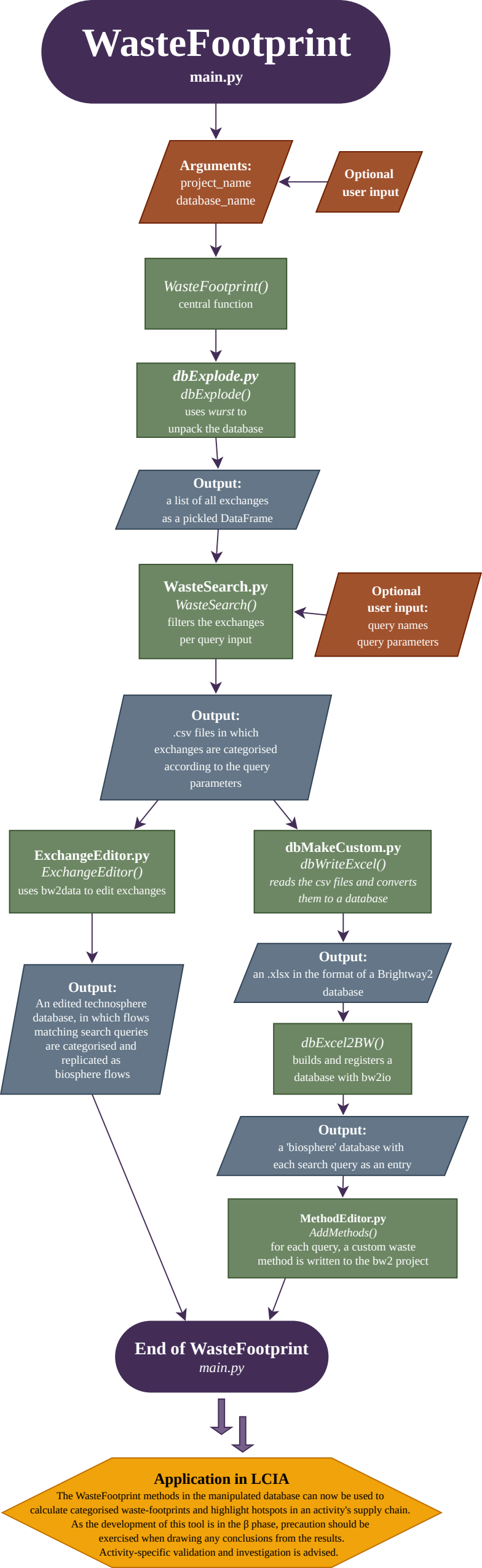
WasteFootprint
main.py
Arguments:
project_name
database_name
Optional
user input
WasteFootprint()
central function
dbExplode.py
dbExplode()
uses wurst to
unpack the database
Output:
a list of all exchanges
as a pickled DataFrame
WasteSearch.py
WasteSearch()
filters the exchanges
per query input
Optional
user input:
query names
query parameters
Output:
.csv files in which
exchanges are categorised
according to the query
parameters
ExchangeEditor.py
ExchangeEditor()
uses bw2data to edit exchanges
dbMakeCustom.py
dbWriteExcel()
reads the csv files and converts
them to a database
Output:
An edited technosphere
database, in which flows
matching search queries
are categorised and
replicated as
biosphere flows
Output:
an .xlsx in the format of a Brightway2
database
dbExcel2BW()
builds and registers a
database with bw2io
Output:
a 'biosphere' database with
each search query as an entry
MethodEditor.py
AddMethods()
for each query, a custom waste
method is written to the bw2 project
End of WasteFootprint
main.py
Application in LCIA
The WasteFootprint methods in the manipulated database can now be used to
calculate categorised waste-footprints and highlight hotspots in an activity's supply chain.
As the development of this tool is in the β phase, precaution should be
exercised when drawing any conclusions from the results.
Activity-specific validation and investigation is advised.