State of Oregon Department of Environmental Quality

Example applications of the Waste Impact Calculator

Martin J. Brown, Martin.Brown@state.or.us
Version October 10, 2020

Introduction

The Waste Impact Calculator (WIC) is a framework for estimating the life cycle environmental impacts of solid waste streams, and projecting the life cycle impacts associated with different solid waste management scenarios. See *Technical Overview of the Waste Impact Calculator* for complete background.

The WIC framework can be exercised in many ways. This document shows several examples that use the R language. First, there are screenshots from an interactive web app created with R's "shiny" package. These screenshots provide a preview of the diversity of graphical output that might be used to express WIC results – and hopefully, to increase the user's understanding of the life cycle impacts associated with solid waste.

The code for the app is not given here, as it is still under development. However, simplified versions of many of the app's displays are coded out in full in the next example: a simple but complete WIC analysis using R Markdown (page 8). The R Markdown format here serves as a kind of analytical notebook, combining explanatory text, runnable R code, graphical output, and links to internet resources.

Screenshots from an interactive web app

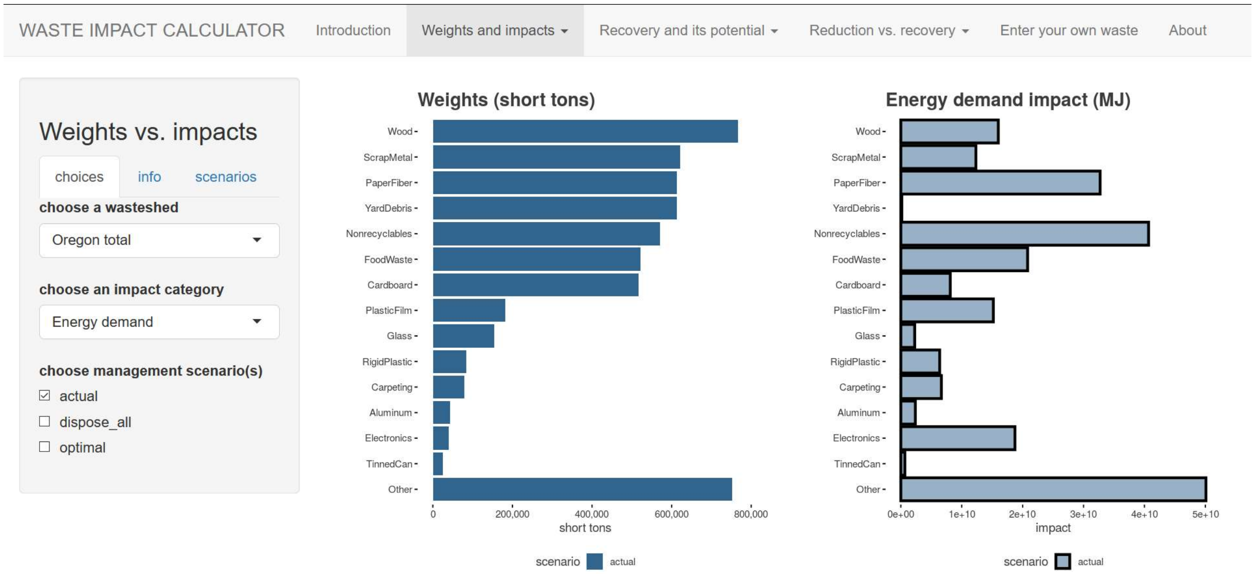
Oregon DEQ has used the WIC framework to create a web app that reports on recent solid waste statistics for Oregon counties, and the estimated life cycle impacts associated with them. It also helps users project the impact changes that might be associated with changing waste management.

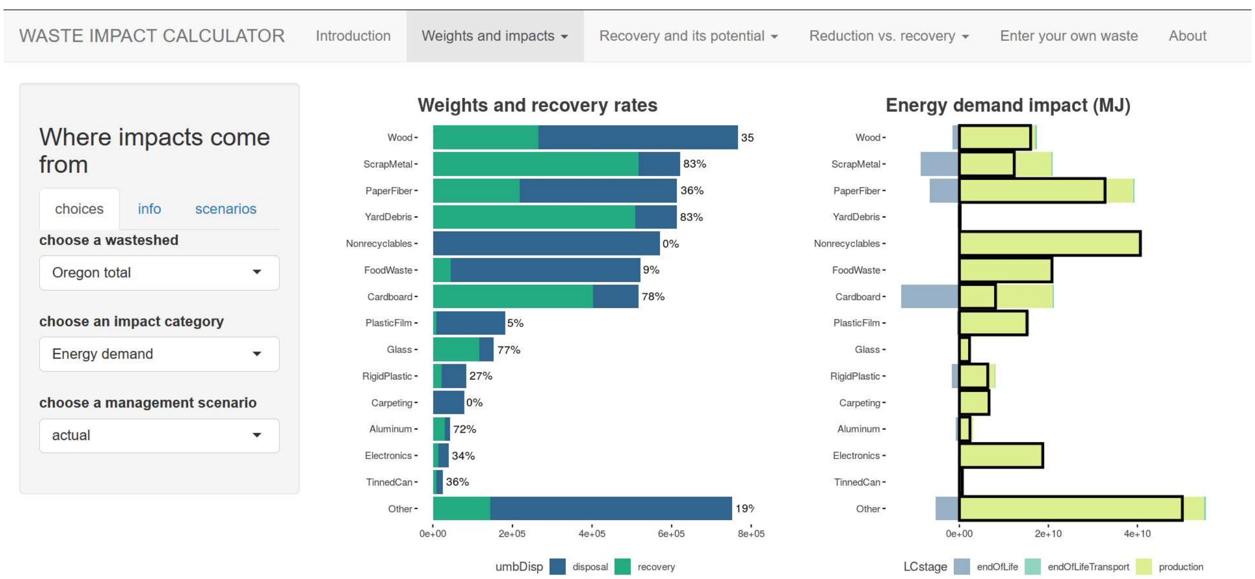
The app includes a pre-loaded massProfiles table representing Oregon "wastesheds" (counties), so the user does not need to labor to create or format one. This massProfiles table is extensive. For each "wasteshed", it describes a solid waste stream under three scenarios: "actual" recent data, "dispose_all" (a zero-recycling scenario), and "optimal" (where recovery and disposal are mixed in a way that minimizes impacts).

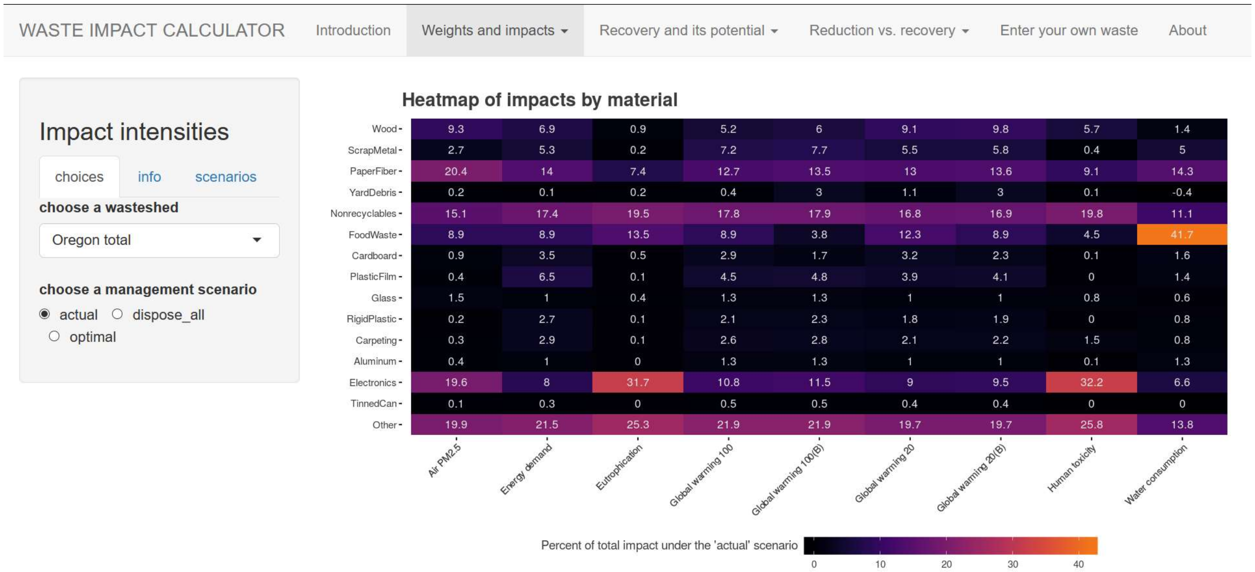
When combined with impactFactors, the resulting impactsInDetail table can be displayed in diverse ways to help users understand how the weight of solid waste, and options for managing it, relate to estimated life cycle impacts. The charts help the user investigate and understand topics like:

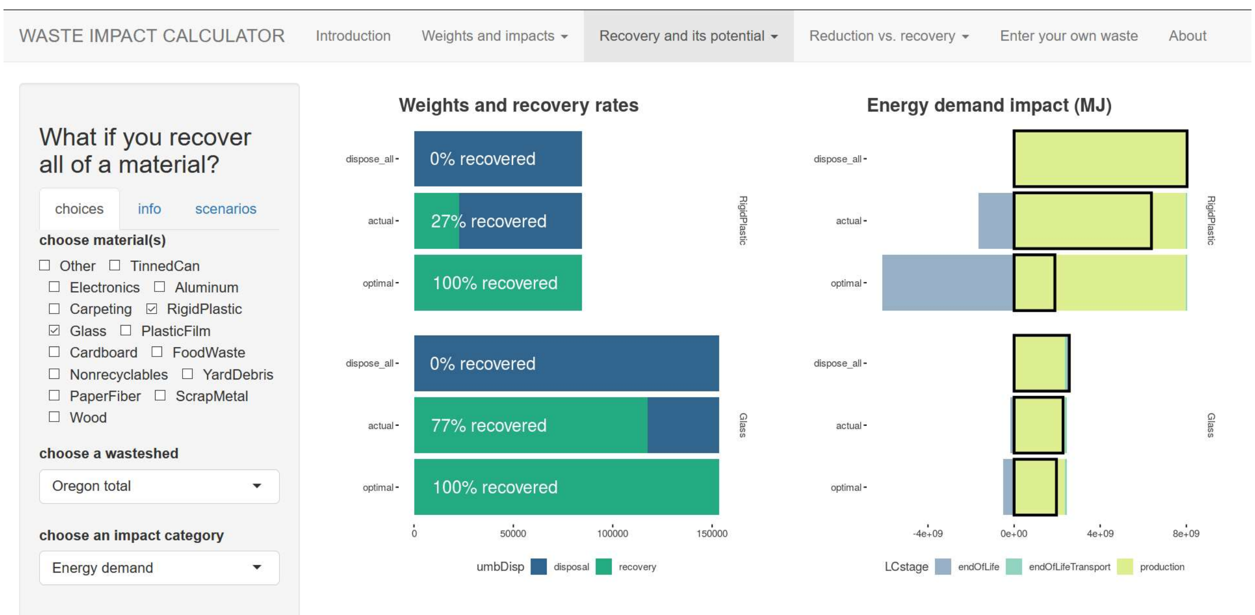
- Which materials represent the greatest weights in the waste stream, and the greatest impacts;
- The correlation between material weight and estimated life cycle impact (which is usually poor); and
- The potential impact reductions associated with increased recovery (which often are less dramatic one might imagine).

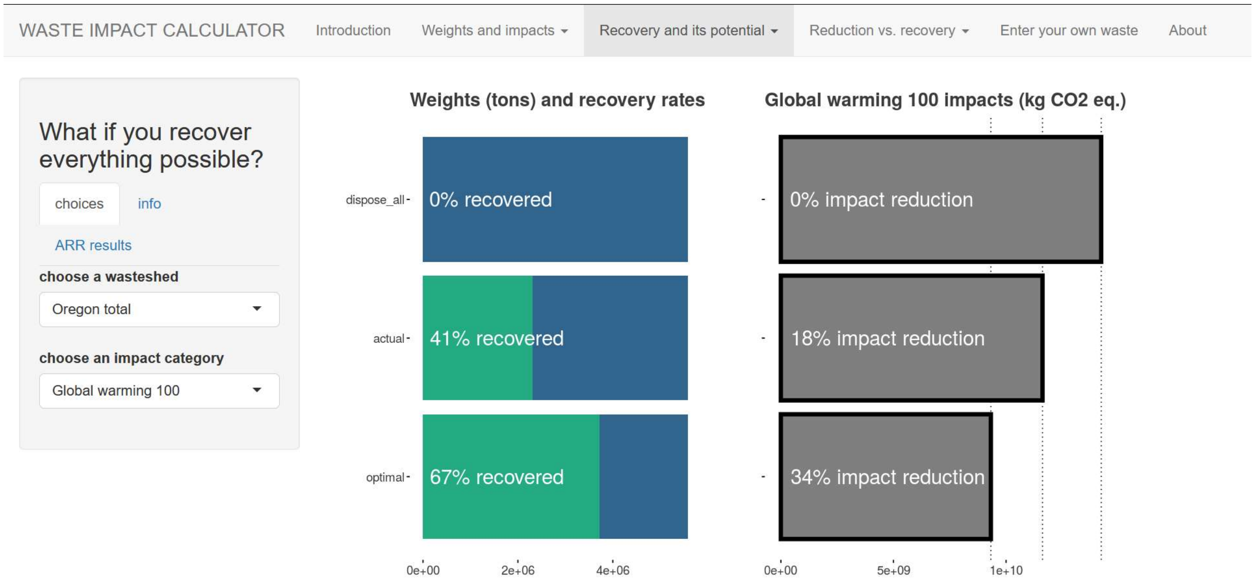
One section of the app allows the user to go beyond the pre-loaded massProfiles table and enter their own data – though this is best suited for quick analyses involving a few materials.


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Weights vs. impacts
choices
info
scenarios
choose a wasteshed
Oregon total
choose an impact category
Energy demand
choose management scenario(s)
actual
dispose_all
optimal
Weights (short tons)
Wood
ScrapMetal
PaperFiber
YardDebris
Nonrecyclables
FoodWaste
Cardboard
PlasticFilm
Glass
RigidPlastic
Carpeting
Aluminum
Electronics
TinnedCan
Other
0
200,000
400,000
600,000
800,000
short tons
scenario
actual
Energy demand impact (MJ)
0e+00
1e+10
2e+10
3e+10
4e+10
5e+10
impact
scenario
actual

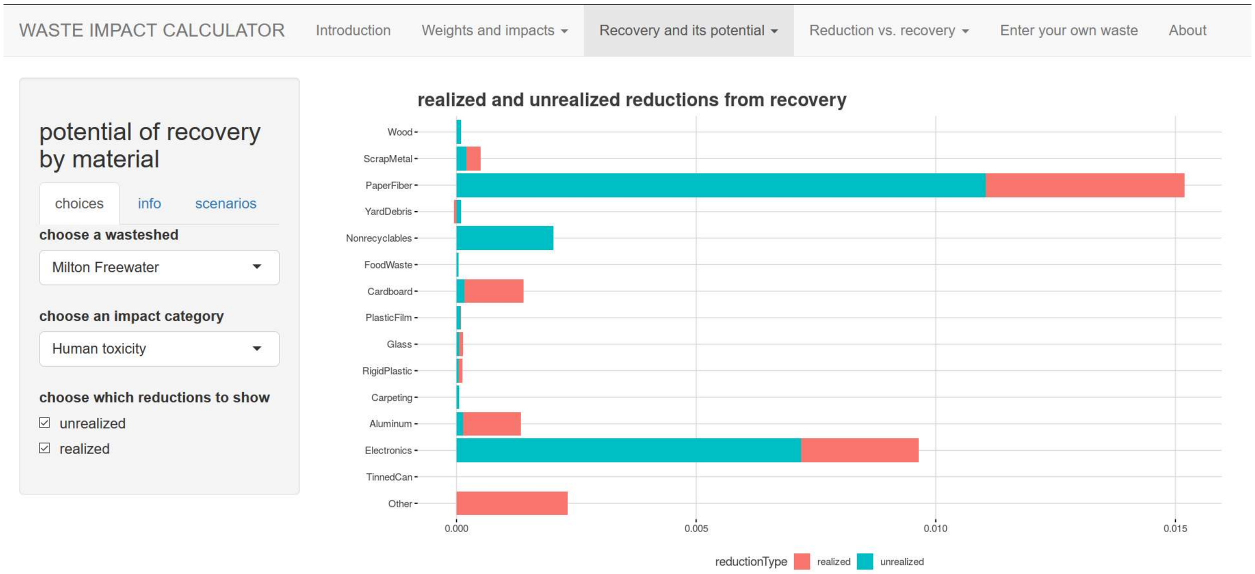

WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Where impacts come from
choices
info
scenarios
choose a wasteshed
Oregon total
choose an impact category
Energy demand
choose a management scenario
actual
Weights and recovery rates
Wood 35
ScrapMetal 83%
PaperFiber 36%
YardDebris 83%
Nonrecyclables 0%
FoodWaste 9%
Cardboard 78%
PlasticFilm 5%
Glass 77%
RigidPlastic 27%
Carpeting 0%
Aluminum 72%
Electronics 34%
TinnedCan 36%
Other 19
0e+00
2e+05
4e+05
6e+05
8e+05
umbDisp
disposal
recovery
Energy demand impact (MJ)
0e+00
2e+10
4e+10
LCstage
endOfLife
endOfLifeTransport
production

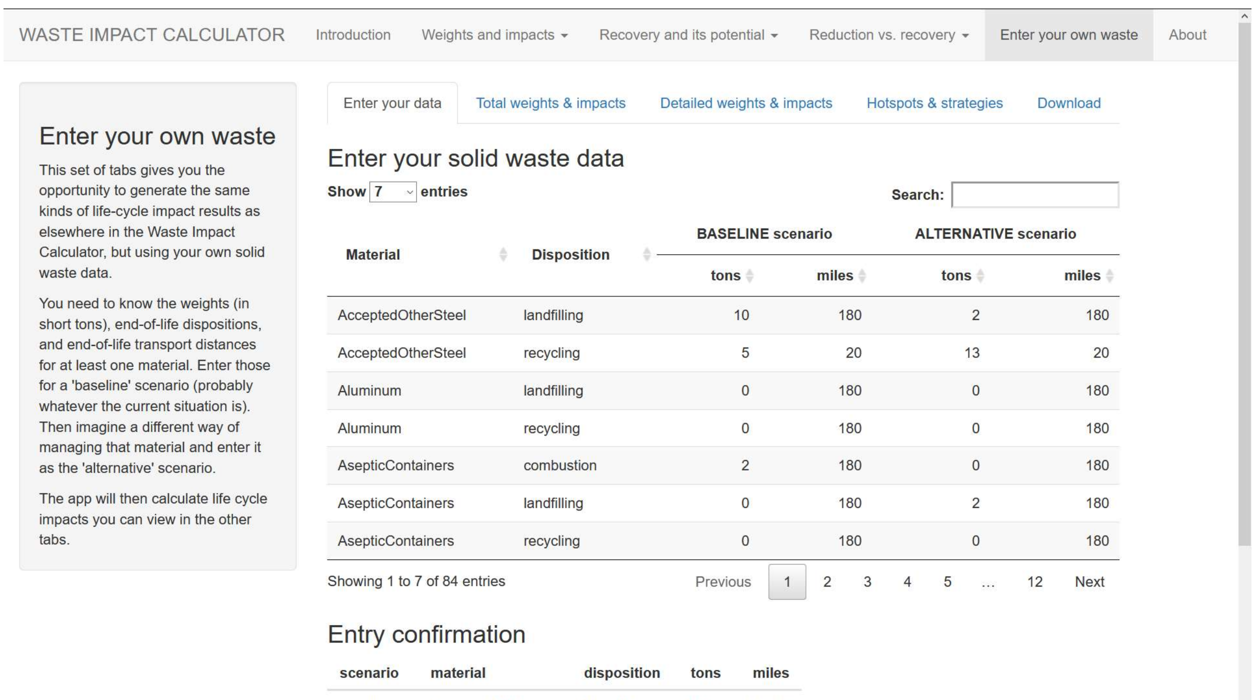

WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Impact intensities
choices
info
scenarios
choose a wasteshed
Oregon total
choose a management scenario
actual
dispose_all
optimal
Heatmap of impacts by material
Wood 9.3 6.9 0.9 5.2 6 9.1 9.8 5.7 1.4
ScrapMetal 2.7 5.3 0.2 7.2 7.7 5.5 5.8 0.4 5
PaperFiber 20.4 14 7.4 12.7 13.5 13 13.6 9.1 14.3
YardDebris 0.2 0.1 0.2 0.4 3 1.1 3 0.1 -0.4
Nonrecyclables 15.1 17.4 19.5 17.8 17.9 16.8 16.9 19.8 11.1
FoodWaste 8.9 8.9 13.5 8.9 3.8 12.3 8.9 4.5 41.7
Cardboard 0.9 3.5 0.5 2.9 1.7 3.2 2.3 0.1 1.6
PlasticFilm 0.4 6.5 0.1 4.5 4.8 3.9 4.1 0 1.4
Glass 1.5 1 0.4 1.3 1.3 1 1 0.8 0.6
RigidPlastic 0.2 2.7 0.1 2.1 2.3 1.8 1.9 0 0.8
Carpeting 0.3 2.9 0.1 2.6 2.8 2.1 2.2 1.5 0.8
Aluminum 0.4 1 0 1.3 1.3 1 1 0.1 1.3
Electronics 19.6 8 31.7 10.8 11.5 9 9.5 32.2 6.6
TinnedCan 0.1 0.3 0 0.5 0.5 0.4 0.4 0 0
Other 19.9 21.5 25.3 21.9 21.9 19.7 19.7 25.8 13.8
Air PM2.5
Energy demand
Eutrophication
Global warming 100
Global warming 100(B)
Global warming 20
Global warming 20(B)
Human toxicity
Water consumption
Percent of total impact under the 'actual' scenario
0
10
20
30
40


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
What if you recover all of a material?
choices
info
scenarios
choose material(s)
Other
TinnedCan
Electronics
Aluminum
Carpeting
RigidPlastic
Glass
PlasticFilm
Cardboard
FoodWaste
Nonrecyclables
YardDebris
PaperFiber
ScrapMetal
Wood
choose a wasteshed
Oregon total
choose an impact category
Energy demand
Weights and recovery rates
dispose_all
actual
optimal
0% recovered
27% recovered
100% recovered
RigidPlastic
0% recovered
77% recovered
100% recovered
Glass
0
50000
100000
150000
umbDisp
disposal
recovery
Energy demand impact (MJ)
dispose_all
actual
optimal
RigidPlastic
Glass
-4e+09
0e+00
4e+09
8e+09
LCstage
endOfLife
endOfLifeTransport
production


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
What if you recover everything possible?
choices
info
ARR results
choose a wasteshed
Oregon total
choose an impact category
Global warming 100
Weights (tons) and recovery rates
dispose_all
actual
optimal
0% recovered
41% recovered
67% recovered
0e+00
2e+06
4e+06
Global warming 100 impacts (kg CO2 eq.)
0% impact reduction
18% impact reduction
34% impact reduction
0e+00
5e+09
1e+10

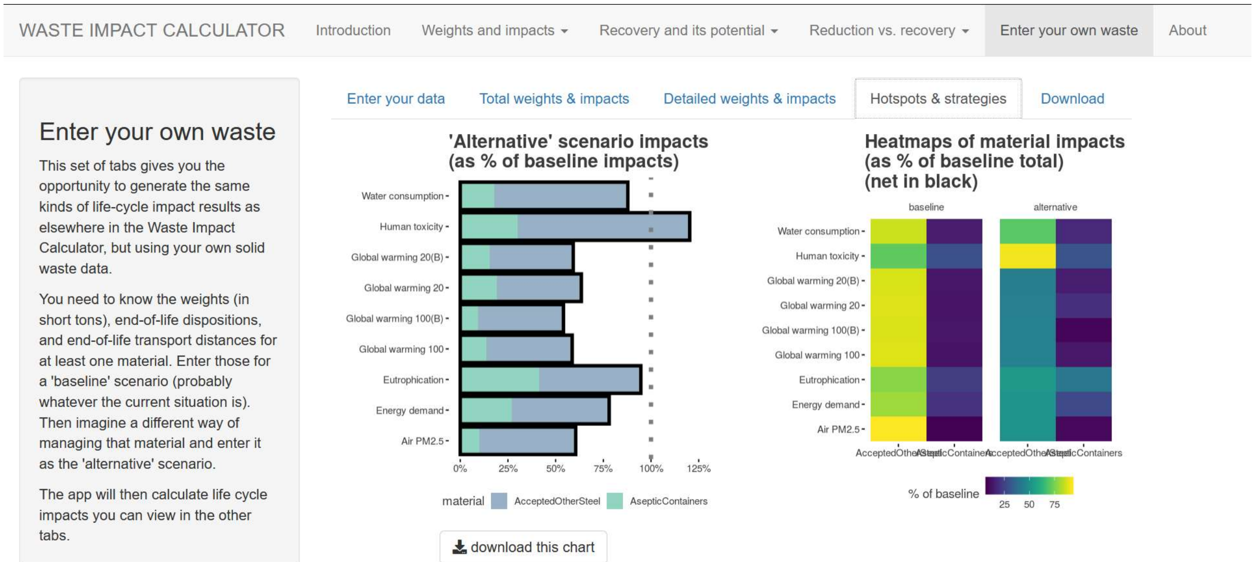

WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Normalized impacts
choices
info
scenarios
choose a wasteshed
Oregon total
Normalized impacts of the 'dispose all' scenario (as % of 'actual' scenario)
Normalized impacts of the 'optimal recovery' scenario (as % of 'actual' scenario)
Water consumption
Human toxicity
Global warming 20(B)
Global warming 20
Global warming 100(B)
Global warming 100
Eutrophication
Energy demand
Air PM2.5
0%
50%
100%
0%
25%
50%
75%
100%


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
potential of recovery by material
choices
info
scenarios
choose a wasteshed
Milton Freewater
choose an impact category
Human toxicity
choose which reductions to show
unrealized
realized
realized and unrealized reductions from recovery
Wood
ScrapMetal
PaperFiber
YardDebris
Nonrecyclables
FoodWaste
Cardboard
PlasticFilm
Glass
RigidPlastic
Carpeting
Aluminum
Electronics
TinnedCan
Other
0.000
0.005
0.010
0.015
reductionType
realized
unrealized


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Enter your own waste
This set of tabs gives you the opportunity to generate the same kinds of life-cycle impact results as elsewhere in the Waste Impact Calculator, but using your own solid waste data.
You need to know the weights (in short tons), end-of-life dispositions, and end-of-life transport distances for at least one material. Enter those for a 'baseline' scenario (probably whatever the current situation is). Then imagine a different way of managing that material and enter it as the 'alternative' scenario.
The app will then calculate life cycle impacts you can view in the other tabs.
Enter your data
Total weights & impacts
Detailed weights & impacts
Hotspots & strategies
Download
Enter your solid waste data
Show 7 entries
Search:
Material
Disposition
BASELINE scenario
ALTERNATIVE scenario
tons
miles
tons
miles
AcceptedOtherSteel landfilling 10 180 2 180
AcceptedOtherSteel recycling 5 20 13 20
Aluminum landfilling 0 180 0 180
Aluminum recycling 0 180 0 180
AsepticContainers combustion 2 180 0 180
AsepticContainers landfilling 0 180 2 180
AsepticContainers recycling 0 180 0 180
Showing 1 to 7 of 84 entries
Previous 1 2 3 4 5 … 12 Next
Entry confirmation
scenario
material
disposition
tons
miles
baseline AcceptedOtherSteel landfilling 10.00 180.00


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Enter your own waste
This set of tabs gives you the opportunity to generate the same kinds of life-cycle impact results as elsewhere in the Waste Impact Calculator, but using your own solid waste data.
You need to know the weights (in short tons), end-of-life dispositions, and end-of-life transport distances for at least one material. Enter those for a 'baseline' scenario (probably whatever the current situation is). Then imagine a different way of managing that material and enter it as the 'alternative' scenario.
The app will then calculate life cycle impacts you can view in the other tabs.
Enter your data
Total weights & impacts
Detailed weights & impacts
Hotspots & strategies
Download
Weight (short tons)
baseline
alternative
0
5
10
15
disposition
combustion
landfilling
recycling
Energy demand (MJ)
baseline
alternative
0e+00
1e+05
2e+05
3e+05
scenario
alternative
baseline
download this chart
download this chart
Choose an impact category
Energy demand


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Enter your own waste
This set of tabs gives you the opportunity to generate the same kinds of life-cycle impact results as elsewhere in the Waste Impact Calculator, but using your own solid waste data.
You need to know the weights (in short tons), end-of-life dispositions, and end-of-life transport distances for at least one material. Enter those for a 'baseline' scenario (probably whatever the current situation is). Then imagine a different way of managing that material and enter it as the 'alternative' scenario.
The app will then calculate life cycle impacts you can view in the other tabs.
Enter your data
Total weights & impacts
Detailed weights & impacts
Hotspots & strategies
Download
Detailed weights (short tons)
baseline
alternative
AcceptedOtherSteel
AsepticContainers
0
5
10
15
disposition
combustion
landfilling
recycling
Air PM2.5 (kg PM2.5 eq.)
-5
0
5
10
LCstage
endOfLife
endOfLifeTransport
production
choose an impact category
Air PM2.5


WASTE IMPACT CALCULATOR
Introduction
Weights and impacts
Recovery and its potential
Reduction vs. recovery
Enter your own waste
About
Enter your data
Total weights & impacts
Detailed weights & impacts
Hotspots & strategies
Download
Enter your own waste
This set of tabs gives you the opportunity to generate the same kinds of life-cycle impact results as elsewhere in the Waste Impact Calculator, but using your own solid waste data.
You need to know the weights (in short tons), end-of-life dispositions, and end-of-life transport distances for at least one material. Enter those for a 'baseline' scenario (probably whatever the current situation is). Then imagine a different way of managing that material and enter it as the 'alternative' scenario.
The app will then calculate life cycle impacts you can view in the other tabs.
'Alternative' scenario impacts (as % of baseline impacts)
Water consumption
Human toxicity
Global warming 20(B)
Global warming 20
Global warming 100(B)
Global warming 100
Eutrophication
Energy demand
Air PM2.5
0%
25%
50%
75%
100%
125%
material
AcceptedOtherSteel
AsepticContainers
download this chart
Heatmaps of material impacts (as % of baseline total) (net in black)
baseline
alternative
% of baseline
25
50
75

Simple WIC analysis using R Markdown

by Martin Brown, *Martin.Brown@state.or.us*

Introduction

This RMarkdown document provides an example of using the Waste Impact Calculator (WIC) framework to estimate the life cycle environmental impacts associated with solid waste streams, and to compare the estimated environmental impacts associated with various waste management scenarios.

Along the way, this document provides pointers on how to properly combine, filter, and summarize WIC's data tables, so that the final results actually represent the scenarios you are interested in evaluating. Before programming your own analyses, you should have a clear understanding of:

- how WIC calculates impacts for individual life cycle stages; and
- how file structures and joining commands (for example *left_join* and *full_join* in R's dplyr package) can create sets of impacts that represent the materials life cycle.

Intended audience

This document is oriented towards more technical users – those familiar with statistical or database operations. This document assumes a beginner-to-intermediate familiarity with the R language, especially the packages "base," "dplyr", and "ggplot2". However, once WIC's principles are understood, the WIC framework does not specifically require R.

Less technical users may want to spend some time with the *interactive web app version of WIC*, which does not require programming. This app may be useful for more technical users as well, as it illustrates a variety of the types of output the WIC framework can create.

Prerequisites

"Technical overview of the Waste Impact Calculator" describes the the purpose, limitations, and basic operation of WIC. It also documents the meaning of all the fields in each of WIC's data tables – information that will, for the most part, not be repeated here.

Outline

This document compares the life cycle environmental impacts associated with 3 management scenarios for fictional municipal wastestreams. The management scenarios are:

- "baseline" (representing an "actual" recent mix of recycling and disposal);
- "eliminate_food_waste" (food waste is assumed to be zero); and

- "recover_nearly_all" (maximize recycling and composting)

The analysis will proceed in this order:

- preparing the R workspace
- loading in the two source data tables, massProfiles and impactFactors
- calculating impacts and creating the master results data table, impactsInDetail
- checking for internal consistency of impactsInDetail
- creating assorted tabular and graphic results based on impactsInDetail

Preparing the R workspace

```
# loading packages useful for the analysis
library(tidyverse)  # many useful functions for data management

## -- Attaching packages --------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(ggthemes)   # some themes for plotting
library(scales)     # useful functions for labeling charts

##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

library(knitr)      # helps generate formatted output of various kinds
library(rmarkdown)  # converts RMarkdown documents to other formats
library(viridis)    # nice & accessible color schemes

## Loading required package: viridisLite

##
## Attaching package: 'viridis'

## The following object is masked from 'package:scales':
##
##     viridis_pal
```

```
library(svglite)    # helps write charts to SVG files
```

Loading *massProfiles* and *impactFactors*

As you recall from *"Technical overview of the Waste Impact Calculator"*, the massProfiles table describes waste management scenarios by listing, in detail, the mass of each waste material going to specific end-of-life dispositions (e.g. landfilling, recycling), from areas of interest ("wastesheds"), as well as (optionally) setting transport distances for those end-of-life treatments. Different waste management ideas, or "scenarios", are expressed as different numbers of tons going to different dispositions, and (optionally) different transport distances.

Here we load the massProfiles table into an R data frame, and print it out in a formatted table.

```
# loading the mass profile data into an R data frame
massProfiles <-
  read.csv(
    file = "source_data/mass_profiles.csv",
    header = TRUE,
    stringsAsFactors = FALSE
  )
# a formatted printout
kable(
  massProfiles,
  caption="a simple massProfiles table for 3 scenarios and 2 materials"
)
```

a simple massProfiles table for 3 scenarios and 2 materials

wasteshed	material	disposition	umbDisp	scenario	tons	miles
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180
City A	Electronics	landfilling	disposal	recover_nearly_all	0.000000e+00	180
City A	Electronics	recycling	recovery	recover_nearly_all	5.194783e+03	180
City A	FoodWaste	anaerobicDigestion	recovery	recover_nearly_all	4.932163e+04	180
City A	FoodWaste	combustion	disposal	recover_nearly_all	0.000000e+00	180
City A	FoodWaste	combustion	recovery	recover_nearly_all	0.000000e+00	180
City A	FoodWaste	composting	recovery	recover_nearly_all	0.000000e+00	180
City A	FoodWaste	landfilling	disposal	recover_nearly_all	0.000000e+00	180

Here we list the fields in massProfiles:

```r
str(massProfiles)
```

```
## 'data.frame':    21 obs. of  7 variables:
##  $ wasteshed  : chr  "City A" "City A" "City A" "City A" ...
##  $ material   : chr  "Electronics" "Electronics" "FoodWaste" "FoodWaste" .
..
##  $ disposition: chr  "landfilling" "recycling" "anaerobicDigestion" "combu
stion" ...
##  $ umbDisp    : chr  "disposal" "recovery" "recovery" "disposal" ...
##  $ scenario   : chr  "baseline" "baseline" "baseline" "baseline" ...
##  $ tons       : num  2.83e+03 2.37e+03 0.00 1.86e-02 1.45e+03 ...
##  $ miles      : int  180 180 180 180 180 180 180 180 180 180 ...
```

As you recall, *tons* is the critical variable. This is a mass of some waste material, in short tons. All the other variables serve to identify or qualify where the *tons* came from, what kind of management they represent, what life cycle stage they represent, etc.

If desired, we can summarise weight-based waste statistics for each scenario from massProfiles. For example, tere is the number of tons of waste for each umbDisp within each scenario:

```r
massProfiles %>%
  group_by(scenario, umbDisp) %>%
  summarize(tons=sum(tons)) %>%
  kable()
```

```
## `summarise()` regrouping output by 'scenario' (override with `.groups` arg
ument)
```

scenario	umbDisp	tons
baseline	disposal	49129.534
baseline	recovery	5386.883
eliminate_food_waste	disposal	2827.717
eliminate_food_waste	recovery	2367.066
recover_nearly_all	disposal	0.000
recover_nearly_all	recovery	54516.417

You could do a different analysis to generate a list of the materials present in the waste stream (across all scenarios).

```r
materialsObserved <-
  massProfiles %>%
  group_by(material) %>%
  summarise(summedTons = sum(tons, na.rm=TRUE)) %>%
  ungroup()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
print(materialsObserved)
```

```
## # A tibble: 2 x 2
##   material    summedTons
##   <chr>            <dbl>
```

```
## 1 Electronics     15584.
## 2 FoodWaste       98643.

rm(materialsObserved)
```

As you recall, the tons, dispositions, and miles listed in massProfiles are the data that distinguish solid waste management scenarios. Here is an example of how massProfile information expresses the difference between solid waste management scenarios, “baseline” and “eliminate_food_waste”:

```
massProfiles %>%
  filter(
    scenario=="baseline" | scenario=="eliminate_food_waste",
    material=="Electronics" | material=="FoodWaste",
    wasteshed=="City A"
    ) %>%
  kable()
```

wasteshed	material	disposition	umbDisp	scenario	tons	miles
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180

Note how the weights and dispositions for electronics are the same between scenarios, but in the “eliminate_food_waste” scenario food waste has been reduced to zero tons.

WIC’s other source data table is impactFactors. It contains environmental impact magnitudes for standard weights of solid waste materials, classified by disposition and life cycle stage.

Here we load the impactFactors table into an R data frame and print it out in a formatted table. The impactFactors table is typically thousands of records long, but for brevity in this example analysis, we filter it to only 2 materials and 2 impact categories.

```
# Loading the impact factor data into an R data frame
impactFactors <-
  read.csv(
    file = "source_data/impact_factors.csv",
    header = TRUE,
    stringsAsFactors = FALSE   #
  ) %>%
  # limiting it to two impact categories and two materials
  filter(
```

```
    (impactCategory== "Energy demand" |
       impactCategory=="Water consumption") &
      (material=="Electronics" | material=="FoodWaste")
  ) %>%
  # sorting it for easier reading
  arrange(impactCategory, material, LCstage, disposition)
# a formatted printout
kable(impactFactors)
```

material	LCstage	disposition	impactCategory	impactUnits	impliedMiles	impactFactor
Electronics	endOfLife	landfilling	Energy demand	MJ	180	832.8763
Electronics	endOfLife	recycling	Energy demand	MJ	180	-13408.5010
Electronics	endOfLifeTransport	landfilling	Energy demand	MJ	180	603.3414
Electronics	endOfLifeTransport	recycling	Energy demand	MJ	180	603.3414
Electronics	production	production	Energy demand	MJ	180	468130.3713
FoodWaste	endOfLife	anaerobicDigestion	Energy demand	MJ	180	-5592.4907
FoodWaste	endOfLife	combustion	Energy demand	MJ	180	-574.1139
FoodWaste	endOfLife	composting	Energy demand	MJ	180	-543.1128
FoodWaste	endOfLife	landfilling	Energy demand	MJ	180	319.6540
FoodWaste	endOfLifeTransport	anaerobicDigestion	Energy demand	MJ	180	603.4704
FoodWaste	endOfLifeTransport	combustion	Energy demand	MJ	180	603.4704
FoodWaste	endOfLifeTransport	composting	Energy demand	MJ	180	603.4704
FoodWaste	endOfLifeTransport	landfilling	Energy demand	MJ	180	603.4704
FoodWaste	production	production	Energy demand	MJ	180	39043.8618
Electronics	endOfLife	landfilling	Water consumption	kg	180	216.3978
Electronics	endOfLife	recycling	Water consumption	kg	180	-8811.4635
Electronics	endOfLifeTransport	landfilling	Water consumption	kg	180	108.2953
Electronics	endOfLifeTransport	recycling	Water consumption	kg	180	108.2953
Electronics	production	production	Water consumption	kg	180	271442.1168
FoodWaste	endOfLife	anaerobicDigestion	Water consumption	kg	180	1230.1352
FoodWaste	endOfLife	combustion	Water consumption	kg	180	-175.0662
FoodWaste	endOfLife	composting	Water consumption	kg	180	158.5119
FoodWaste	endOfLife	landfilling	Water consumption	kg	180	-250.7538
FoodWaste	endOfLifeTransport	anaerobicDigestion	Water consumption	kg	180	108.3185
FoodWaste	endOfLifeTransport	combustion	Water consumption	kg	180	108.3185
FoodWaste	endOfLifeTransport	composting	Water consumption	kg	180	108.3185
FoodWaste	endOfLifeTransport	landfilling	Water consumption	kg	180	108.3185
FoodWaste	production	production	Water consumption	kg	180	132157.0933

The fields making up impactFactors can be viewed as:

```
str(impactFactors)

## 'data.frame':    28 obs. of  7 variables:
##  $ material      : chr  "Electronics" "Electronics" "Electronics" "Electro
nics" ...
##  $ LCstage       : chr  "endOfLife" "endOfLife" "endOfLifeTransport" "endO
fLifeTransport" ...
##  $ disposition   : chr  "landfilling" "recycling" "landfilling" "recycling
" ...
##  $ impactCategory: chr  "Energy demand" "Energy demand" "Energy demand" "E
nergy demand" ...
##  $ impactUnits   : chr  "MJ" "MJ" "MJ" "MJ" ...
##  $ impliedMiles  : int  180 180 180 180 180 180 180 180 180 180 ...
##  $ impactFactor  : num  833 -13409 603 603 468130 ...
```

As you recall, the critical field in this table is *impactFactor*. This number expresses an environmental impact for a particular tonnage of a particular material in a particular life cycle stage. All the other variables in each record identify or qualify the impact factor somehow – e.g. name the material, label its units, etc.

There should be EXACTLY one record for each combination of material, life cycle stage, disposition, and impactCategory of interest. Material and disposition names must be identical in spelling and case to the material and disposition names in the massProfiles table.

If desired, we can produce summaries of records in the impactFactors table. E.g., Here are the numbers of records representing each disposition and LCstage:

```
table(impactFactors$disposition, impactFactors$LCstage)

##
##                     endOfLife endOfLifeTransport production
##   anaerobicDigestion         2                  2          0
##   combustion                 2                  2          0
##   composting                 2                  2          0
##   landfilling                4                  4          0
##   production                 0                  0          4
##   recycling                  2                  2          0
```

Such tables can be used to confirm that impactFactors does not contain unnecessary duplicates. For example, since there are 2 materials and 2 impact categories under consideration, there should be exactly 4 production-related impact factors – as the table shows.

Merging the two tables to produce impactsInDetail.

massProfiles and impactsInDetail will be merged to allow us to calculate impacts, but before we do that we must address a limitation of massProfiles. So far massProfiles only includes tons of materials handled at the end-of-life phase of the life cycle. We must also account for the tons of those materials that are handled at two other life cycle stages: end-of-life transport and production.

We will add tonnages representing production here using a simple copy- and append operation. In the following code, all the cases from massProfiles are copied, labeled with a disposition (and umbDisp) of “production,” and then added back to massProfiles, creating a new data frame, massProfilesPlus.

```
# copy end-of-life tons and label them as production tons
tempProductionMasses <-
  massProfiles %>%
  mutate(
    disposition="production",
    umbDisp="production",
    miles=NA
  )
```

```r
# add the production tons to the end-of-life tons
massProfilesPlus <-
  bind_rows(
    massProfiles,
    tempProductionMasses
  ) %>%
  # sort the new, larger table
  arrange(
    scenario, wasteshed, material, disposition
  )
rm(tempProductionMasses) # remove temporary table
```

The resulting table, massProfilesPlus, should have exactly twice the total tonnage of massProfiles. Moreover, within each scenario, production tons should have the same sum as end-of-life tons.

```r
sum(massProfiles$tons)

## [1] 114227.6

sum(massProfilesPlus$tons)

## [1] 228455.2

massProfilesPlus %>%
  group_by(scenario, umbDisp) %>%
  summarise(tons=sum(tons))

## `summarise()` regrouping output by 'scenario' (override with `.groups` arg
ument)

## # A tibble: 9 x 3
## # Groups:   scenario [3]
##   scenario             umbDisp      tons
##   <chr>                <chr>       <dbl>
## 1 baseline             disposal   49130.
## 2 baseline             production 54516.
## 3 baseline             recovery    5387.
## 4 eliminate_food_waste disposal    2828.
## 5 eliminate_food_waste production  5195.
## 6 eliminate_food_waste recovery    2367.
## 7 recover_nearly_all   disposal       0
## 8 recover_nearly_all   production 54516.
## 9 recover_nearly_all   recovery   54516.
```

The tonnages associated with end-of-life transport are still missing, but they will be generated during the following merge of massProfiles and impactFactors.

The merge is made on unique combinations of material and disposition name. Since the resulting file has both tons (from the massProfiles table) and impactFactor (from the impactFactors table), these can be multiplied to get an impact in units of impactUnits.

Like so:

```
impactsInDetail <-
  # joining all impact factors relevant to massProfiles
  left_join(  # important: use left_join not full_join
    massProfilesPlus,
    impactFactors,
    by = c("material", "disposition")
  ) %>%
  # calculating impacts with special considerations
  # for end-of-life transport impacts
  mutate(
    # if miles is missing replace it with default value
    miles = ifelse(is.na(miles), impliedMiles, miles),
    # calculate impact
    impact =
      case_when(
        LCstage != "endOfLifeTransport" ~ tons*impactFactor,
        LCstage == "endOfLifeTransport" ~
          tons*(miles/impliedMiles)*impactFactor
      )
  ) %>%
  arrange(impactCategory, scenario, material, LCstage, disposition)
```

This creates a data frame, impactsInDetail, with records for each combination of scenario, wasteshed, material, LCstage, disposition, and impactCategory. A printout of this table is relatively lengthy:

```
kable(
  impactsInDetail
)
```

wasteshed	material	disposition	umbDisp	scenario	tons	miles	LCstage	impactCategory	impactUnits	impliedMiles	impactFactor	impact
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180	endOfLife	Energy demand	MJ	180	832.8763	2.355138e+06
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180	endOfLife	Energy demand	MJ	180	-13408.5010	-3.173881e+07
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	1.706078e+06
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	1.428149e+06
City A	Electronics	production	production	baseline	2.827717e+03	180	production	Energy demand	MJ	180	468130.3713	1.323740e+09
City A	Electronics	production	production	baseline	2.367066e+03	180	production	Energy demand	MJ	180	468130.3713	1.108095e+09
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-5592.4907	0.000000e+00
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180	endOfLife	Energy demand	MJ	180	-574.1139	-1.068965e+01
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180	endOfLife	Energy demand	MJ	180	-574.1139	-8.335508e+05
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180	endOfLife	Energy demand	MJ	180	-543.1128	-8.515606e+05
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180	endOfLife	Energy demand	MJ	180	319.6540	1.480056e+07
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	1.123625e+01
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	8.761733e+05
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	9.461970e+05
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	2.794177e+07
City A	FoodWaste	production	production	baseline	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	baseline	1.861940e-02	180	production	Energy demand	MJ	180	39043.8618	7.269730e+02
City A	FoodWaste	production	production	baseline	1.451891e+03	180	production	Energy demand	MJ	180	39043.8618	5.668743e+07
City A	FoodWaste	production	production	baseline	1.567926e+03	180	production	Energy demand	MJ	180	39043.8618	6.121789e+07
City A	FoodWaste	production	production	baseline	4.630180e+04	180	production	Energy demand	MJ	180	39043.8618	1.807801e+09
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180	endOfLife	Energy demand	MJ	180	832.8763	2.355138e+06
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180	endOfLife	Energy demand	MJ	180	-13408.5010	-3.173881e+07
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	1.706078e+06
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	1.428149e+06
City A	Electronics	production	production	eliminate_food_waste	2.827717e+03	180	production	Energy demand	MJ	180	468130.3713	1.323740e+09
City A	Electronics	production	production	eliminate_food_waste	2.367066e+03	180	production	Energy demand	MJ	180	468130.3713	1.108095e+09
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-5592.4907	0.000000e+00
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-574.1139	0.000000e+00
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-574.1139	0.000000e+00
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-543.1128	0.000000e+00
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180	endOfLife	Energy demand	MJ	180	319.6540	0.000000e+00
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	Electronics	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Energy demand	MJ	180	832.8763	0.000000e+00
City A	Electronics	recycling	recovery	recover_nearly_all	5.194783e+03	180	endOfLife	Energy demand	MJ	180	-13408.5010	-6.965425e+07
City A	Electronics	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	0.000000e+00
City A	Electronics	recycling	recovery	recover_nearly_all	5.194783e+03	180	endOfLifeTransport	Energy demand	MJ	180	603.3414	3.134227e+06
City A	Electronics	production	production	recover_nearly_all	0.000000e+00	180	production	Energy demand	MJ	180	468130.3713	0.000000e+00
City A	Electronics	production	production	recover_nearly_all	5.194783e+03	180	production	Energy demand	MJ	180	468130.3713	2.431836e+09
City A	FoodWaste	anaerobicDigestion	recovery	recover_nearly_all	4.932163e+04	180	endOfLife	Energy demand	MJ	180	-5592.4907	-2.758308e+08
City A	FoodWaste	combustion	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-574.1139	0.000000e+00
City A	FoodWaste	combustion	recovery	recover_nearly_all	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-574.1139	0.000000e+00
City A	FoodWaste	composting	recovery	recover_nearly_all	0.000000e+00	180	endOfLife	Energy demand	MJ	180	-543.1128	0.000000e+00
City A	FoodWaste	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Energy demand	MJ	180	319.6540	0.000000e+00
City A	FoodWaste	anaerobicDigestion	recovery	recover_nearly_all	4.932163e+04	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	2.976415e+07
City A	FoodWaste	combustion	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00

wasteshed	material	disposition	umbDisp	scenario	tons	miles	LCstage	impactCategory	impactUnits	impliedMiles	impactFactor	impact
City A	FoodWaste	combustion	recovery	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	composting	recovery	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Energy demand	MJ	180	603.4704	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	4.932163e+04	180	production	Energy demand	MJ	180	39043.8618	1.925707e+09
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Energy demand	MJ	180	39043.8618	0.000000e+00
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180	endOfLife	Water consumption	kg	180	216.3978	6.119116e+05
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180	endOfLife	Water consumption	kg	180	-8811.4635	-2.085732e+07
City A	Electronics	landfilling	disposal	baseline	2.827717e+03	180	endOfLifeTransport	Water consumption	kg	180	108.2953	3.062285e+05
City A	Electronics	recycling	recovery	baseline	2.367066e+03	180	endOfLifeTransport	Water consumption	kg	180	108.2953	2.563422e+05
City A	Electronics	production	production	baseline	2.827717e+03	180	production	Water consumption	kg	180	271442.1168	7.675614e+08
City A	Electronics	production	production	baseline	2.367066e+03	180	production	Water consumption	kg	180	271442.1168	6.425214e+08
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180	endOfLife	Water consumption	kg	180	1230.1352	0.000000e+00
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180	endOfLife	Water consumption	kg	180	-175.0662	-3.259627e+00
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180	endOfLife	Water consumption	kg	180	-175.0662	-2.541771e+05
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180	endOfLife	Water consumption	kg	180	158.5119	2.485349e+05
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180	endOfLife	Water consumption	kg	180	-250.7538	-1.161035e+07
City A	FoodWaste	anaerobicDigestion	recovery	baseline	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	combustion	disposal	baseline	1.861940e-02	180	endOfLifeTransport	Water consumption	kg	180	108.3185	2.016824e+00
City A	FoodWaste	combustion	recovery	baseline	1.451891e+03	180	endOfLifeTransport	Water consumption	kg	180	108.3185	1.572666e+05
City A	FoodWaste	composting	recovery	baseline	1.567926e+03	180	endOfLifeTransport	Water consumption	kg	180	108.3185	1.698354e+05
City A	FoodWaste	landfilling	disposal	baseline	4.630180e+04	180	endOfLifeTransport	Water consumption	kg	180	108.3185	5.015341e+06
City A	FoodWaste	production	production	baseline	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	baseline	1.861940e-02	180	production	Water consumption	kg	180	132157.0933	2.460685e+03
City A	FoodWaste	production	production	baseline	1.451891e+03	180	production	Water consumption	kg	180	132157.0933	1.918777e+08
City A	FoodWaste	production	production	baseline	1.567926e+03	180	production	Water consumption	kg	180	132157.0933	2.072125e+08
City A	FoodWaste	production	production	baseline	4.630180e+04	180	production	Water consumption	kg	180	132157.0933	6.119111e+09
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180	endOfLife	Water consumption	kg	180	216.3978	6.119116e+05
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180	endOfLife	Water consumption	kg	180	-8811.4635	-2.085732e+07
City A	Electronics	landfilling	disposal	eliminate_food_waste	2.827717e+03	180	endOfLifeTransport	Water consumption	kg	180	108.2953	3.062285e+05
City A	Electronics	recycling	recovery	eliminate_food_waste	2.367066e+03	180	endOfLifeTransport	Water consumption	kg	180	108.2953	2.563422e+05
City A	Electronics	production	production	eliminate_food_waste	2.827717e+03	180	production	Water consumption	kg	180	271442.1168	7.675614e+08
City A	Electronics	production	production	eliminate_food_waste	2.367066e+03	180	production	Water consumption	kg	180	271442.1168	6.425214e+08
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Water consumption	kg	180	1230.1352	0.000000e+00
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180	endOfLife	Water consumption	kg	180	-175.0662	0.000000e+00
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Water consumption	kg	180	-175.0662	0.000000e+00
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180	endOfLife	Water consumption	kg	180	158.5119	0.000000e+00
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180	endOfLife	Water consumption	kg	180	-250.7538	0.000000e+00
City A	FoodWaste	anaerobicDigestion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	combustion	disposal	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	combustion	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	composting	recovery	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	landfilling	disposal	eliminate_food_waste	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	eliminate_food_waste	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	Electronics	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Water consumption	kg	180	216.3978	0.000000e+00
City A	Electronics	recycling	recovery	recover_nearly_all	5.194783e+03	180	endOfLife	Water consumption	kg	180	-8811.4635	-4.577364e+07
City A	Electronics	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.2953	0.000000e+00
City A	Electronics	recycling	recovery	recover_nearly_all	5.194783e+03	180	endOfLifeTransport	Water consumption	kg	180	108.2953	5.625707e+05
City A	Electronics	production	production	recover_nearly_all	0.000000e+00	180	production	Water consumption	kg	180	271442.1168	0.000000e+00

wasteshed	material	disposition	umbDisp	scenario	tons	miles	LCstage	impactCategory	impactUnits	impliedMiles	impactFactor	impact
City A	Electronics	production	production	recover_nearly_all	5.194783e+03	180	production	Water consumption	kg	180	271442.1168	1.410083e+09
City A	FoodWaste	anaerobicDigestion	recovery	recover_nearly_all	4.932163e+04	180	endOfLife	Water consumption	kg	180	1230.1352	6.067228e+07
City A	FoodWaste	combustion	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Water consumption	kg	180	-175.0662	0.000000e+00
City A	FoodWaste	combustion	recovery	recover_nearly_all	0.000000e+00	180	endOfLife	Water consumption	kg	180	-175.0662	0.000000e+00
City A	FoodWaste	composting	recovery	recover_nearly_all	0.000000e+00	180	endOfLife	Water consumption	kg	180	158.5119	0.000000e+00
City A	FoodWaste	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLife	Water consumption	kg	180	-250.7538	0.000000e+00
City A	FoodWaste	anaerobicDigestion	recovery	recover_nearly_all	4.932163e+04	180	endOfLifeTransport	Water consumption	kg	180	108.3185	5.342445e+06
City A	FoodWaste	combustion	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	combustion	recovery	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	composting	recovery	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	landfilling	disposal	recover_nearly_all	0.000000e+00	180	endOfLifeTransport	Water consumption	kg	180	108.3185	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	4.932163e+04	180	production	Water consumption	kg	180	132157.0933	6.518204e+09
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00
City A	FoodWaste	production	production	recover_nearly_all	0.000000e+00	180	production	Water consumption	kg	180	132157.0933	0.000000e+00

Some things to note about the impactsInDetails table:

- Each line is labeled with the *umbDisp* from massProfiles, so distinctions can be made between recovery and disposal impacts or tonnages if desired.
- There are 3 LCstages (production, end-of-life transport, and end-of-life treatment) represented in impactsInDetail. The properties of the left_join merge have created the records needed to represent the (hitherto missing) end-of-life transport stage. Since in every impact category, each material & end-of-life disposition has *two* records in impactFactors (one for the LCstage endOfLife and another for endOfLifeTransport), a left_join has added tonnages related to end-of-life transport.

Checking the internal consistency of *impactsInDetail*

Before using the impactsInDetail file to calculate results, some basic quality checks should be performed.

For example, tonnages associated with all life cycles should have the same value within each scenario. That is, within each scenario, tons for "production" should be the same as tons for "endOfLifeTransport" and "endOfLife". Code like the following can confirm that:

```
impactsInDetail %>%
  group_by(LCstage, scenario) %>%
  summarise(tons=sum(tons)) %>%
  arrange(scenario, LCstage) %>%
  kable()

## `summarise()` regrouping output by 'LCstage' (override with `.groups` argu
ment)
```

LCstage	scenario	tons
endOfLife	baseline	109032.83
endOfLifeTransport	baseline	109032.83
production	baseline	109032.83
endOfLife	eliminate_food_waste	10389.57
endOfLifeTransport	eliminate_food_waste	10389.57
production	eliminate_food_waste	10389.57
endOfLife	recover_nearly_all	109032.83
endOfLifeTransport	recover_nearly_all	109032.83
production	recover_nearly_all	109032.83

Note that tonnages in the table above are not identical to tonnages listed in the massProfiles table. Besides the recent addition of production-related tons, and end-of-life transport tons, impactsInDetails has a complete set of tons for every impactCategory in use.

It is also valuable to check that every record has an impact factor. No impact factors should be missing, and any that are zero should be viewed with suspicion (because impactFactors of exactly zero are unlikely, and may represent a computation error or lazy assumption). In addition, impact and tons may be zero but should not be missing. These things can be checked with code like this:

```
impactsInDetail %>%
  filter(is.na(impactFactor) | impactFactor==0) %>%
  nrow()

## [1] 0

impactsInDetail %>%
  filter(is.na(impact)) %>%
  nrow()

## [1] 0

impactsInDetail %>%
  filter(is.na(tons)) %>%
  nrow()

## [1] 0
```

In each of these cases, the nrow() call has output 0. This means that our impactsInDetail table has passed these particular quality checks. If nrow() output >1, then it would be necessary to backtrack and correct something.

When impactsInDetail fails such simple internal-consistency checks, it is likely to be the result of mismatches between the massProfiles and impactFactors tables. Spellings of material and disposition names must match exactly, and every field in every table (with the exception of the *miles* field) must be filled in with a reasonable value.

Creating tabular and graphical output

Guidelines

The impactsInDetail data frame is the source of all future output from this analysis. Most results of interest – for example, the total waste tonnages and total impacts linked to each scenario – are simple summations of tons or impacts from impactsInDetail.

When creating results from impactsInDetail, recall that:

- there is much redundancy in this data table now – an abundance of records representing every combination of scenario, wasteshed, material, LCstage, disposition, and impactCategory. So data must be filtered down to the desired specific content to avoid miscalculation.
- when tons are summed, they should be restricted to tons marked with the "endOfLife" LCstage. The tons that appear in other LCstages are redundant and only serve for the calculation of the impacts of those stages.
- furthermore when tons are summed, they should be restricted to a single impact category (it should not matter which) – as the complete set of tonnages has been repeated for every impact category.
- impacts should be summed within the context of a single impactCategory – unless users are willing to create, program, and defend a method for normalizing and/or summarizing across multiple impact categories.

Some utility objects

For the purpose of creating charts and tables, a few miscellaneous objects could be useful:

- a plaintext list of material names, sorted in descending order of abundance
- a table of likely impact category labels
- a graphical theme for charts
- an ordered list of scenario names

```
# most abundant materials in the wastestream, in order
materialSortOrder <-
  massProfiles %>%
  group_by(material) %>%
  summarise(tons=sum(tons)) %>%
  arrange(desc(tons)) %>%
  pull(material)

## `summarise()` ungrouping output (override with `.groups` argument)

# a table of impact categories combined with units
# (for use in chart labels)
impactLabels <-
  impactFactors %>%
  select(impactCategory, impactUnits) %>%
  distinct() %>%
  mutate(
    impactLabel =
      paste(
        impactCategory,
        " (",
        impactUnits,
        ")",
        sep=""
      )
  )
# a custom graphic theme for charts, inspired by
# the fivethirtyeight theme
theme_539 <- function() {
  theme_fivethirtyeight() +
  theme(
    rect=element_rect(fill="transparent"),
    panel.grid = element_blank(),
    axis.ticks = element_line()
  )
}

# making an ordered list of scenarios, where "baseline" is first
scenarioOrder <-
  c(
    "baseline",
    setdiff(
```

```
        massProfiles %>%
        select(scenario) %>%
        distinct() %>%
        pull(scenario),
      "baseline")
  )
```

Weights of waste in each of the scenarios

Note that when weights are summed, only the “endOfLife” LCstage is used, and only a single impactCategory is used.

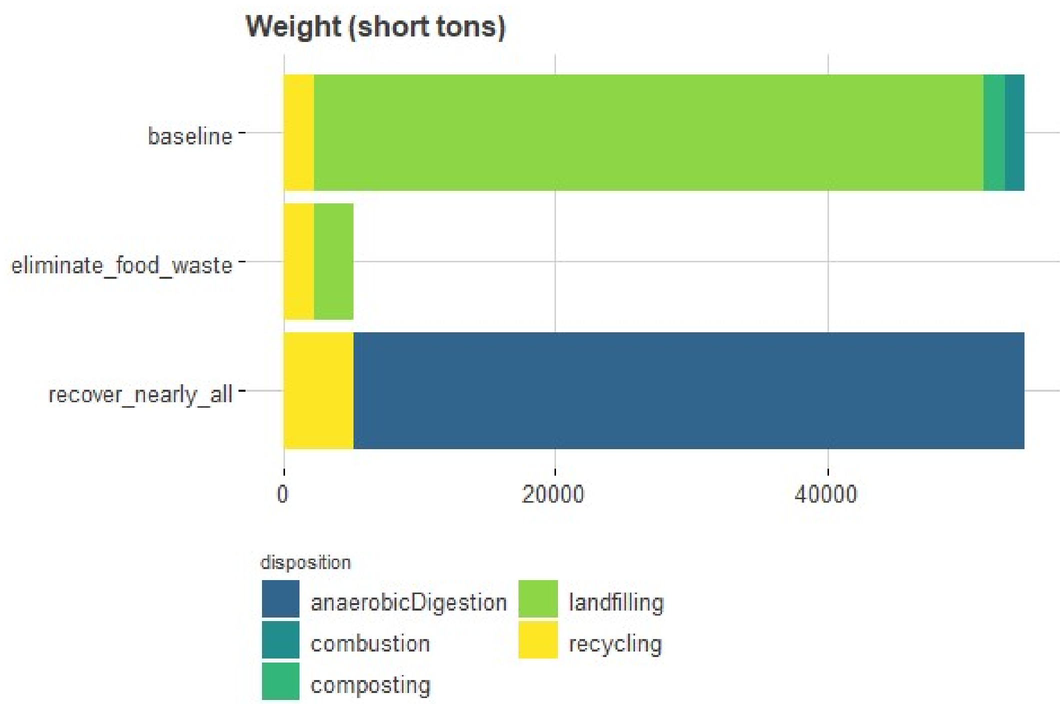
```
# summing weights by disposition for each scenario
tempWeightData1 <-
  impactsInDetail %>%
  filter(
    LCstage == "endOfLife" & impactCategory=="Energy demand"
    ) %>% # correct set for weight calculations
  group_by(scenario, disposition) %>%
  summarise(tons=sum(tons)) %>%
  ungroup() %>%
  filter(tons != 0) %>%
  mutate(scenario= factor(scenario, levels=rev(scenarioOrder)))

## `summarise()` regrouping output by 'scenario' (override with `.groups` arg
ument)

kable(tempWeightData1)
```

scenario	disposition	tons
baseline	combustion	1451.910
baseline	composting	1567.926
baseline	landfilling	49129.515
baseline	recycling	2367.066
eliminate_food_waste	landfilling	2827.717
eliminate_food_waste	recycling	2367.066
recover_nearly_all	anaerobicDigestion	49321.634
recover_nearly_all	recycling	5194.783

making that weight data into a chart…

```
tempWeightChart1 <-
  ggplot()+
  ggtitle("Weight (short tons)")+
  theme_539()+
  geom_bar(
    data = tempWeightData1,
    aes(x = scenario, y= tons, fill= disposition),
    color=NA,
    stat="identity"
  )+
  scale_fill_viridis(begin=0.32, end=1, discrete = TRUE)+
  coord_flip()+
```

```
  guides(fill=guide_legend(ncol=2, title.position = "top"))+
  theme(
    rect=element_rect(fill="transparent"),
    plot.title = element_text(size=12),
    legend.position="bottom",
    legend.title = element_text(size=8),
    legend.justification="left"
    )
# printing the chart to the current device
tempWeightChart1
```




```
# saving the chart as external files
ggsave("chart_output/weights.png")

## Saving 6.5 x 4.5 in image

ggsave("chart_output/weights.svg")

## Saving 6.5 x 4.5 in image
```

Life cycle impacts for waste in each scenario

Now for comparison, let's look at the impacts associated with those scenarios. But here we will have to deal with more voluminous output, because there are multiple impact categories.

First, sum up the impacts in similar detail to the weight chart:

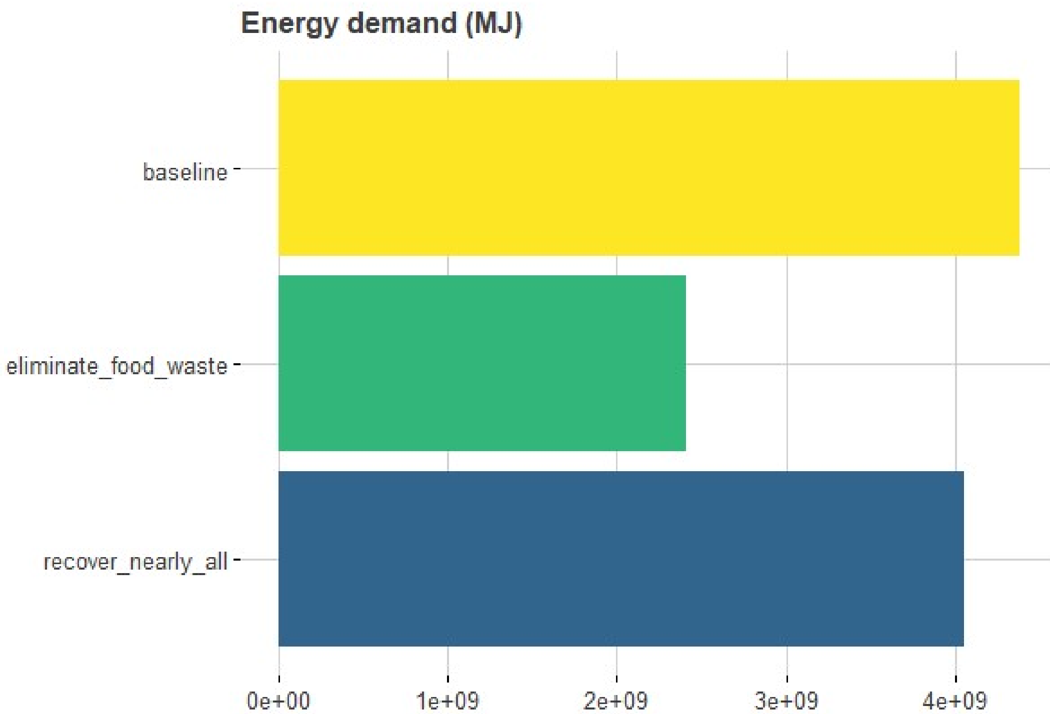
```r
tempImpactData1 <-
  impactsInDetail %>%
  group_by(scenario, impactCategory, impactUnits) %>%
  summarise(impact=sum(impact)) %>%
  ungroup() %>%
  mutate(
    scenario = factor(scenario, levels = rev(scenarioOrder)),
    impactLabel =
      paste(
        impactCategory,
        " (",
        impactUnits,
        ")",
        sep=""
      )
  )

## `summarise()` regrouping output by 'scenario', 'impactCategory' (override
with `.groups` argument)

kable(tempImpactData1)
```

scenario	impactCategory	impactUnits	impact	impactLabel
baseline	Energy demand	MJ	4374172763	Energy demand (MJ)
baseline	Water consumption	kg	7902330212	Water consumption (kg)
eliminate_food_waste	Energy demand	MJ	2405586118	Energy demand (MJ)
eliminate_food_waste	Water consumption	kg	1390399982	Water consumption (kg)
recover_nearly_all	Energy demand	MJ	4044955971	Energy demand (MJ)
recover_nearly_all	Water consumption	kg	7949090257	Water consumption (kg)

For a single impactCategory, we can make an impact chart analagous to the weight chart:

```r
# chose a single impactCategory
tempImpactCategory <- sample_n(impactLabels, 1)
# get the impacts for that category
tempImpactChart1 <-
  ggplot()+
  ggtitle(tempImpactCategory$impactLabel)+
  theme_539()+
  geom_bar(
    data =
      tempImpactData1 %>%
        filter(impactCategory==tempImpactCategory$impactCategory),
    aes(x = scenario, y= impact, fill=scenario),
    color=NA,
    stat="identity"
  )+
  scale_fill_viridis(begin=0.32, end=1, discrete = TRUE)+
  coord_flip()+
  guides(fill=guide_legend(ncol=2, title.position = "top"))+
  theme(
    rect=element_rect(fill="transparent"),
```

```
    plot.title = element_text(size=12),
    legend.position="none",
    legend.title = element_text(size=8),
    legend.justification="left"
    )
tempImpactChart1
```

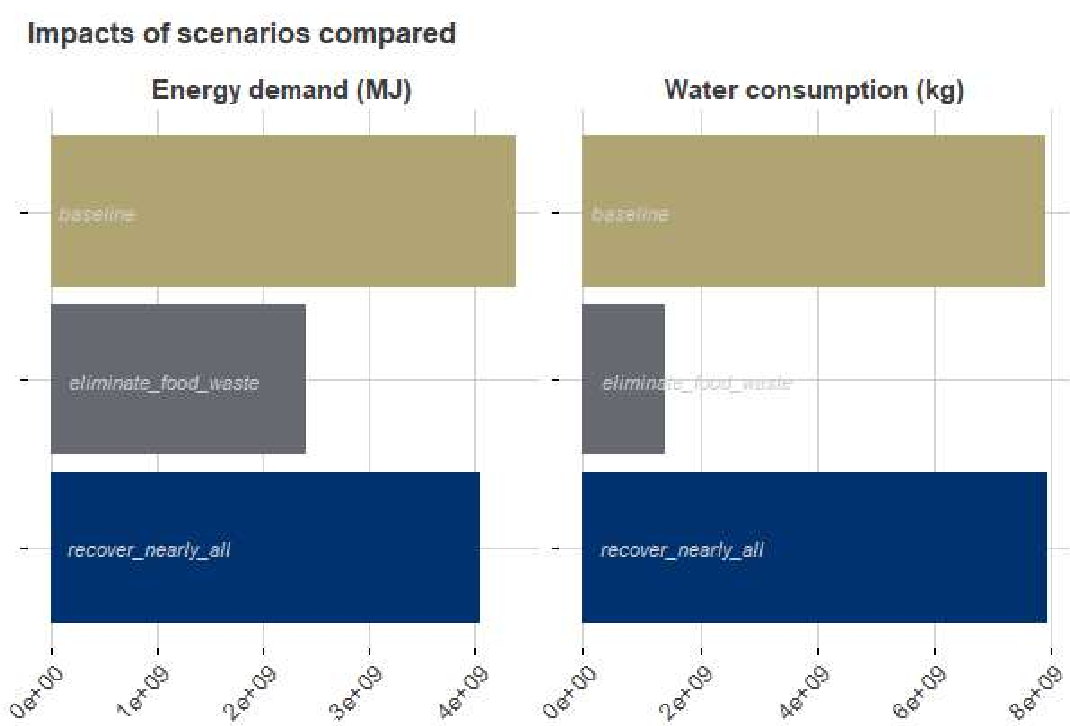



```
ggsave("chart_output/impacts1.png")

## Saving 6.5 x 4.5 in image

ggsave("chart_output/impacts1.svg")

## Saving 6.5 x 4.5 in image
```

If we use a more complex layout, we can create a single image with facets expressing all of the available impact categories:

```
tempImpactChart1 <-
  ggplot()+
  ggtitle("Impacts of scenarios compared")+
  theme_539()+
  geom_bar(
    data = tempImpactData1,
    aes(x = scenario, y= impact, fill=scenario),
    color=NA,
    # size=2,
    stat="identity"
```

```
  )+
  geom_text(
    data=tempImpactData1,
    aes(x=scenario, y=0, label=scenario),
    color="gray80",
    size=3,
    fontface="italic",
    hjust=-0.1
  )+
  facet_wrap(~impactLabel, ncol=2, scales="free")+
  scale_fill_viridis(
    begin=0.1, end=0.7, option="cividis", discrete = TRUE
  )+
  coord_flip()+
#  guides(fill=guide_legend(ncol=2, title.position = "top"))+
  theme(
    rect=element_rect(fill="transparent"),
    plot.title = element_text(size=12),
    legend.position="none",
    axis.text.x=element_text(angle=45, hjust=1),
    axis.text.y=element_blank(),
    strip.text = element_text(size=11, face="bold")
    )
tempImpactChart1
```




```
ggsave("chart_output/impacts2.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
ggsave("chart_output/impacts2.svg")
```

```
## Saving 6.5 x 4.5 in image
```

Such side-by-side results show that management scenarios do not affect all impact categories equally. In the chart above:

- maximizing recovery ("recover_nearly_all") decreases energy demand somewhat compared to baseline, but does not decrease water consumption.
- eliminating food waste dramatically reduces water consumption, but the decrease for energy demand is somewhat smaller.

Detailing weight by umbrella disposition

The charts above don't have a lot of detail about waste management. Let's redo the weight chart so that it characterizes the weight-based recovery rate in each scenario. For that, we sum weights by scenario and umbDisp:

```
# summing weights by umbDisp for each scenario
tempWeightData2 <-
  impactsInDetail %>%
  filter(
    LCstage == "endOfLife" & impactCategory == "Energy demand"
  ) %>% # correct set for weight calculations
  group_by(scenario, umbDisp) %>%
  summarise(tons=sum(tons)) %>%
  ungroup() %>%
  filter(tons != 0) %>%
  mutate(scenario= factor(scenario, levels=rev(scenarioOrder))) %>%
  group_by(scenario) %>%
  mutate(recovTons=ifelse(umbDisp=="recovery",tons,0)) %>%
  ungroup()
```

```
## `summarise()` regrouping output by 'scenario' (override with `.groups` arg
ument)
```

```
kable(tempWeightData2)
```

scenario	umbDisp	tons	recovTons
baseline	disposal	49129.534	0.000
baseline	recovery	5386.883	5386.883
eliminate_food_waste	disposal	2827.717	0.000
eliminate_food_waste	recovery	2367.066	2367.066
recover_nearly_all	recovery	54516.417	54516.417

```
# creating a data file that expresses weight-based
# recovery rate for each scenario
tempWeightData2a <-
  tempWeightData2 %>%
  group_by(scenario) %>%
  summarise(
```

```
    recovTons=sum(recovTons),
    tons=sum(tons)
  ) %>%
  ungroup() %>%
  mutate(recovRate=recovTons/tons)
```

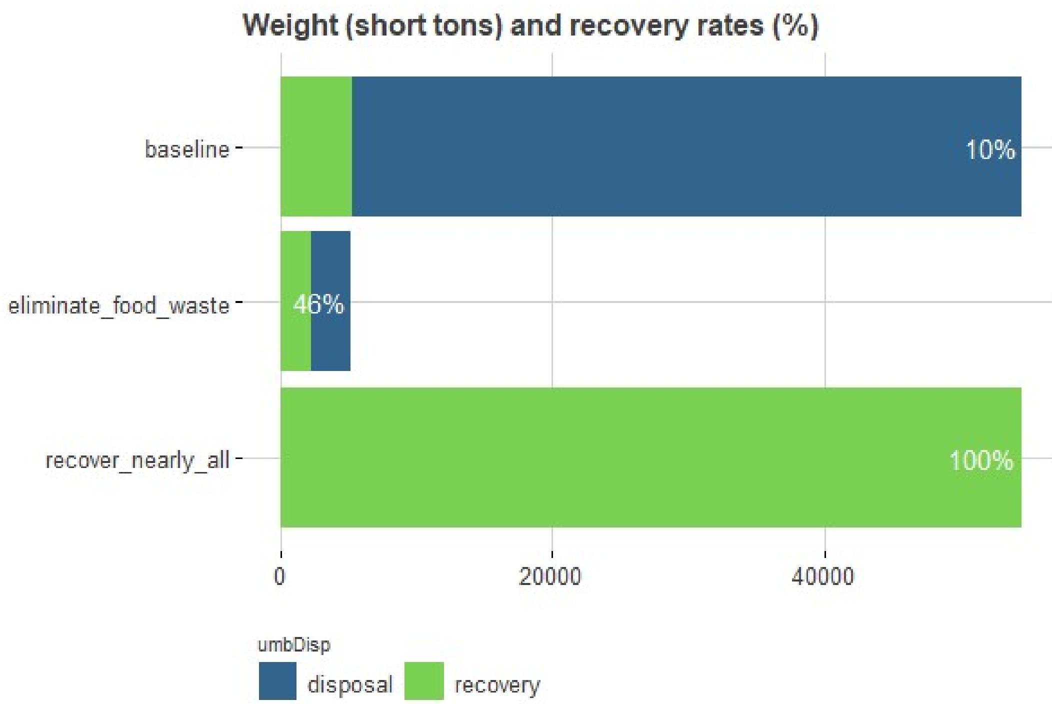
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
kable(tempWeightData2a)
```

scenario	recovTons	tons	recovRate
recover_nearly_all	54516.417	54516.417	1.0000000
eliminate_food_waste	2367.066	5194.783	0.4556622
baseline	5386.883	54516.417	0.0988121

making that weight data into a chart…

```
tempWeightChart2 <-
  ggplot()+
  ggtitle("Weight (short tons) and recovery rates (%)")+
  theme_539()+
  geom_bar(
    data = tempWeightData2,
    aes(x = scenario, y= tons, fill= umbDisp),
    color=NA,
    stat="identity"
  )+
  geom_text(
    data=tempWeightData2a,
    aes(x=scenario, y=tons, label=percent(recovRate)),
    hjust=1.1,
    color="white"
  )+
  scale_fill_viridis(
    begin=0.32, end=0.8, option="viridis", discrete = TRUE
    )+
  coord_flip()+
  guides(fill=guide_legend(ncol=2, title.position = "top"))+
  theme(
    rect=element_rect(fill="transparent"),
    plot.title = element_text(size=12),
    legend.position="bottom",
    legend.title = element_text(size=8),
    legend.justification="left"
    )
tempWeightChart2
```




```
ggsave("chart_output/weights2.png")

## Saving 6.5 x 4.5 in image

ggsave("chart_output/weights2.svg")

## Saving 6.5 x 4.5 in image
```

Illustrating impacts by life cycle stage

Previously the impacts associated with each scenario were given only as net values – the sum of three life cycle stages. It can be interesting to show how those three stages contribute to the net. To do that, we sum impacts by scenario and LCstage:

```
tempImpactData2 <-
  impactsInDetail %>%
  group_by(scenario, LCstage, impactCategory, impactUnits) %>%
  summarise(impact=sum(impact)) %>%
  ungroup() %>%
  mutate(
    scenario = factor(scenario, levels = rev(scenarioOrder)),
    impactLabel =
      paste(
        impactCategory,
        " (",
        impactUnits,
        ")",
        sep=""
```

```
      ),
    LCstage=
      factor(
        LCstage,
        levels=c("production","endOfLife","endOfLifeTransport")
      )
  )
```

```
## `summarise()` regrouping output by 'scenario', 'LCstage', 'impactCategory'
(override with `.groups` argument)
```

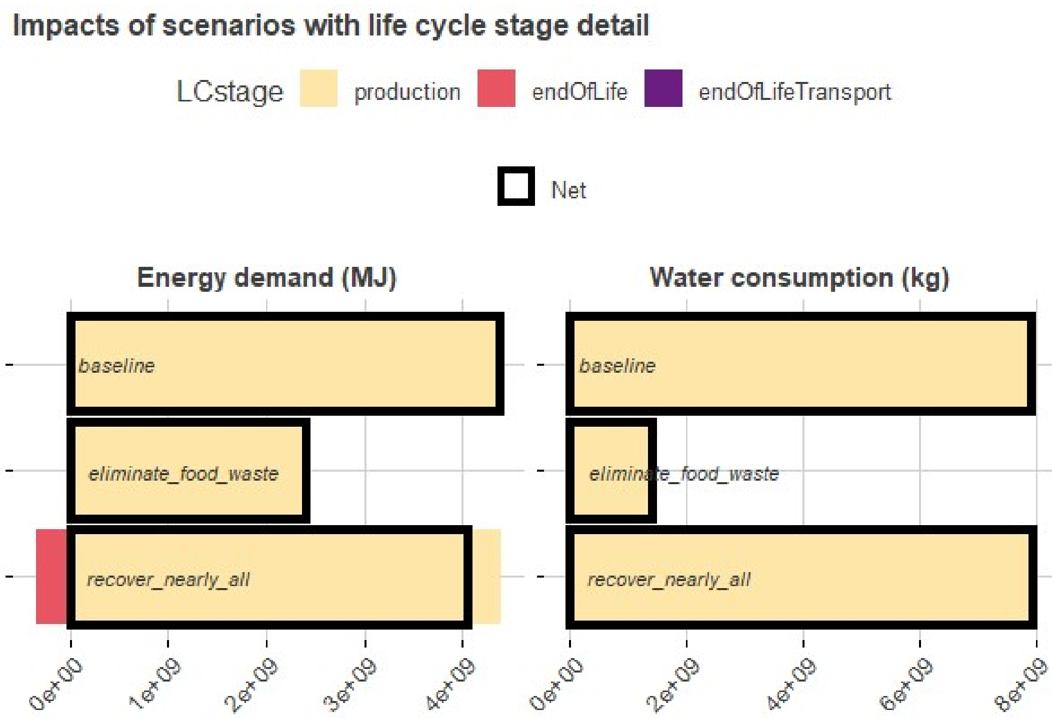
```
kable(tempImpactData2)
```

scenario	LCstage	impactCategory	impactUnits	impact	impactLabel
baseline	endOfLife	Energy demand	MJ	-16268235.5	Energy demand (MJ)
baseline	endOfLife	Water consumption	kg	-31861403.1	Water consumption (kg)
baseline	endOfLifeTransport	Energy demand	MJ	32898375.9	Energy demand (MJ)
baseline	endOfLifeTransport	Water consumption	kg	5905015.5	Water consumption (kg)
baseline	production	Energy demand	MJ	4357542622.3	Energy demand (MJ)
baseline	production	Water consumption	kg	7928286599.9	Water consumption (kg)
eliminate_food_waste	endOfLife	Energy demand	MJ	-29383668.5	Energy demand (MJ)
eliminate_food_waste	endOfLife	Water consumption	kg	-20245404.0	Water consumption (kg)
eliminate_food_waste	endOfLifeTransport	Energy demand	MJ	3134227.3	Energy demand (MJ)
eliminate_food_waste	endOfLifeTransport	Water consumption	kg	562570.7	Water consumption (kg)
eliminate_food_waste	production	Energy demand	MJ	2431835559.1	Energy demand (MJ)
eliminate_food_waste	production	Water consumption	kg	1410082815.2	Water consumption (kg)
recover_nearly_all	endOfLife	Energy demand	MJ	-345485027.1	Energy demand (MJ)
recover_nearly_all	endOfLife	Water consumption	kg	14898641.1	Water consumption (kg)
recover_nearly_all	endOfLifeTransport	Energy demand	MJ	32898375.9	Energy demand (MJ)
recover_nearly_all	endOfLifeTransport	Water consumption	kg	5905015.5	Water consumption (kg)
recover_nearly_all	production	Energy demand	MJ	4357542622.4	Energy demand (MJ)
recover_nearly_all	production	Water consumption	kg	7928286600.0	Water consumption (kg)

Now make that into a chart, with life cycle stage impacts in colors, and the (previously calculated) net impact as a black outline.

```
tempImpactChart2 <-
  ggplot()+
  ggtitle("Impacts of scenarios with life cycle stage detail")+
  theme_539()+
  geom_bar(
    data = tempImpactData2,
    aes(x = scenario, y= impact, fill=LCstage),
    color=NA,
    # size=2,
    stat="identity"
  )+
  geom_bar(
    data=tempImpactData1 %>% mutate(LCstage="Net"),
    aes(x=scenario, y=impact, color=LCstage),
    stat="identity",
    fill=NA,
    size=2
  )+
  geom_text(
```

```
    data=tempImpactData1 %>% mutate(LCstage="Net"),
    aes(x=scenario, y=0, label=scenario),
    stat="identity",
    color="gray20",
    size=3,
    fontface="italic",
    hjust=-0.1,
    vjust=0.5
  )+
  facet_wrap(~impactLabel, ncol=2, scales="free")+
  scale_color_manual(values="black")+
  scale_fill_viridis(
    begin=0.32, end=0.95, option="magma", discrete = TRUE,
    direction = -1
  )+
  coord_flip()+
  guides(
    color=guide_legend(nrow=1, title=NULL),
    fill=guide_legend(nrow=1, title.position = "left")
    )+
  theme(
    rect=element_rect(fill="transparent"),
    plot.title = element_text(size=12),
    legend.position="top",
    axis.text.x=element_text(angle=45, hjust=1),
    axis.text.y=element_blank(),
    strip.text = element_text(size=11, face="bold")
    )
tempImpactChart2
```




```
ggsave("chart_output/impacts3.png")
## Saving 6.5 x 4.5 in image
ggsave("chart_output/impacts3.svg")
## Saving 6.5 x 4.5 in image
```

The chart above shows that, at least in this analysis, production impacts (the tan color) make by far the biggest contribution to the net impact (black outline). End-of-life credits are visible, in the "recover_nearly_all" scenario, but they are smaller than many might guess. End-of-life transport impacts are so small they are not visible, probably obscured by the black "net" line.

Heatmap

Another way to compare impact results across scenarios is the "heatmap", where all impacts are scaled to the baseline value.

Here is the code to produce a data table which can be drawn as a heatmap:

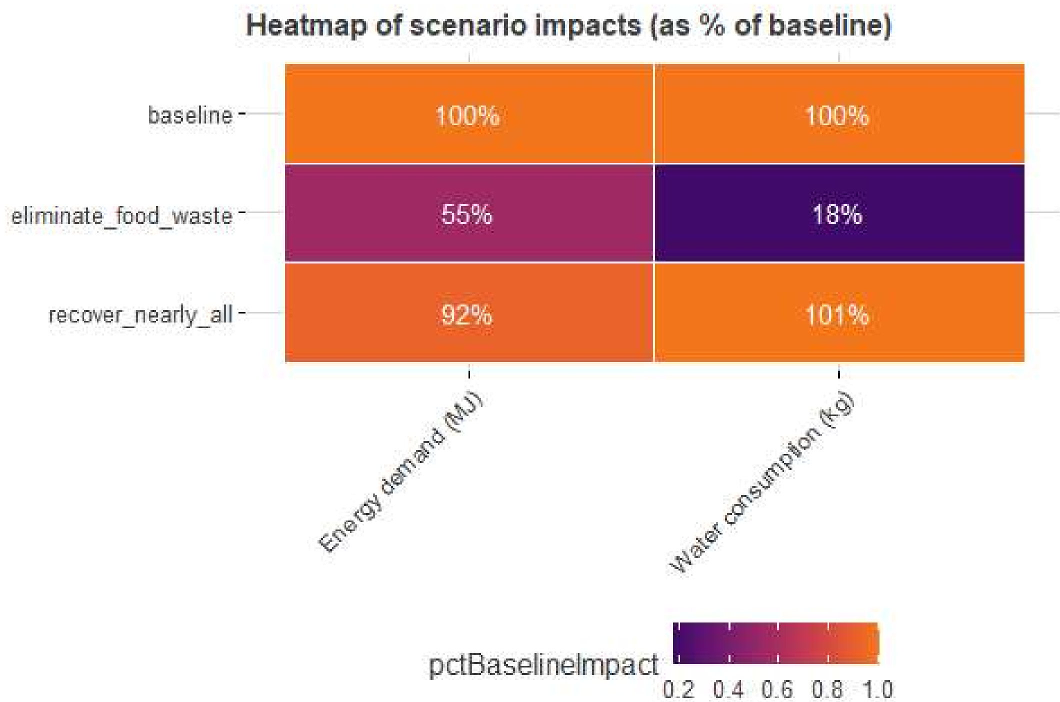
```
tempImpactData3 <-
  tempImpactData1 %>%
  filter(scenario=="baseline") %>%
  select(impactLabel, impact) %>%
  rename(baselineImpact=impact)
tempImpactData3a <-
  left_join(
```

```
    tempImpactData1,
    tempImpactData3,
    by= c("impactLabel")
  ) %>%
  mutate(
    pctBaselineImpact=impact/baselineImpact
  )
kable(tempImpactData3a)
```

scenario	impactCategory	impactUnits	impact	impactLabel	baselineImpact	pctBaselineImpact
baseline	Energy demand	MJ	4374172763	Energy demand (MJ)	4374172763	1.0000000
baseline	Water consumption	kg	7902330212	Water consumption (kg)	7902330212	1.0000000
eliminate_food_waste	Energy demand	MJ	2405586118	Energy demand (MJ)	4374172763	0.5499522
eliminate_food_waste	Water consumption	kg	1390399982	Water consumption (kg)	7902330212	0.1759481
recover_nearly_all	Energy demand	MJ	4044955971	Energy demand (MJ)	4374172763	0.9247362
recover_nearly_all	Water consumption	kg	7949090257	Water consumption (kg)	7902330212	1.0059172

now, to make that into a chart:

```
tempImpactChart3 <-
  ggplot()+
  ggtitle("Heatmap of scenario impacts (as % of baseline)")+
  theme_539()+
  geom_tile(
    data=tempImpactData3a,
    aes(y=scenario, x=impactLabel, fill=pctBaselineImpact),
    color="white"
  )+
  geom_text(
    data=tempImpactData3a,
    aes(
      y=scenario, x=impactLabel, label=percent(pctBaselineImpact,1)
    ),
    color="white"
  )+
  scale_fill_viridis(begin=0.2, end=0.7, option="inferno")+
  theme(
    plot.title = element_text(size=12),
    rect=element_rect(fill="transparent"),
    panel.grid = element_blank(),
    axis.ticks = element_line(),
    axis.text.x = element_text(hjust=1, angle=45)
  )
tempImpactChart3
```




```r
ggsave("chart_output/impacts4.png")

## Saving 6.5 x 4.5 in image

ggsave("chart_output/impacts4.svg")

## Saving 6.5 x 4.5 in image
```

Weight vs impacts within a single scenario

Most of the results displayed above are high-level summaries. But it's also helpful to look at a local waste stream in detail, to see how weight and impacts compare across individual materials. Here I set up a data table that can draw a comparison between weights and impacts of individual materials within a single scenario. Both waste and impacts are characterized as a percentage of the relevant waste-shed wide total.

Setting up such comparisons involves some more extended data processing.

```r
tempWeightData4 <-
  impactsInDetail %>%
  filter(
    LCstage == "endOfLife" & impactCategory=="Energy demand"
  ) %>% # correct set for weight calculations
  group_by(wasteshed, scenario, material, umbDisp) %>%
  summarise(tons=sum(tons)) %>%
  ungroup() %>%
  filter(tons != 0) %>%
  mutate(scenario= factor(scenario, levels=rev(scenarioOrder))) %>%
```

```
  mutate(recovTons=ifelse(umbDisp=="recovery",tons,0)) %>%
  mutate(material=factor(material, levels=materialSortOrder))

## `summarise()` regrouping output by 'wasteshed', 'scenario', 'material' (ov
erride with `.groups` argument)

# creating a data file that expresses weight-based
# recovery rate for each scenario
tempWeightData4a <-
  tempWeightData4 %>%
  group_by(wasteshed, scenario, material) %>%
  summarise(
    recovTons=sum(recovTons),
    tons=sum(tons)
  ) %>%
  ungroup() %>%
  mutate(recovRate=recovTons/tons)

## `summarise()` regrouping output by 'wasteshed', 'scenario' (override with
`.groups` argument)

# total weight for the scenario
tempWeightData4b <-
  tempWeightData4a %>%
  group_by(wasteshed, scenario) %>%
  summarise(allTons=sum(tons)) %>%
  ungroup()

## `summarise()` regrouping output by 'wasteshed' (override with `.groups` ar
gument)

# combining that total
tempWeightData4c <-
  full_join(
    tempWeightData4a,
    tempWeightData4b,
    by=c("wasteshed", "scenario")
  ) %>%
  mutate(pctTons=tons/allTons)

# adding impact categories
tempWeightData4d <-
  left_join(
    tempWeightData4c %>% mutate(dummy=1),
    impactLabels %>% mutate(dummy=1),
    by="dummy"
  ) %>%
  select(-dummy) %>%
  filter(impactCategory != "Energy demand")
```

Now processing the impact data into parallel form…

```
tempImpactData4 <-
  impactsInDetail %>%
  group_by(
    wasteshed,
    scenario,
    material,
    LCstage,
    impactCategory, impactUnits
  ) %>%
  summarise(impact=sum(impact)) %>%
  ungroup() %>%
  mutate(
    scenario = factor(scenario, levels = rev(scenarioOrder)),
    impactLabel =
      paste(
        impactCategory,
        " (",
        impactUnits,
        ")",
        sep=""
      ),
    LCstage=
      factor(
        LCstage,
        levels=c("production","endOfLife","endOfLifeTransport")
      )
  ) %>%
  filter(impactCategory != "Energy demand") #removing to save space

## `summarise()` regrouping output by 'wasteshed', 'scenario', 'material', 'L
Cstage', 'impactCategory' (override with `.groups` argument)

tempImpactData4a <-
  tempImpactData4 %>%
  group_by(
    wasteshed,
    scenario,
    material,
    impactCategory,
    impactUnits,
    impactLabel
  ) %>%
  summarise(impact=sum(impact)) %>%
  ungroup()

## `summarise()` regrouping output by 'wasteshed', 'scenario', 'material', 'i
mpactCategory', 'impactUnits' (override with `.groups` argument)

tempImpactData4b <-
  tempImpactData4a %>%
  group_by(
```

```
    wasteshed, scenario, impactCategory, impactUnits, impactLabel
  ) %>%
  summarise(allImpact=sum(impact)) %>%
  ungroup()
```

```
## `summarise()` regrouping output by 'wasteshed', 'scenario', 'impactCategor
y', 'impactUnits' (override with `.groups` argument)
```

```
tempImpactData4c <-
  full_join(
    tempImpactData4a,
    tempImpactData4b,
    by=c(
      "wasteshed", "scenario", "impactCategory",
      "impactUnits", "impactLabel"
    )
  ) %>%
  mutate(pctImpact=impact/allImpact)
```

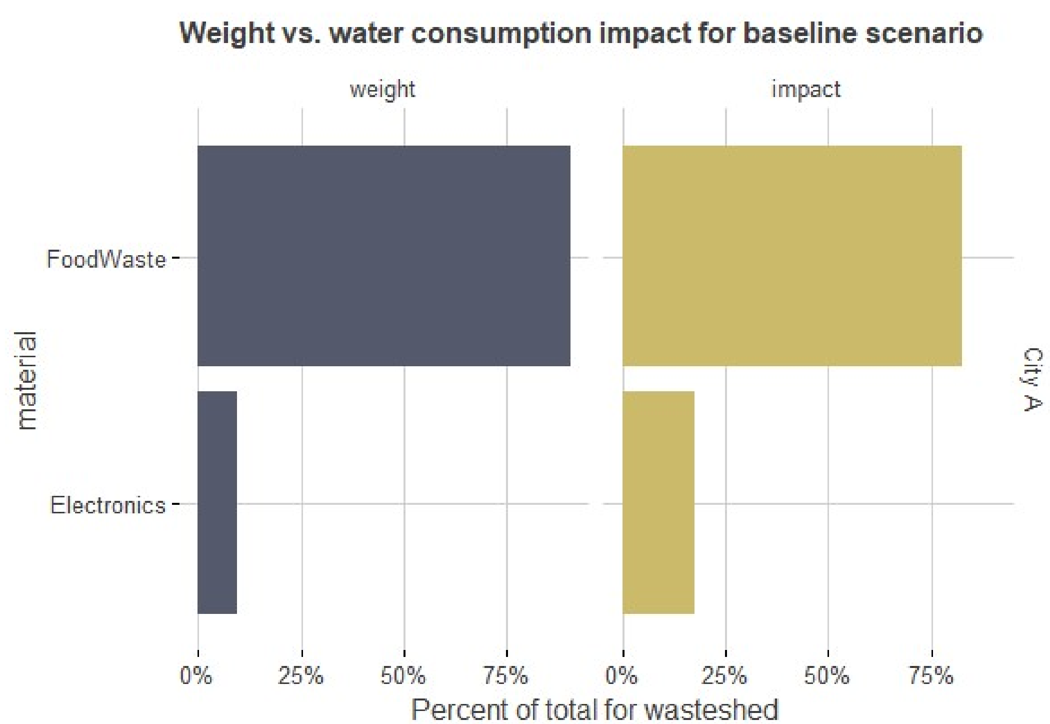
now lining those percentages up into a long, skinny file

```
weightImpactComparisonData1 <-
  bind_rows(
    tempWeightData4d %>%
      select(scenario, wasteshed, material, pctTons, impactLabel) %>%
      rename(pctTotal=pctTons) %>%
      mutate(dataType="weight"),
    tempImpactData4c %>%
      select(scenario, wasteshed, material, pctImpact, impactLabel) %>%
      rename(pctTotal=pctImpact) %>%
      mutate(dataType="impact")
  ) %>%
  arrange(wasteshed, scenario, material, impactLabel)
```

For the chart I'll select only 1 wasteshed and 1 impact category.

```
weightImpactComparisonData2 <-
  filter(
    weightImpactComparisonData1,
    scenario=="baseline",
    impactLabel=="Water consumption (kg)"
  ) %>%
  mutate(dataType=factor(dataType, levels=c("weight", "impact")))
ggplot()+
  ggtitle("Weight vs. water consumption impact for baseline scenario")+
  theme_539()+
  geom_bar(
    data=weightImpactComparisonData2,
    aes(x=material, y=pctTotal, fill=dataType),
    stat="identity",
    color=NA
```

```r
  )+
  scale_y_continuous(
    name="Percent of total for wasteshed",
    labels=percent
  )+
  scale_fill_viridis(begin = 0.32, end=0.8,
                     discrete=TRUE,
                     option="cividis",
                     direction=1)+
  facet_grid(wasteshed~dataType)+
  coord_flip()+
  theme(
    rect=element_rect(fill="transparent"),
    axis.ticks=element_line(),
    legend.position="none",
    axis.title=element_text(),
    plot.title=element_text(size=12)
  )
```




```r
ggsave("chart_output/weight_impact_comparison.png")

## Saving 6.5 x 4.5 in image

ggsave("chart_output/weight_impact_comparison.svg")

## Saving 6.5 x 4.5 in image
```

This kind of chart often ends up exposing the way weight can be a poor proxy for life cycle impacts. Above, though electronics are a small portion of weight, they are a noticeably large portion of impacts.

Alternative formats

DEQ can provide documents in an alternate format or in a language other than English upon request. Call DEQ at 800-452-4011 or email deqinfo@deq.state.or.us.