

carbon_model.rb

Copyright © 2010 Brighter Planet. See LICENSE for details. Contact Brighter Planet for dual-license arrangements.

(grams CO2 / kcal food) * (kcal food)

```
module BrighterPlanet
  module Diet
    module CarbonModel
      def self.included(base)
        base.decide :emission, :with =>
          :characteristics do
            committee :emission do # returns kg
              CO2

              quorum 'from intensity and size',
:needs => [:intensity, :size,
:active_subtimeframe] do |characteristics|

                (characteristics[:intensity] *
characteristics[:size]).grams.to(:kilograms) *
characteristics[:active_subtimeframe].days
              end
              quorum 'default' do
                raise 'Diet emission committee
default quorum should never be called'
              end
            end
          end

          committee :intensity do # returns
            grams CO2 / kcal food
            quorum 'from food groups', :needs
=> [:red_meat_share, :poultry_share,
:fish_share, :eggs_share, :nuts_share,
:dairy_share, :cereals_and_grains_share,
:fruit_share, :vegetables_share,
:oils_and_sugars_share] do |characteristics|
```

```

        [:red_meat, :poultry, :fish,
:eggs, :nuts, :dairy, :cereals_and_grains,
:fruit, :vegetables, :oils_and_sugars].sum do
|group|

characteristics["#{group}_share".to_sym].to_f *
FoodGroup[group].intensity.to_f
        end
    end

    quorum 'from diet class', :needs =>
:diet_class do |characteristics|

characteristics[:diet_class].intensity
        end
    end

    committee :size do # returns kcal
food

        quorum 'default' do
            base.fallback.size
        end
    end

    committee :diet_class do
        quorum 'default' do
            DietClass.fallback
        end
    end

    committee :active_subtimeframe do
        quorum 'from start and end dates',
:needs => [:start_date, :end_date] do
|characteristics, timeframe|
            Timeframe.constrained_new
characteristics[:start_date],

```

```
characteristics[:end_date], timeframe
    end
end

    committee :start_date do
        quorum 'from end date',
:appreciates => :end_date do |characteristics,
timeframe|
            [ timeframe.from,
characteristics[:end_date] ].compact.min
        end
    end

    committee :end_date do
        quorum 'from start date',
:appreciates => :start_date do
|characteristics, timeframe|
            [ timeframe.to,
characteristics[:start_date] ].compact.max
        end
    end
end
end
end
end
end
end
```