International Institute for
Applied Systems Analysis
IIASA www.iiasa.ac.at

Partial Equilibrium Modeling Training
FABLE Workshop
IIASA, June 4 – June 5, 2018

GLOBIOM
Equations, Variables, Calibration

Petr Havlík & Hugo Valin
havlikpt@iiasa.ac.at

science for global insight

Overview of GLOBIOM equations

- Different types of equations
  - Objective equation: producer + consumer surplus
  - Linearisation and convexity equation
    - Non linear variable <= weighted sum of closest fixed points
  - Balance equations:
    - supply >= demand
    - new land use >= previous land use + land expansion
  - Flexibity constraint equations
    - crop new >= 50% crop previous
  - Resource constraints:
    - water available >= water use
  - Accounting equations
    - GHG Emissions = Activity Level * GHG Emissions Coefficient

Key model equations

* Objective function	
OBJECTIVE_EQU	Maximization of the sum of producer and consumer surplus [mio USD]
* Separable programming equations	
DEMAND_IDENTITY	Separable Demand Variable Identities
DEMAND_CONVEXITY	Separable Demand Variable Convexities
RESOURCE_IDENTITY	Separable Resource Variable Convexities
RESOURCE_CONVEXITY	Separable Resource Variable Identities
LUCDET_IDENTITY	Separable Land Use Variable Identities
LUCDET_CONVEXITY	Separable Land Use Variable Convexities
TRADECOST_IDENTITY	Separable Trade Cost Variable Identities
TRADECOST_CONVEXITY	Separable Trade Cost Variable Convexities
TRANSPORT_IDENTITY	Separable Local Trade Cost Variable Identies
TRANSPORT_CONVEXITY	Separable Local Trade Cost Variable Convexities
* Market balance	
DS_BALANCE_EQU	Market balances [1000 tonnes or 1000 m3]

Crop production

- 18 crops in 4 management systems

- Variables

  CROP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH)

  - crop area [1000 ha]

- Parameters

  CROP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH,ALLITEM)

  - Base area, yield, cost, N/P requirements, GHG coefficients

Crop production

- Equations

SUBSFARMING_EQU	Fixes the subsistence farming area to the observed levels in the base year [1000 ha]
CROPLANDUSE_EQU	Keeps low input and high input management systems proportional to each other as observed in the base year at the regional level [1000 ha]
CROPLANDUSE2_EQU	Keeps low input and high input management systems proportional to each other as observed in the base year at the Simulation Unit and crop level [1000 ha]
MINCROP_EQU	Limits the relative decrease in crop area by crop and management system compared to the previous period [1000 ha]
MAXCROP_EQU	Limits the realtive increase in individual crops compared to the previous period [1000 ha]
MAXCROPSYS_EQU	Limits the realtive increase in management system compared to the previous period [1000 ha]

Livestock production – Herd dynamics

- 10 livestock types in 8 production systems

- Variables

  `LIVE_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMALS)`

  - Livestock numbers [1000 TLU]

- Parameters

  `LIVE_DATA(COUNTRY,LIVE_SYSTEM,ANIMALS,AllITEM)`

  - Feed requirements, Milk and meat yield, GHG coefficients

Livestock production – Herd dynamics

DAIRYHERD_EQU	Controls the proportion of replacement females in dairy ruminant herds [1000 TLUs]
NOSUCKLERHERD_EQU	Keeps the proportions of dairy and non-dairy ruminants at the base year level in regions where mixed purpose herds assumed [1000 TLUs]
HERDFIXED_EQU	Keeps the proportion of dairy and non-dairy small ruminants at the base year level [1000 TLU]
MINLIVESTOCK_EQU	Limits the relative decrease in livestock numbers by animal type and production system compared to the previous period [1000 TLUs]
MAXLIVESTOCK_EQU	Limits the relative increase in livestock numbers by animal type and production system compared to the previous period [1000 TLUs]
TRANSFORM_EQU	Auxiliary calculation of LIVE_VAR2 based on LIVE_VAR for more efficient computation in some constraints
MINLIVESTOCK2_EQU	Limits the relative decrease in livestock numbers by animal type compared to the previous period [1000 TLUs]
MAXLIVESTOCK2_EQU	Limits the relative increase in livestock numbers by animal type compared to the previous period [1000 TLUs]
MNGASTOTHER_EQU	Keeps by assumption the number of pigs and poultry in backyard systems at the base year levels - all growth comes from industrial systems [1000 TLU]
RUMURBAN_EQU	Keeps by assumption the number of ruminants in urban and "other" systems at the base year levels [1000 TLU]
LIVEFIXED_EQU	Keeps the relative distribution of ruminants across production systems within a country at the level from previous period - scenario constraint [1000 TLU]
LIVEFIXEDSIMU_EQU	Keeps the relative distribution of ruminants across production systems within a "SimU" at the level from previous period - scenario constraint [1000]

Livestock production – Feed

- Feed aggregates: Grains, Grass, Stover and Occasional
- Feed grains: All 18 crops

- Variables

  FEEDQUANTITY(REGION,CROPS)

  - Total feed demand for crops [1000 ton]

  GRAS_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)

  - Utilized grassland area [1000 ha]

- Parameters

  GRAS_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ALLITEM)

  - Base area and yield

Livestock production - Feed

- Equations

FEED_BALANCE_EQU	Balances grain feed requirements at the feed aggregates level [1000 tonnes DM]
FEED_BALANCE_FIXEDCROPS_EQU	Keeps the proportion of individual crops in the feed aggregates level at the base year level [1000 tonnes DM]
STOVER_BALANCE_EQU	Stover demand-supply balance [1000 tonnes DM]
GRAS_BALANCE_EQU	Grass demand-supply balance [1000 tonnes DM]
GRAS_BALANCE2_EQU	Grass & occasional demand supply balance [1000 tonnes DM]

Forest production – Biomass

- Biomass types: Sawn wood, Other wood, Branches & Stumps, Harvest losses
- Biomass uses: Saw logs, Pulp logs, Other industrial round wood, Fuel wood

- Variables

  HARVEST_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMng Type)

  Area of forest harvested during the rotation time [1000 ha]

  SQUANTITY_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS, FORMNGTYPE,PRODUCT)

  - Harvested quantity of a particular biomass grade[1000 ha]
- Parameters

  FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,FORMNG TYPE,ALLITEM)

  - Mean annual increment, proportion of different types of biomass, carbon balance

Forest production - Biomass

- Equations

MAXSAWLOG_EQU	Limits amount of harvested sawlogs [1000 m3]
MAXLOG_EQU	Limits amount of harvested sawlogs, pulplogs and other industrial round wood [1000 m3]'
MAXWOODBIOMASS_EQU	Matches the sum of all biomass grades with total harvested woody biomass [1000 m3]
MAXHARVESTLOSS_EQU	Calculates share of harvest losses proportional to sawlogs, pulplogs and other industrial round wood [1000 m3]
MAXBRANCHES_EQU	Limits amount of branches and stumps [1000 m3]
MAXLOGGINGRESIDUES_EQU	Limits amount of logging residues [1000 m3]
DEFORLOGS_EQU	Accounts for fuel wood supply from deforestationbalance of logs from deforestation [1000 m3]
WOODPELLETS_EQU	Allows to impose depending on scenario minimum wood pellets production
MAXRECYCLEDWOOD_EQU	Accounts for logging residues availability as sum of harvet losses and stumps and branches [1000 m3]

Forest production – Industry

- Industry products: Sawnwood, Mechanical and Chemical Pulp, Plywood, Fiberboard, Woodpellets, Sawdust, Woodchips, Black liquor
- Variables

  PQUANTITY(REGION,PROCESS)

  Quantity of processed primary product [1000 m3]

  CAPACITY_VAR(REGION,PRODUCT)

  Processing capacity of main final products [1000 m3 or 1000 t]
- Parameters

  PROCESSDATA(REGION,PROCESS,PRODUCT)

  - Input-output relationships between primary and final products and processing cost

Forest production - Industry

- Equations

CAPACITY_EQU	Accounts for forest product processing capacity availabiliy [1000 m3]
CAPACITYACCUM_EQU	Accounts for forest product processing capacity dynamics [1000 m3]
CAPACITYREPLAC_EQU	Accounts for forest product processing capacity replacement needs [1000 m3]
MINTRADE1_EQU	Limits trade in forest products to levels from base period [1000 m3]
MINTRADE2_EQU	Requires trade in forest products equal or larger than base period levels [1000 m3]
TRANSPORT_ZERO_EQU	Prevents production to appear during simulation when no data for the base year [1000 m3]

Natural resources – Land

- Land cover types: Cropland, Grassland, Short rotation plantations, Managed forest, Unmanaged forest, Other natural vegetation
- Variables

  SRP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)

  Short rotation plantations area [1000 ha]

  LANDAVAIL_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)

  Land cover/use area [1000 ha]

  LUCDET_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT,LC_CRNT)

  Land use change [1000 ha]

  RESOURCE_VAR(REGION,'LAND')

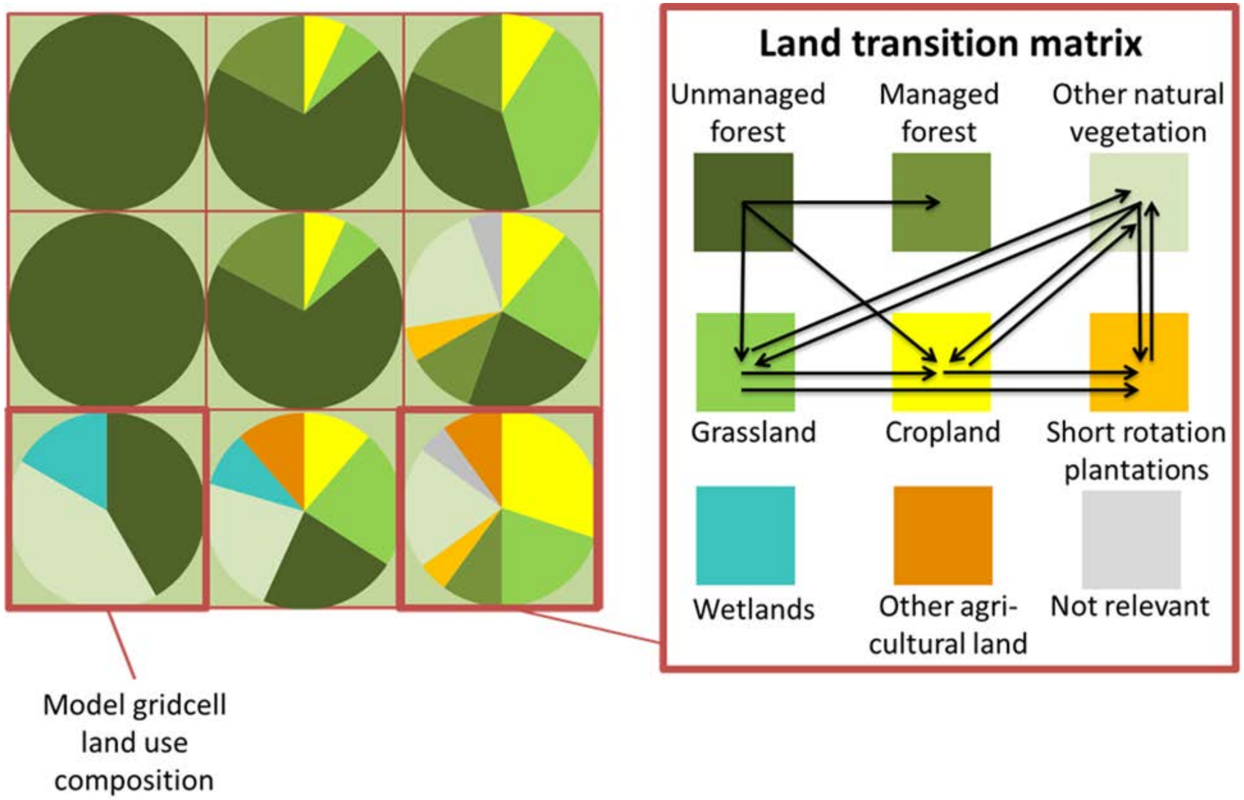
  Cultivated land – cropland, grassland, SRP [1000 ha]

Natural resources - Land

- Equations

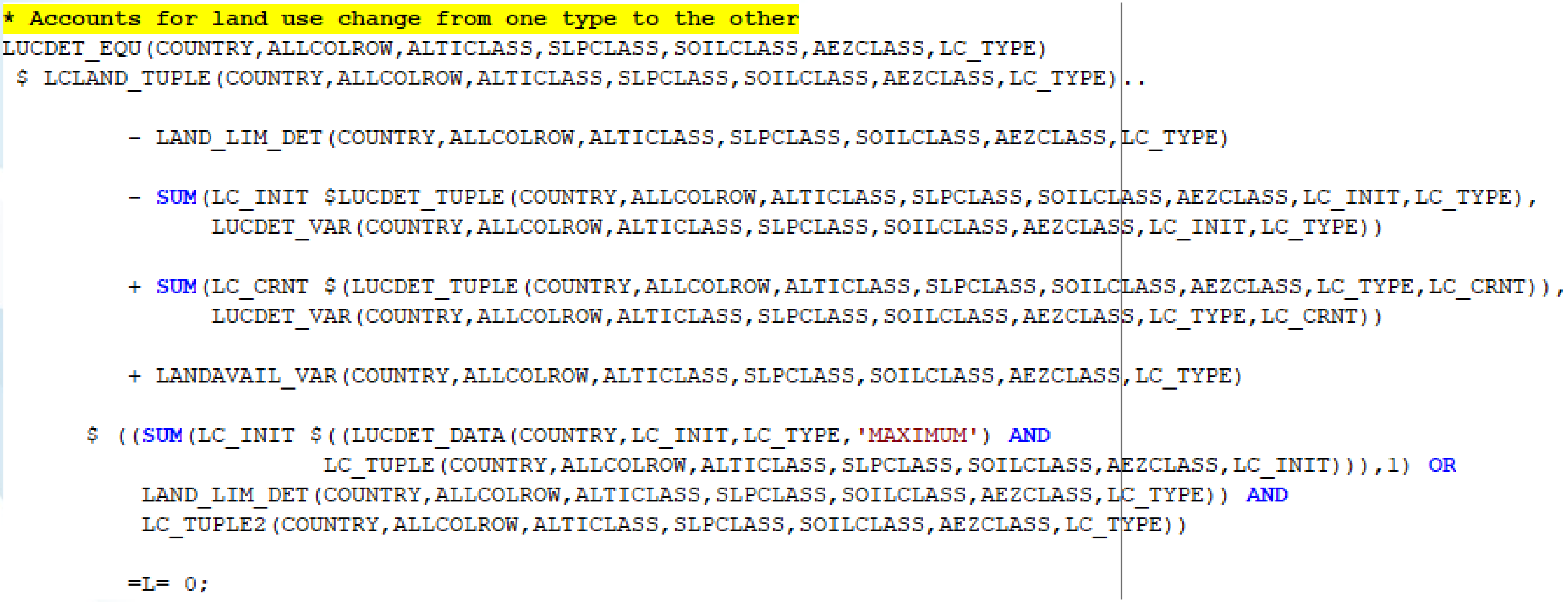
CROPLAND_EQU	Matches physical crop areas with total cropland [1000 ha]
GRASLAND_EQU	Matches grassland use with available grassland [1000 ha]
SRPLAND_EQU	Matches short rotation plantations area with available land [1000 ha]
HARVLAND_EQU	Matches harvested forest area with the managed forest area [1000 ha]
LUCDET_EQU	Accounts for land use change from one type to the other [1000 ha])
SRPSUIT_EQU	Limits area that can be converted to short rotation plantations according to suitability [1000 ha]
LAND_ACCOUNT_EQU	Accounts for overall resource land use as sum of cropland incl. perrenial crops, grassland, and short rotation plantations [1000 ha]

Land cover change



- Land cover change endogenous depending on relative profitability
- Conversion implies a conversion cost
- Max conversion rates can be capped to mimic policy/social constraints

Land cover change: Equation



```
* Accounts for land use change from one type to the other
LUCDET_EQU(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)
 $ LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)..

         - LAND_LIM_DET(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)

         - SUM(LC_INIT $LUCDET_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT,LC_TYPE),
               LUCDET_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT,LC_TYPE))

         + SUM(LC_CRNT $(LUCDET_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE,LC_CRNT)),
               LUCDET_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE,LC_CRNT))

         + LANDAVAIL_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)

      $ ((SUM(LC_INIT $((LUCDET_DATA(COUNTRY,LC_INIT,LC_TYPE,'MAXIMUM') AND
                        LC_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT))),1) OR
          LAND_LIM_DET(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE)) AND
          LC_TUPLE2(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_TYPE))

         =L= 0;
```

Natural resources - Water

- Variables

  `RESOURCE_VAR(REGION,'LAND')`

  Irrigation water use [km3]

- Equations

WATER_ACCOUNT_EQU	Accounts for irrigation water use [km3]

Bioenergy demand

- Bioenergy types: Biomass for 1st and 2nd generation biofuels, power & heat cogeneration, direct biomass use for energy
- Variables

  PQuantity(REGION,PRODUCT)

  1st generation biofuel feedstock processing [1000 tonnes]

  DQuantity(REGION,PRODUCT)

  Demand for bioenergy products [TJ]

- Parameters

  ABS_FUEL_REQ(ANYREGION,PRODUCT)

  1st gen biofuel feedstock processing target [1000 tonnes]

  BIOEN_SCEN(ANYREGION,POLES_AGGENG)

  Target for bioenergy demand [TJ]

- Equations

BIOEN_DEMAND_EQU	Base processed quantity of feedstoc [1000] tonnes
BIOEN_SCEN_EQU	Bioenergy Production as defined in NRG scenarios [1000 GJ]

GHG emissions - Accounting

Sector	Source	GHG	Reference
Land use change	Deforestation	CO_2	FRA 2005 carbon in above ground and below ground living biomass downscaled at 0.5 degree (Kindermann et al. 2008)
	Conversion of other vegetation types	CO_2	Ruesch and Gibbs (2008)
Crops	Fertilizer use	N_2O	Requirements from EPIC/IFA, emission coefficients from IPCC
	Rice production	CH_4	IPCC Tier 1 approach
Livestock	Enteric fermentation	CH_4	RUMINANT model (Herrero et al. 2008)/IPCC
	Manure management	N_2O, CH_4	RUMINANT model (Herrero et al. 2008)/IPCC
	Manure dropped/applied to pastures/cropland	N_2O	RUMINANT model (Herrero et al. 2008)/IPCC

GHG emissions - Accounting

- Variables

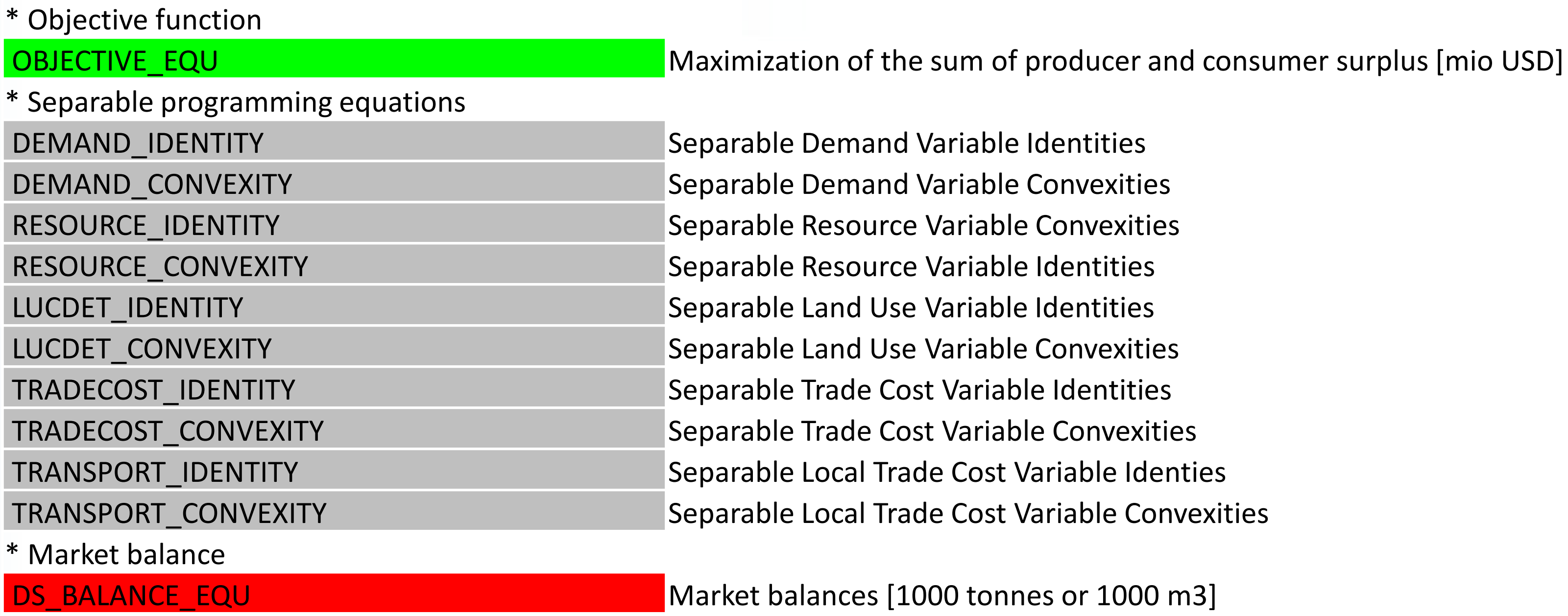
  EMISSION_VAR(REGION,GHGACCOUNT)

  AFOLU GHG emissions [Million tonnes CO2 equivalent]

- Equations

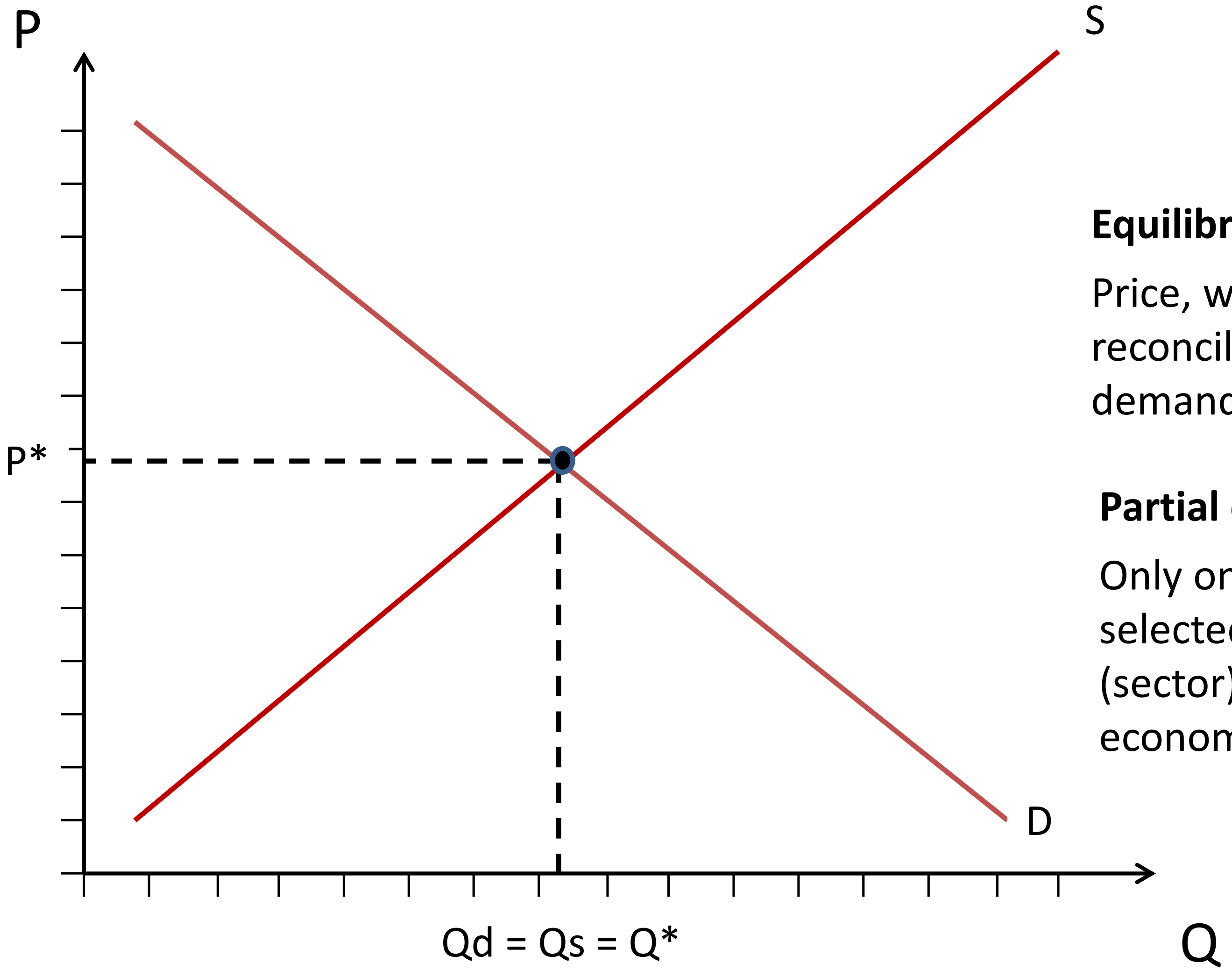
EMISSION_EQU	GHG emission accounting equation [MtCO2eq]

Key model equations



* Objective function	
OBJECTIVE_EQU	Maximization of the sum of producer and consumer surplus [mio USD]
* Separable programming equations	
DEMAND_IDENTITY	Separable Demand Variable Identities
DEMAND_CONVEXITY	Separable Demand Variable Convexities
RESOURCE_IDENTITY	Separable Resource Variable Convexities
RESOURCE_CONVEXITY	Separable Resource Variable Identities
LUCDET_IDENTITY	Separable Land Use Variable Identities
LUCDET_CONVEXITY	Separable Land Use Variable Convexities
TRADECOST_IDENTITY	Separable Trade Cost Variable Identities
TRADECOST_CONVEXITY	Separable Trade Cost Variable Convexities
TRANSPORT_IDENTITY	Separable Local Trade Cost Variable Identies
TRANSPORT_CONVEXITY	Separable Local Trade Cost Variable Convexities
* Market balance	
DS_BALANCE_EQU	Market balances [1000 tonnes or 1000 m3]

Market equilibrium / partial equilibrium



Equilibrium price:

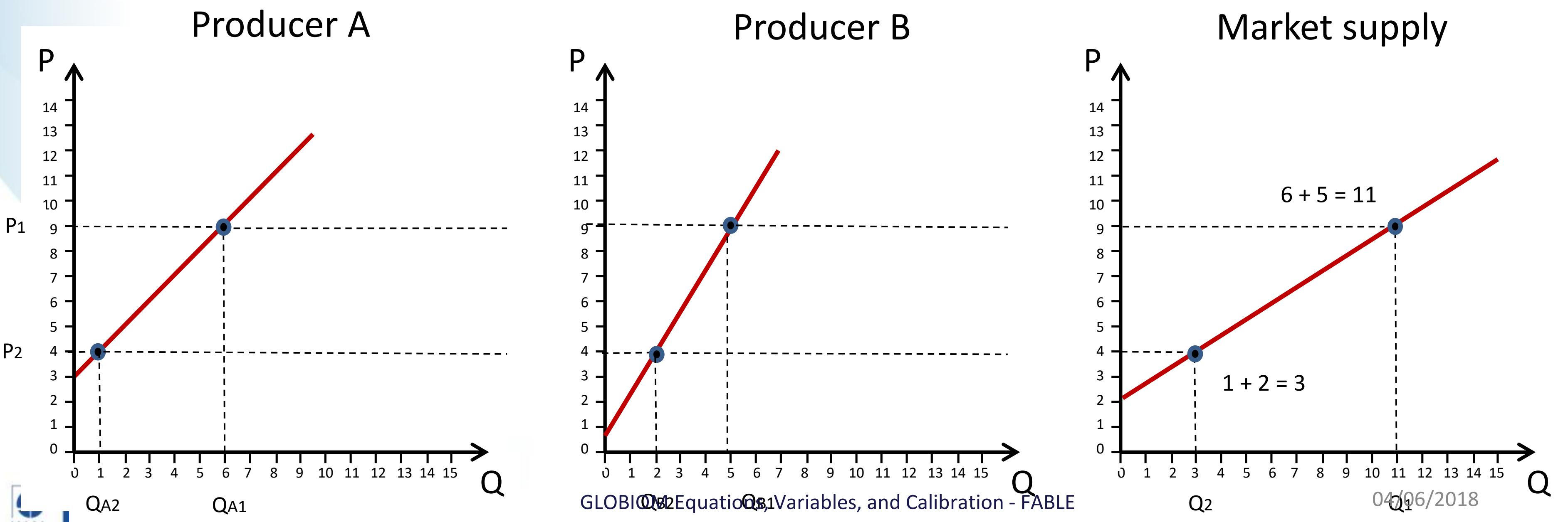
Price, which reconciles supply and demand

Partial equilibrium:

Only one market or selected markets (sector) of an economy

Supply

- Producers (enterprises) maximize profits, taking into account
  - Product prices, prices of other products
  - Factor (input) prices
  - Technology/ technological progress
  - Subsidies and other policies
  - Weather, …



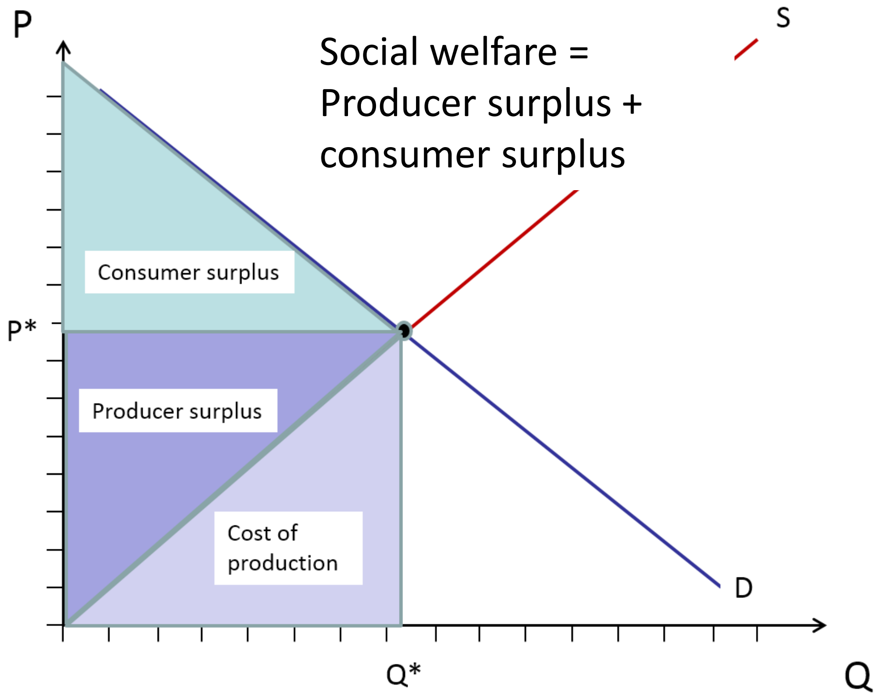
Producer A

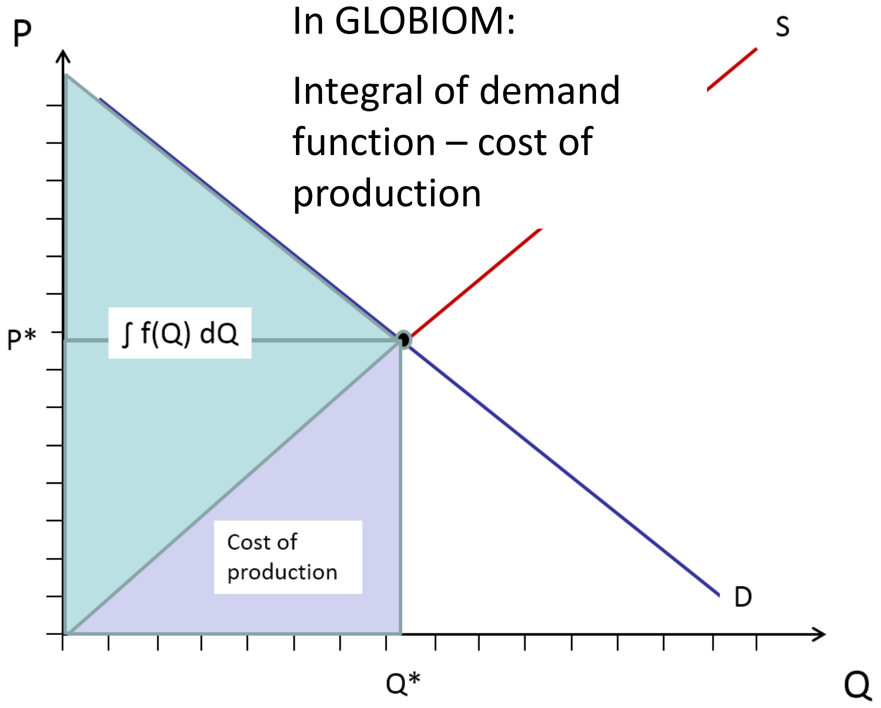
Producer B

Market supply

PE as optimization problem (McCarl and Spreen, 1980)

- Maximizing social welfare (under constraints)
- Prices are represented as dual variables

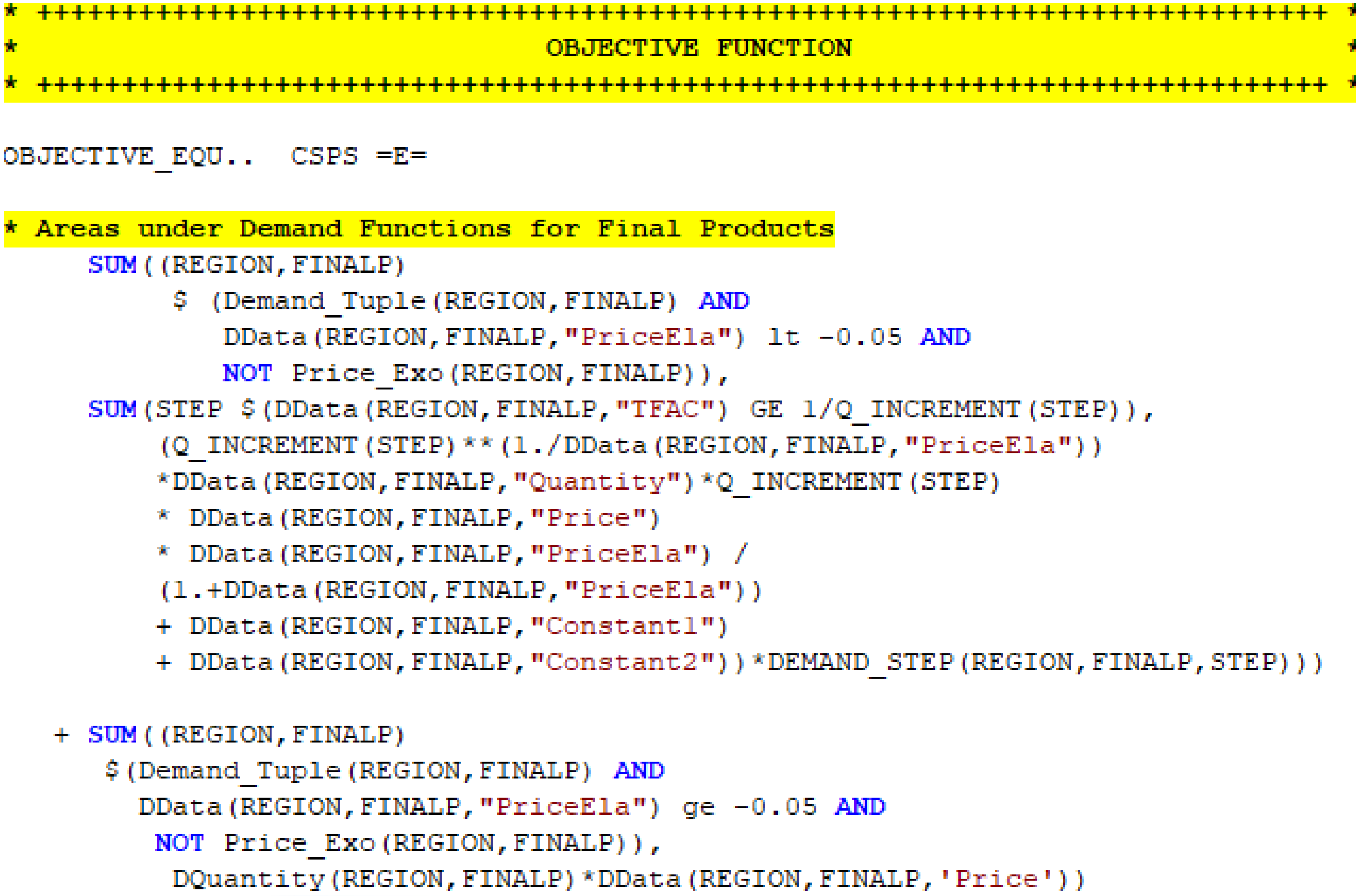

Social welfare = Producer surplus + consumer surplus


In GLOBIOM:

Integral of demand function – cost of production

Objective function

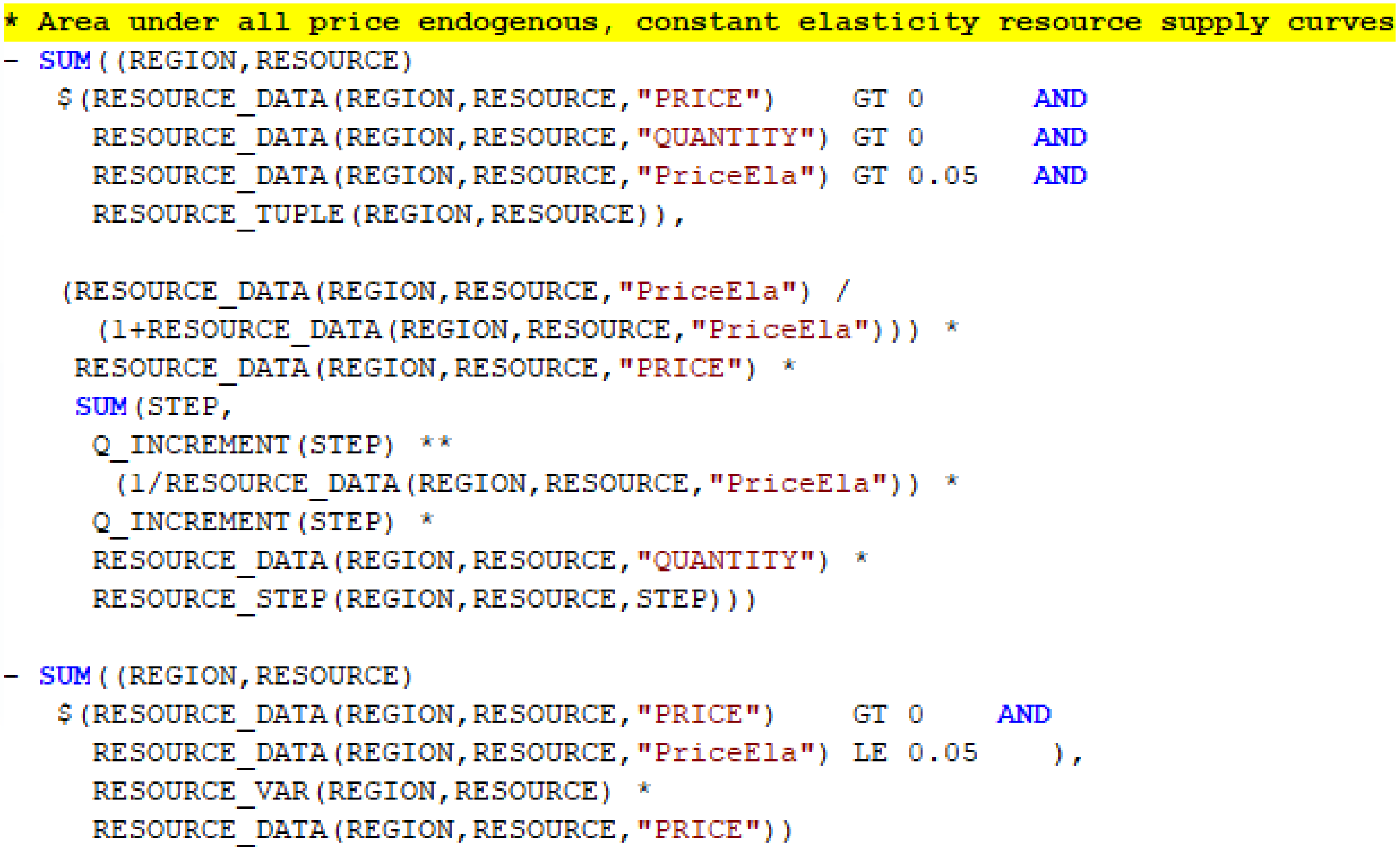
- Demand function integral → Consumer surplus



```
* ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ *
*                               OBJECTIVE FUNCTION                             *
* ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ *

OBJECTIVE_EQU..  CSPS =E=

* Areas under Demand Functions for Final Products
     SUM((REGION,FINALP)
          $  (Demand_Tuple(REGION,FINALP) AND
             DData(REGION,FINALP,"PriceEla") lt -0.05 AND
             NOT Price_Exo(REGION,FINALP)),
     SUM(STEP $(DData(REGION,FINALP,"TFAC") GE 1/Q_INCREMENT(STEP)),
         (Q_INCREMENT(STEP)**(1./DData(REGION,FINALP,"PriceEla"))
         *DData(REGION,FINALP,"Quantity")*Q_INCREMENT(STEP)
         *  DData(REGION,FINALP,"Price")
         *  DData(REGION,FINALP,"PriceEla") /
         (1.+DData(REGION,FINALP,"PriceEla"))
         +  DData(REGION,FINALP,"Constant1")
         +  DData(REGION,FINALP,"Constant2"))*DEMAND_STEP(REGION,FINALP,STEP)))

   + SUM((REGION,FINALP)
      $(Demand_Tuple(REGION,FINALP) AND
        DData(REGION,FINALP,"PriceEla") ge -0.05 AND
         NOT Price_Exo(REGION,FINALP)),
          DQuantity(REGION,FINALP)*DData(REGION,FINALP,'Price'))
```

Objective function

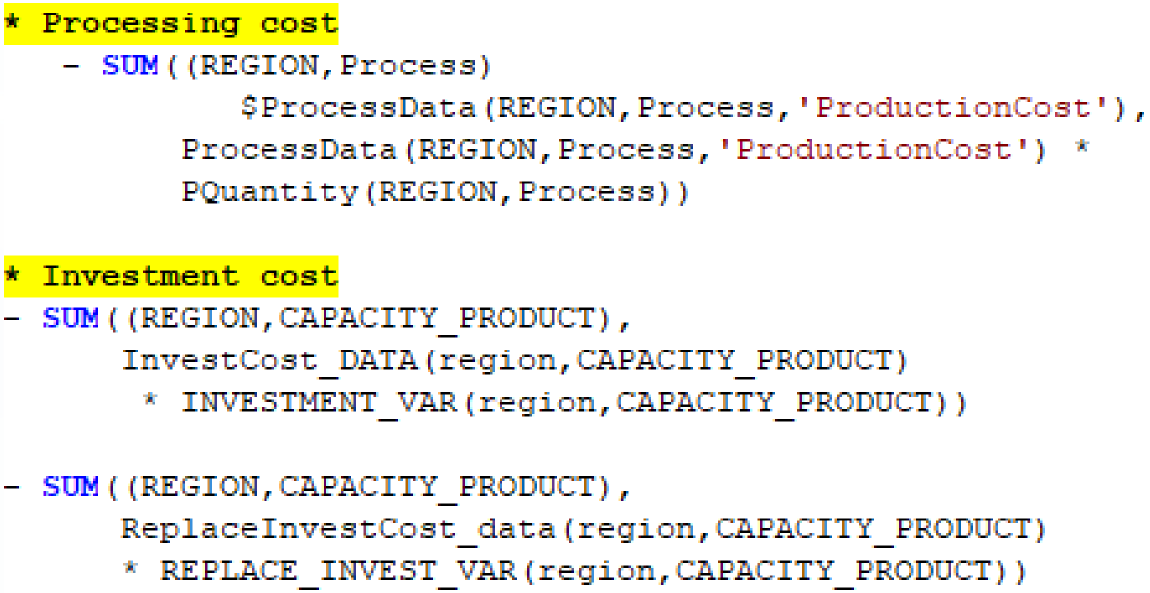
- Resource cost - Resource supply function integral



```
* Area under all price endogenous, constant elasticity resource supply curves
- SUM((REGION,RESOURCE)
   $(RESOURCE_DATA(REGION,RESOURCE,"PRICE")    GT 0      AND
     RESOURCE_DATA(REGION,RESOURCE,"QUANTITY") GT 0      AND
     RESOURCE_DATA(REGION,RESOURCE,"PriceEla") GT 0.05   AND
     RESOURCE_TUPLE(REGION,RESOURCE)),

   (RESOURCE_DATA(REGION,RESOURCE,"PriceEla") /
     (1+RESOURCE_DATA(REGION,RESOURCE,"PriceEla"))) *
    RESOURCE_DATA(REGION,RESOURCE,"PRICE") *
    SUM(STEP,
     Q_INCREMENT(STEP) **
      (1/RESOURCE_DATA(REGION,RESOURCE,"PriceEla")) *
     Q_INCREMENT(STEP) *
     RESOURCE_DATA(REGION,RESOURCE,"QUANTITY") *
     RESOURCE_STEP(REGION,RESOURCE,STEP)))

- SUM((REGION,RESOURCE)
   $(RESOURCE_DATA(REGION,RESOURCE,"PRICE")    GT 0    AND
     RESOURCE_DATA(REGION,RESOURCE,"PriceEla") LE 0.05    ),
     RESOURCE_VAR(REGION,RESOURCE) *
     RESOURCE_DATA(REGION,RESOURCE,"PRICE"))
```

Objective function

- Processing and forest industry investment cost

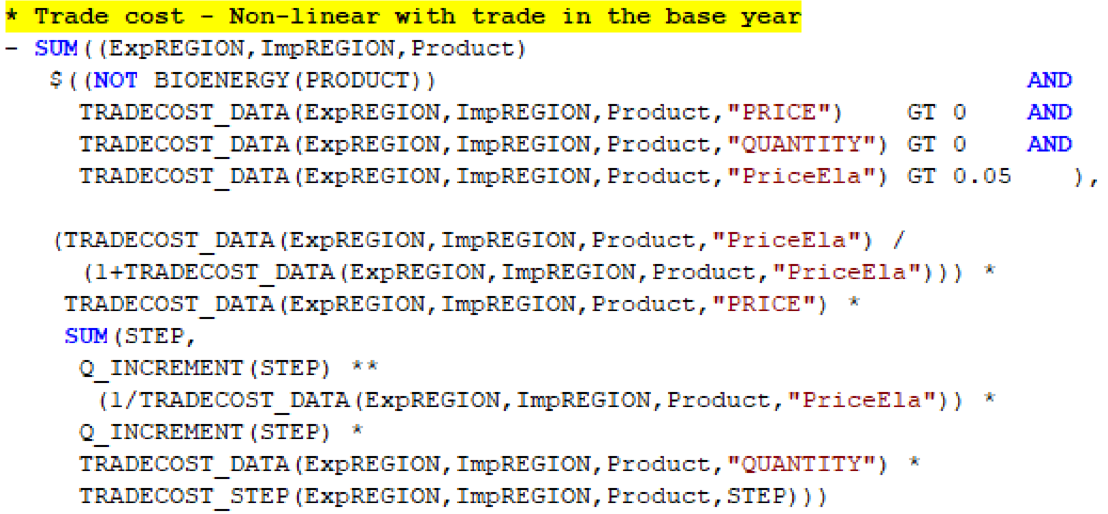


```
* Processing cost
   - SUM((REGION,Process)
           $ProcessData(REGION,Process,'ProductionCost'),
        ProcessData(REGION,Process,'ProductionCost')  *
        PQuantity(REGION,Process))

* Investment cost
- SUM((REGION,CAPACITY_PRODUCT),
     InvestCost_DATA(region,CAPACITY_PRODUCT)
      *  INVESTMENT_VAR(region,CAPACITY_PRODUCT))

- SUM((REGION,CAPACITY_PRODUCT),
     ReplaceInvestCost_data(region,CAPACITY_PRODUCT)
     *  REPLACE_INVEST_VAR(region,CAPACITY_PRODUCT))
```

Objective function

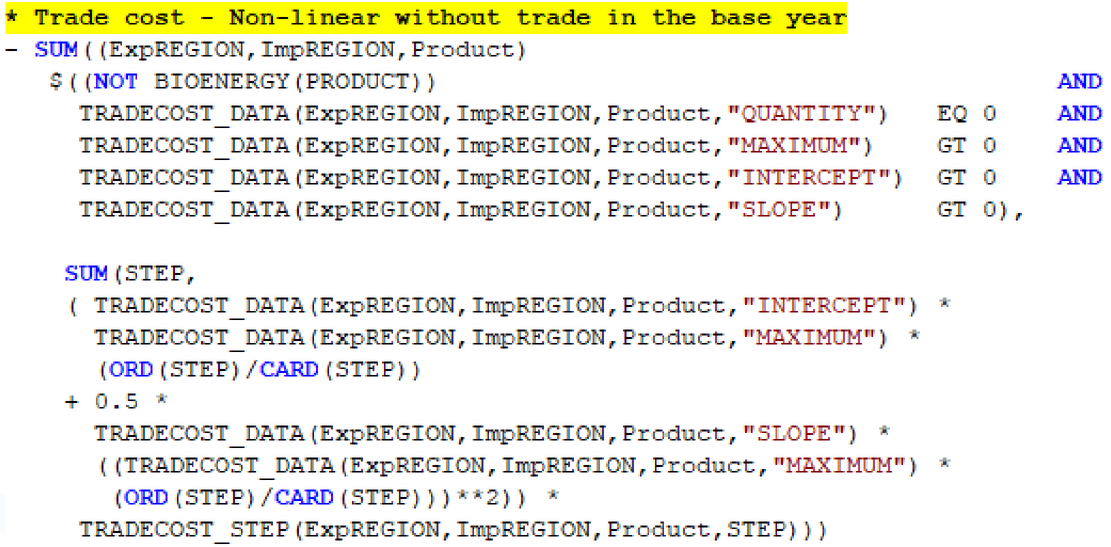
- Trade cost

Constant elasticity cost function



```
* Trade cost - Non-linear with trade in the base year
- SUM((ExpREGION,ImpREGION,Product)
   $((NOT BIOENERGY(PRODUCT))                                       AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PRICE")     GT 0     AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"QUANTITY")  GT 0     AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PriceEla")  GT 0.05     ),

   (TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PriceEla") /
     (1+TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PriceEla"))) *
    TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PRICE")  *
    SUM(STEP,
     Q_INCREMENT(STEP)  **
      (1/TRADECOST_DATA(ExpREGION,ImpREGION,Product,"PriceEla")) *
     Q_INCREMENT(STEP)  *
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"QUANTITY")  *
     TRADECOST_STEP(ExpREGION,ImpREGION,Product,STEP)))
```

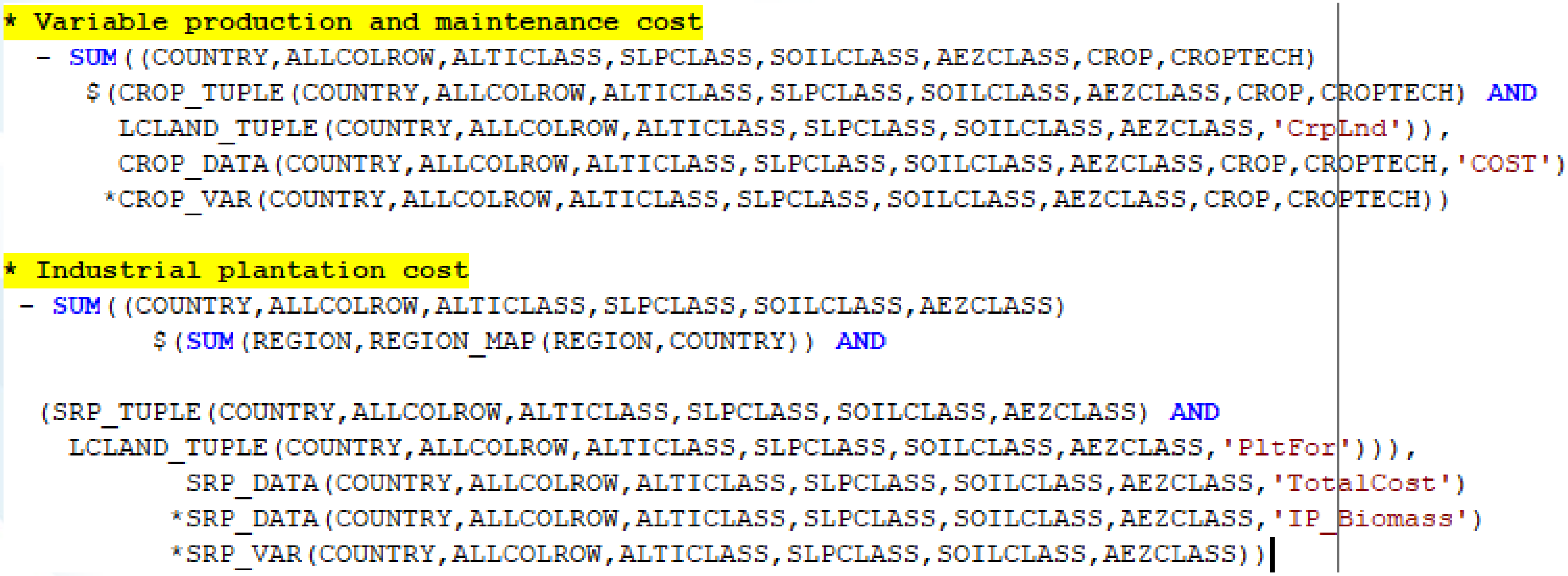
Quadratic cost function



```
* Trade cost - Non-linear without trade in the base year
- SUM((ExpREGION,ImpREGION,Product)
   $((NOT BIOENERGY(PRODUCT))                                         AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"QUANTITY")   EQ 0     AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"MAXIMUM")    GT 0     AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"INTERCEPT")  GT 0     AND
     TRADECOST_DATA(ExpREGION,ImpREGION,Product,"SLOPE")      GT 0),

    SUM(STEP,
    ( TRADECOST_DATA(ExpREGION,ImpREGION,Product,"INTERCEPT")  *
      TRADECOST_DATA(ExpREGION,ImpREGION,Product,"MAXIMUM")  *
      (ORD(STEP)/CARD(STEP))
    + 0.5 *
      TRADECOST_DATA(ExpREGION,ImpREGION,Product,"SLOPE")  *
      ((TRADECOST_DATA(ExpREGION,ImpREGION,Product,"MAXIMUM")  *
       (ORD(STEP)/CARD(STEP)))**2))  *
     TRADECOST_STEP(ExpREGION,ImpREGION,Product,STEP)))
```

Objective function

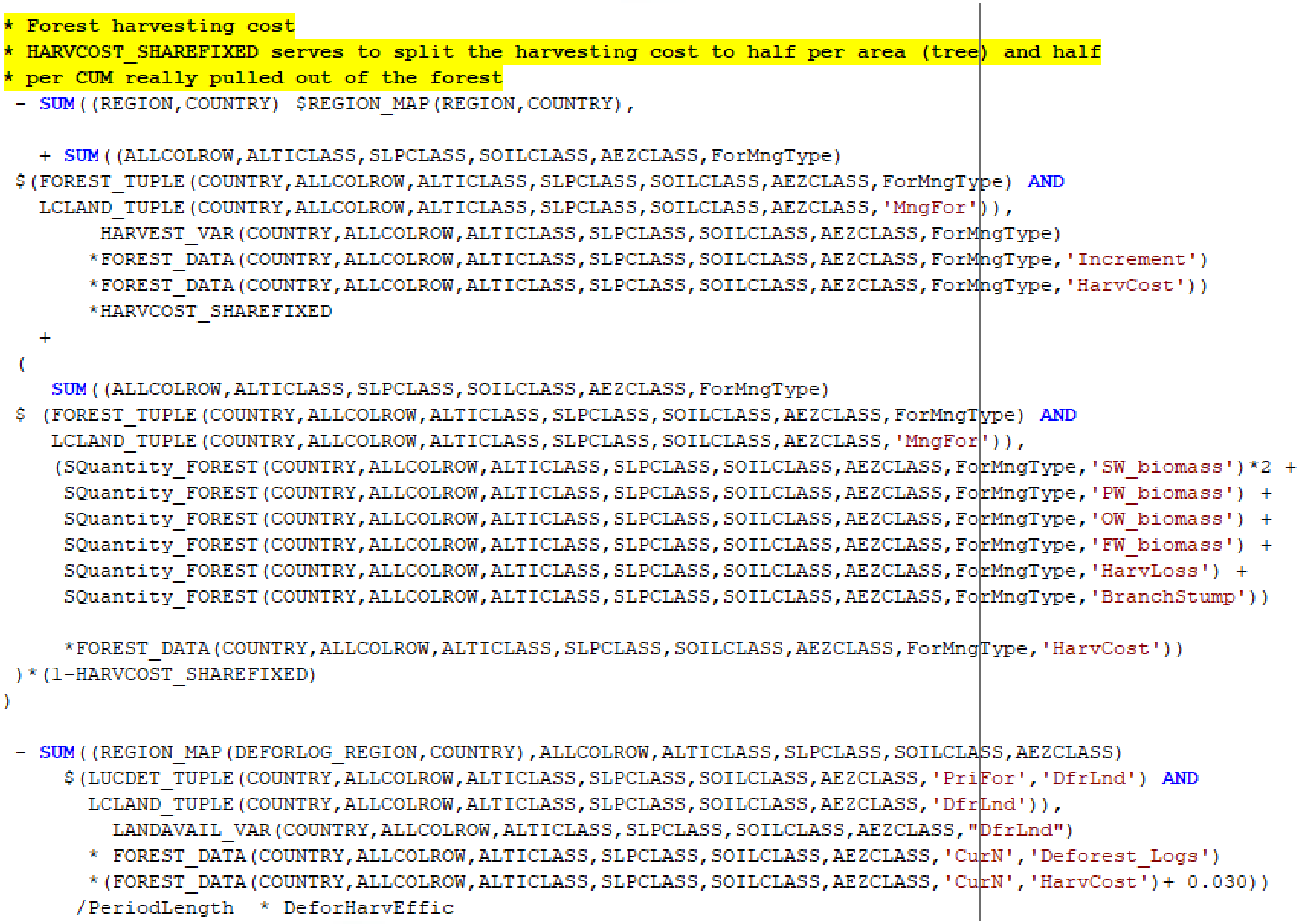
- Crop and plantations cost



```
* Variable production and maintenance cost
  - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH)
     $(CROP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH) AND
       LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CrpLnd')),
       CROP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH,'COST')
      *CROP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH))

* Industrial plantation cost
 - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)
         $(SUM(REGION,REGION_MAP(REGION,COUNTRY)) AND

  (SRP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS) AND
    LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'PltFor'))),
           SRP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'TotalCost')
          *SRP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'IP_Biomass')
          *SRP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS))
```

Objective function

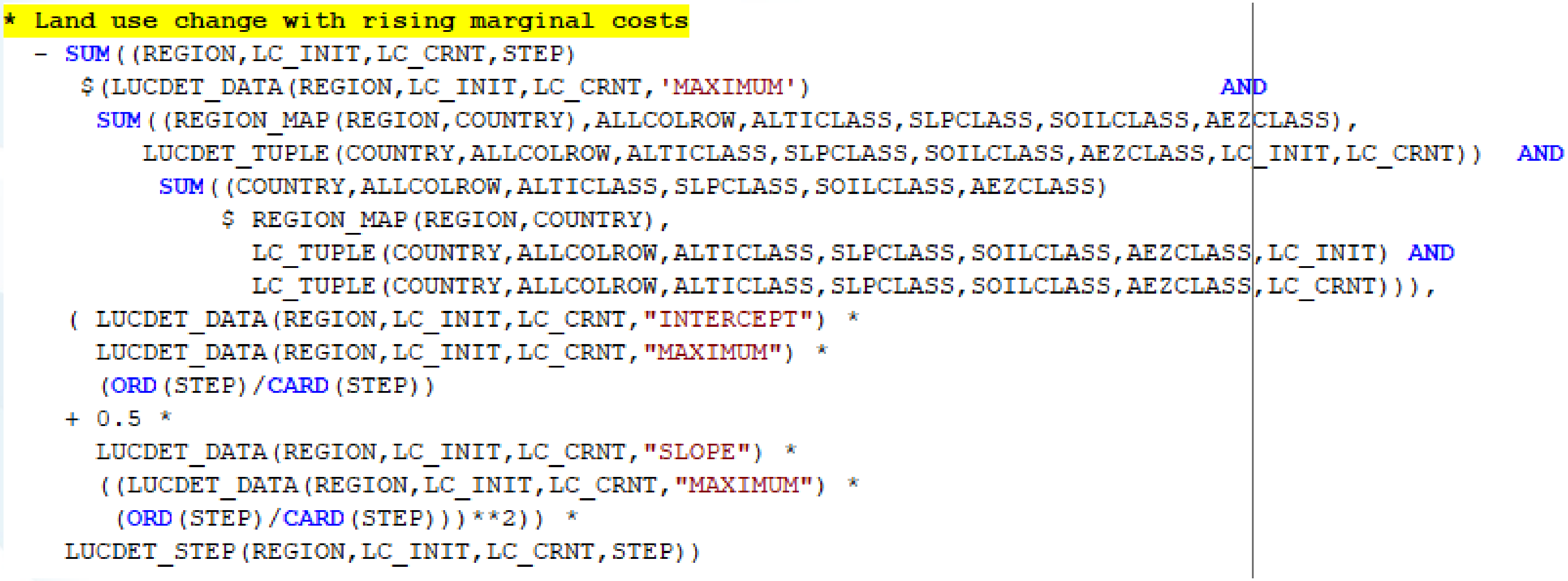
- Forest harvesting cost



```
* Forest harvesting cost
* HARVCOST_SHAREFIXED serves to split the harvesting cost to half per area (tree) and half
* per CUM really pulled out of the forest
 - SUM((REGION,COUNTRY) $REGION_MAP(REGION,COUNTRY),

   + SUM((ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)
 $(FOREST_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType) AND
   LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'MngFor')),
        HARVEST_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)
       *FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'Increment')
       *FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'HarvCost'))
       *HARVCOST_SHAREFIXED
   +
 (
    SUM((ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)
 $  (FOREST_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType) AND
    LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'MngFor')),
    (SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'SW_biomass')*2 +
     SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'PW_biomass') +
     SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'OW_biomass') +
     SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'FW_biomass') +
     SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'HarvLoss') +
     SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'BranchStump'))

     *FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'HarvCost'))
 )*(1-HARVCOST_SHAREFIXED)
)

 - SUM((REGION_MAP(DEFORLOG_REGION,COUNTRY),ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)
     $(LUCDET_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'PriFor','DfrLnd') AND
       LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'DfrLnd')),
         LANDAVAIL_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,"DfrLnd")
       * FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CurN','Deforest_Logs')
       *(FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CurN','HarvCost')+ 0.030))
      /PeriodLength  * DeforHarvEffic
```

Objective function

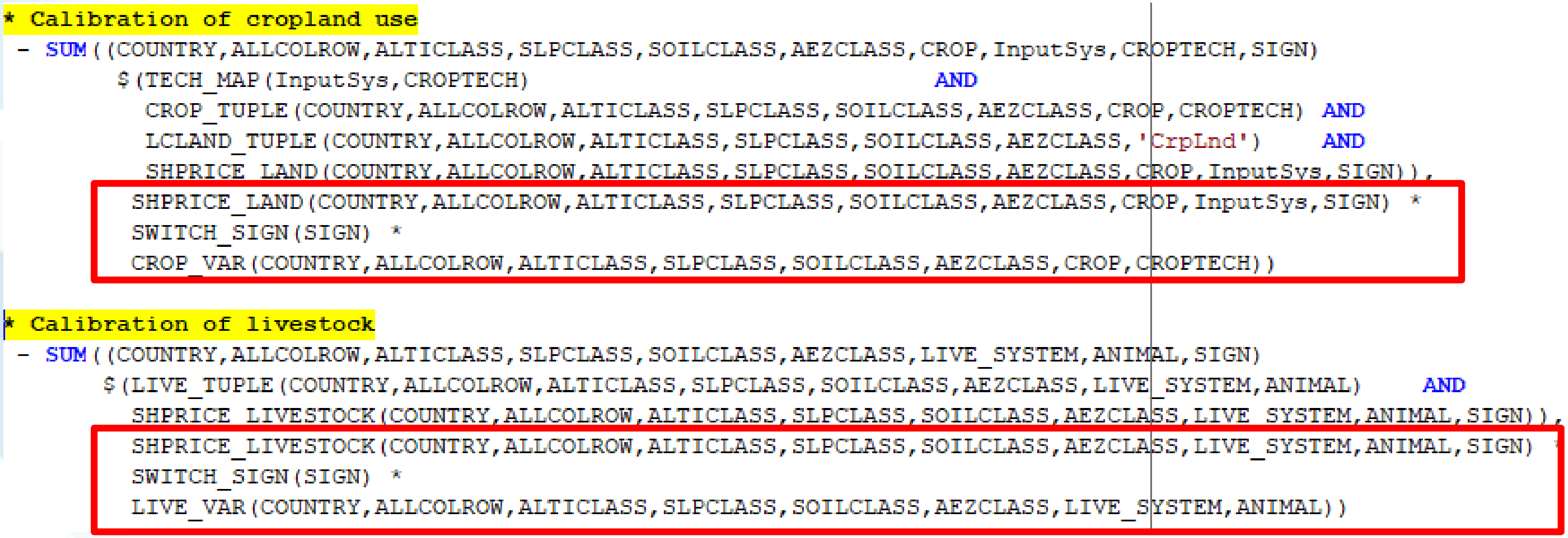
- Land use change cost



```
* Land use change with rising marginal costs
  - SUM((REGION,LC_INIT,LC_CRNT,STEP)
     $(LUCDET_DATA(REGION,LC_INIT,LC_CRNT,'MAXIMUM')                          AND
      SUM((REGION_MAP(REGION,COUNTRY),ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS),
         LUCDET_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT,LC_CRNT))  AND
          SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)
              $ REGION_MAP(REGION,COUNTRY),
                LC_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_INIT) AND
                LC_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LC_CRNT))),
    ( LUCDET_DATA(REGION,LC_INIT,LC_CRNT,"INTERCEPT")  *
      LUCDET_DATA(REGION,LC_INIT,LC_CRNT,"MAXIMUM")  *
      (ORD(STEP)/CARD(STEP))
    + 0.5 *
      LUCDET_DATA(REGION,LC_INIT,LC_CRNT,"SLOPE")  *
      ((LUCDET_DATA(REGION,LC_INIT,LC_CRNT,"MAXIMUM")  *
       (ORD(STEP)/CARD(STEP)))**2))  *
    LUCDET_STEP(REGION,LC_INIT,LC_CRNT,STEP))
```

Objective function

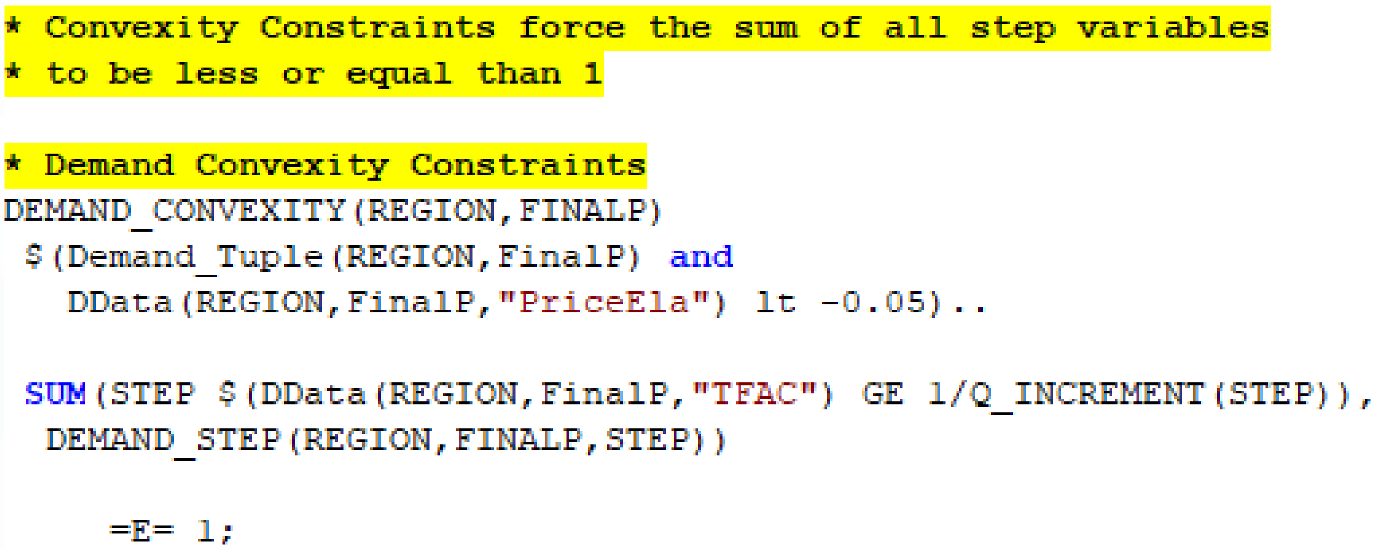
- Crop and livestock calibration cost



```
* Calibration of cropland use
 - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,InputSys,CROPTECH,SIGN)
        $(TECH_MAP(InputSys,CROPTECH)                              AND
          CROP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH)  AND
          LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CrpLnd')     AND
          SHPRICE_LAND(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,InputSys,SIGN)),
         SHPRICE_LAND(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,InputSys,SIGN) *
         SWITCH_SIGN(SIGN) *
         CROP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH))

* Calibration of livestock
 - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,SIGN)
       $(LIVE_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL)     AND
         SHPRICE_LIVESTOCK(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,SIGN)),
         SHPRICE_LIVESTOCK(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,SIGN) *
         SWITCH_SIGN(SIGN) *
         LIVE_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL))
```

Linearization

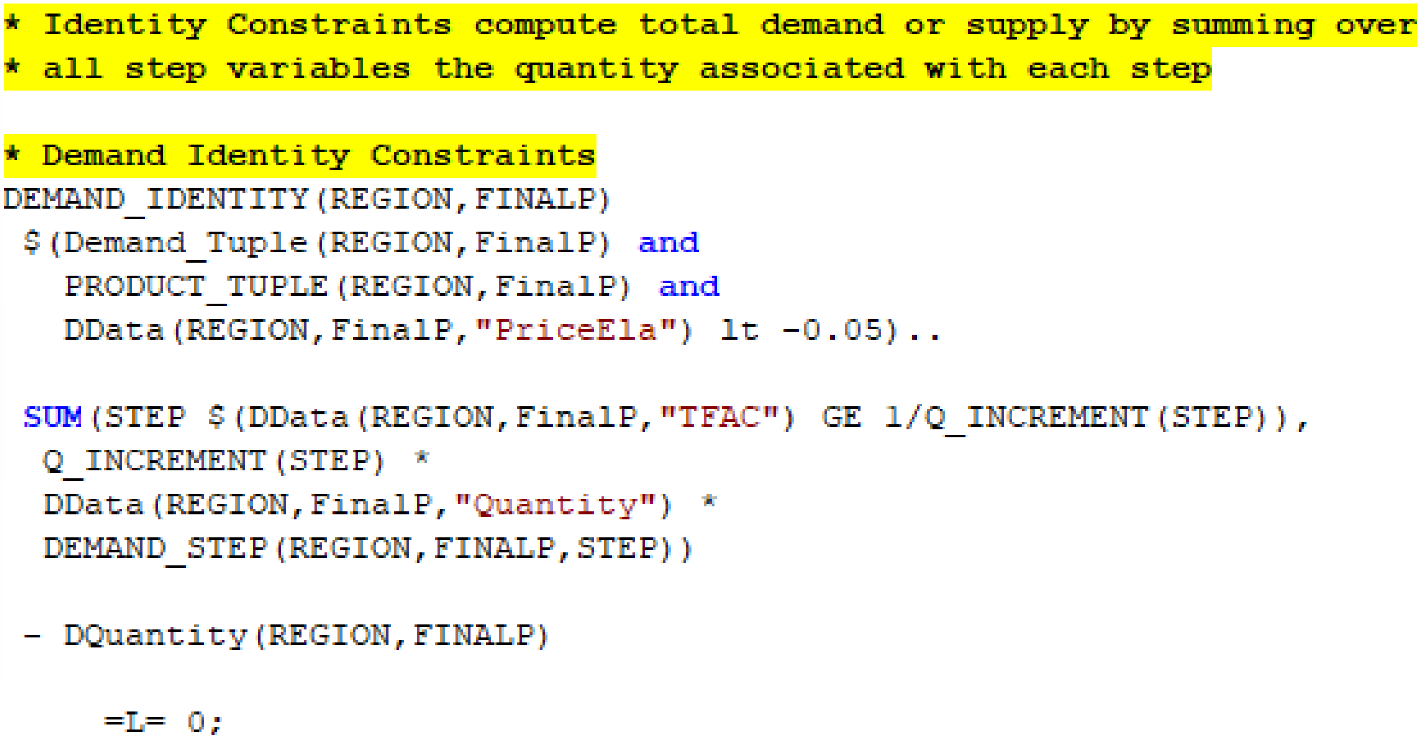
- Convexity constraints



```
* Convexity Constraints force the sum of all step variables
* to be less or equal than 1

* Demand Convexity Constraints
DEMAND_CONVEXITY(REGION,FINALP)
 $(Demand_Tuple(REGION,FinalP) and
   DData(REGION,FinalP,"PriceEla") lt -0.05)..

 SUM(STEP $(DData(REGION,FinalP,"TFAC") GE 1/Q_INCREMENT(STEP)),
  DEMAND_STEP(REGION,FINALP,STEP))

     =E= 1;
```

Linearization

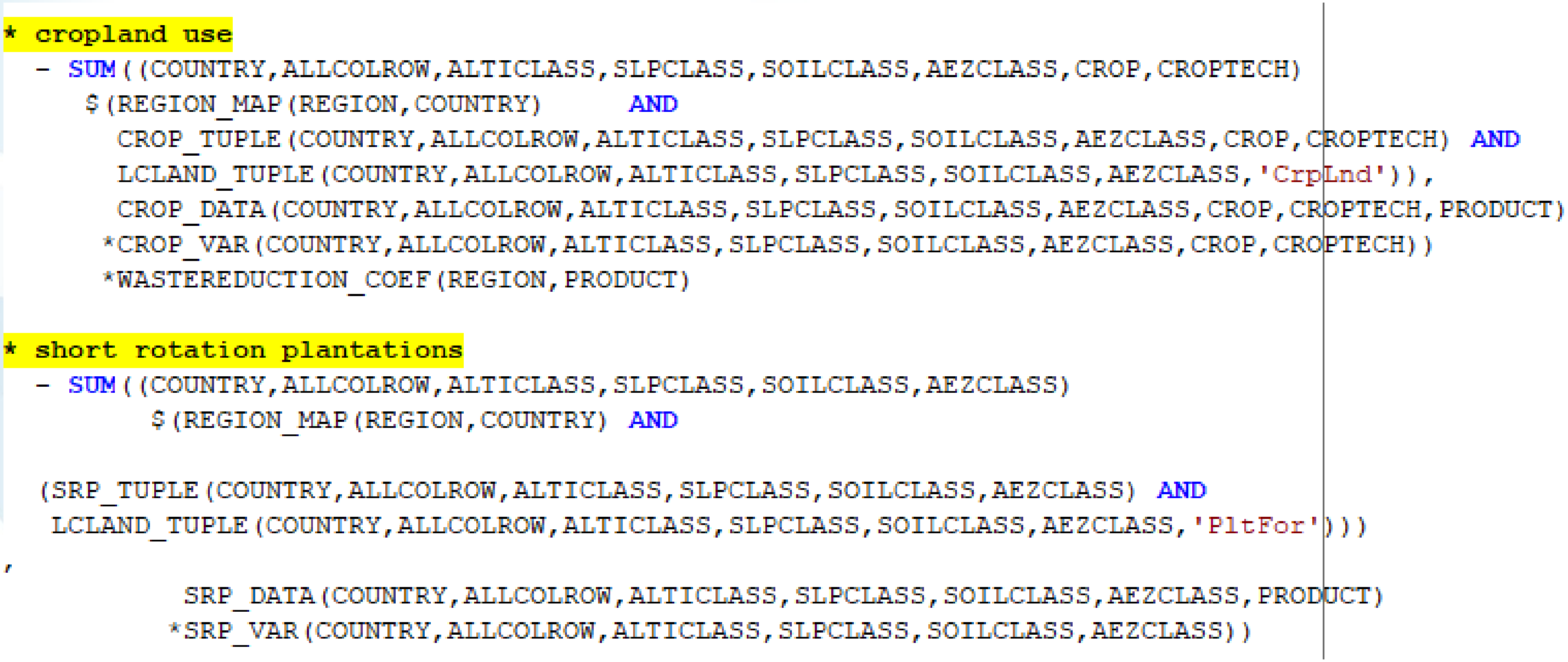
- Identity constraints



```
* Identity Constraints compute total demand or supply by summing over
* all step variables the quantity associated with each step

* Demand Identity Constraints
DEMAND_IDENTITY(REGION,FINALP)
 $(Demand_Tuple(REGION,FinalP) and
   PRODUCT_TUPLE(REGION,FinalP) and
   DData(REGION,FinalP,"PriceEla") lt -0.05)..

 SUM(STEP $(DData(REGION,FinalP,"TFAC") GE 1/Q_INCREMENT(STEP)),
  Q_INCREMENT(STEP) *
  DData(REGION,FinalP,"Quantity") *
  DEMAND_STEP(REGION,FINALP,STEP))

 - DQuantity(REGION,FINALP)

     =L= 0;
```

Market balance

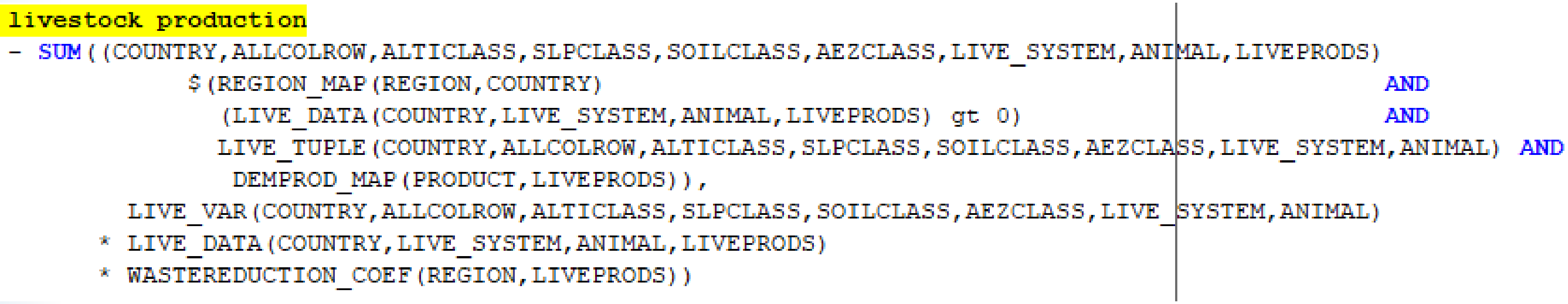
- Crop and plantation biomass supply



```
* cropland use
  - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH)
     $(REGION_MAP(REGION,COUNTRY)      AND
       CROP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH)  AND
       LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CrpLnd')),
       CROP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH,PRODUCT)
      *CROP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,CROPTECH))
      *WASTEREDUCTION_COEF(REGION,PRODUCT)

* short rotation plantations
  - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)
         $(REGION_MAP(REGION,COUNTRY)  AND

  (SRP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS)  AND
   LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'PltFor')))

,
           SRP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,PRODUCT)
          *SRP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS))
```

Market balance

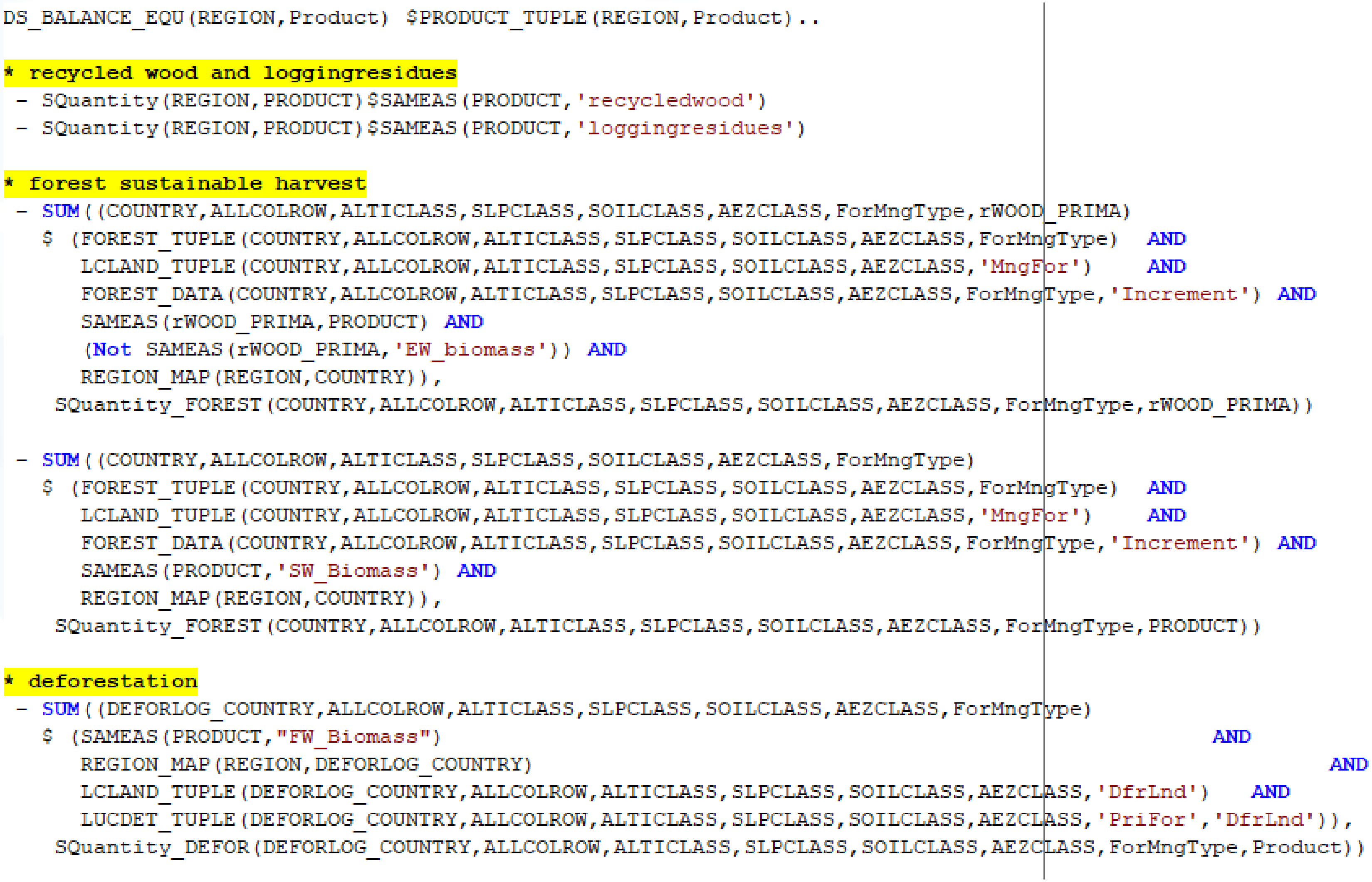
- Livestock production



```
livestock production
- SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,LIVEPRODS)
            $(REGION_MAP(REGION,COUNTRY)                                                  AND
              (LIVE_DATA(COUNTRY,LIVE_SYSTEM,ANIMAL,LIVEPRODS) gt 0)                      AND
              LIVE_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL) AND
               DEMPROD_MAP(PRODUCT,LIVEPRODS)),
        LIVE_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL)
      * LIVE_DATA(COUNTRY,LIVE_SYSTEM,ANIMAL,LIVEPRODS)
      * WASTEREDUCTION_COEF(REGION,LIVEPRODS))
```

Market balance

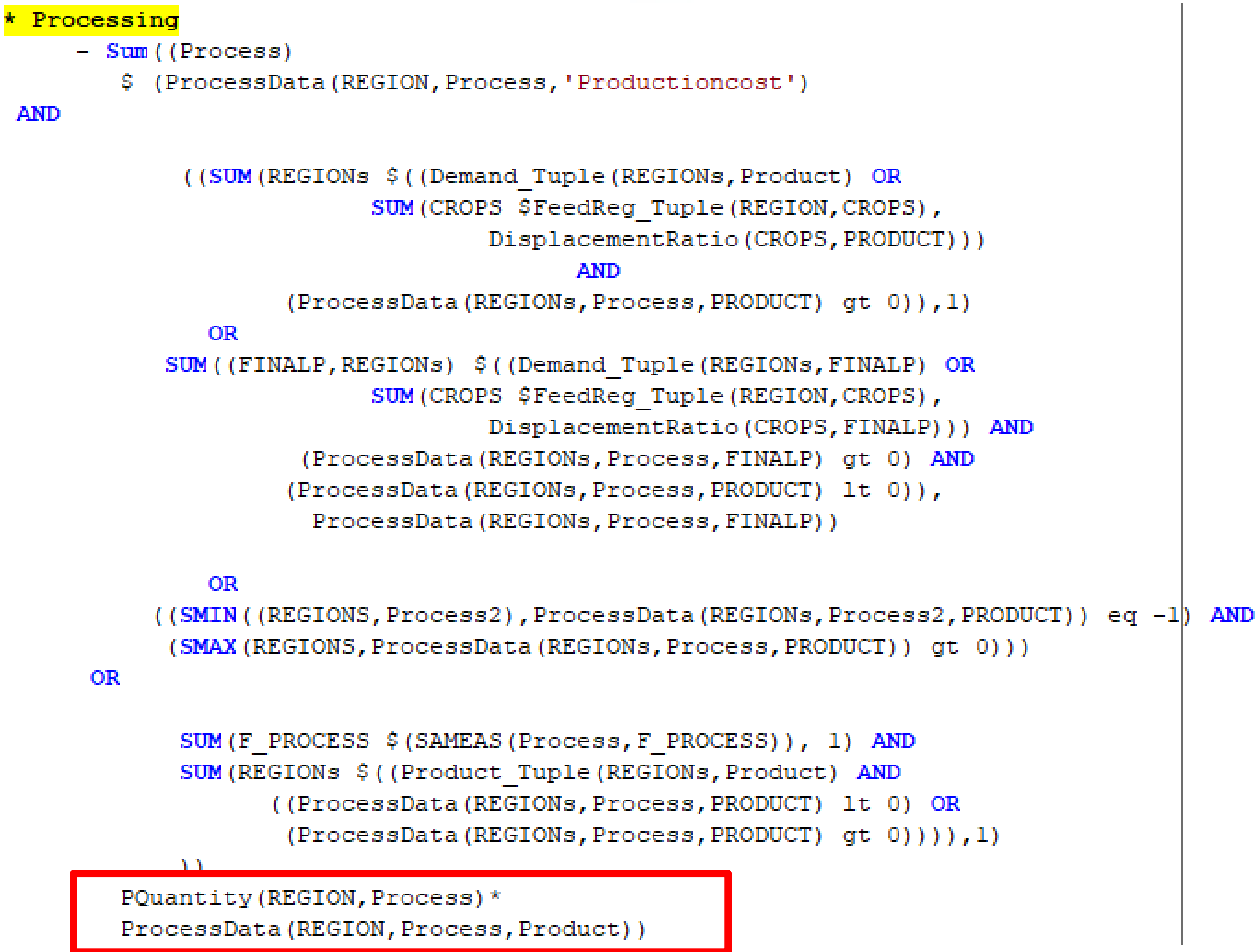
- Forest biomass supply



```
DS_BALANCE_EQU(REGION,Product) $PRODUCT_TUPLE(REGION,Product)..

* recycled wood and loggingresidues
 - SQuantity(REGION,PRODUCT)$SAMEAS(PRODUCT,'recycledwood')
 - SQuantity(REGION,PRODUCT)$SAMEAS(PRODUCT,'loggingresidues')

* forest sustainable harvest
 - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,rWOOD_PRIMA)
   $ (FOREST_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)  AND
      LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'MngFor')    AND
      FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'Increment') AND
      SAMEAS(rWOOD_PRIMA,PRODUCT) AND
      (Not SAMEAS(rWOOD_PRIMA,'EW_biomass')) AND
      REGION_MAP(REGION,COUNTRY)),
    SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,rWOOD_PRIMA))

 - SUM((COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)
   $ (FOREST_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)  AND
      LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'MngFor')    AND
      FOREST_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,'Increment') AND
      SAMEAS(PRODUCT,'SW_Biomass') AND
      REGION_MAP(REGION,COUNTRY)),
    SQuantity_FOREST(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,PRODUCT))

* deforestation
 - SUM((DEFORLOG_COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType)
   $ (SAMEAS(PRODUCT,"FW_Biomass")                                                        AND
      REGION_MAP(REGION,DEFORLOG_COUNTRY)                                                          AND
      LCLAND_TUPLE(DEFORLOG_COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'DfrLnd')   AND
      LUCDET_TUPLE(DEFORLOG_COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'PriFor','DfrLnd')),
    SQuantity_DEFOR(DEFORLOG_COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,ForMngType,Product))
```

Market balance

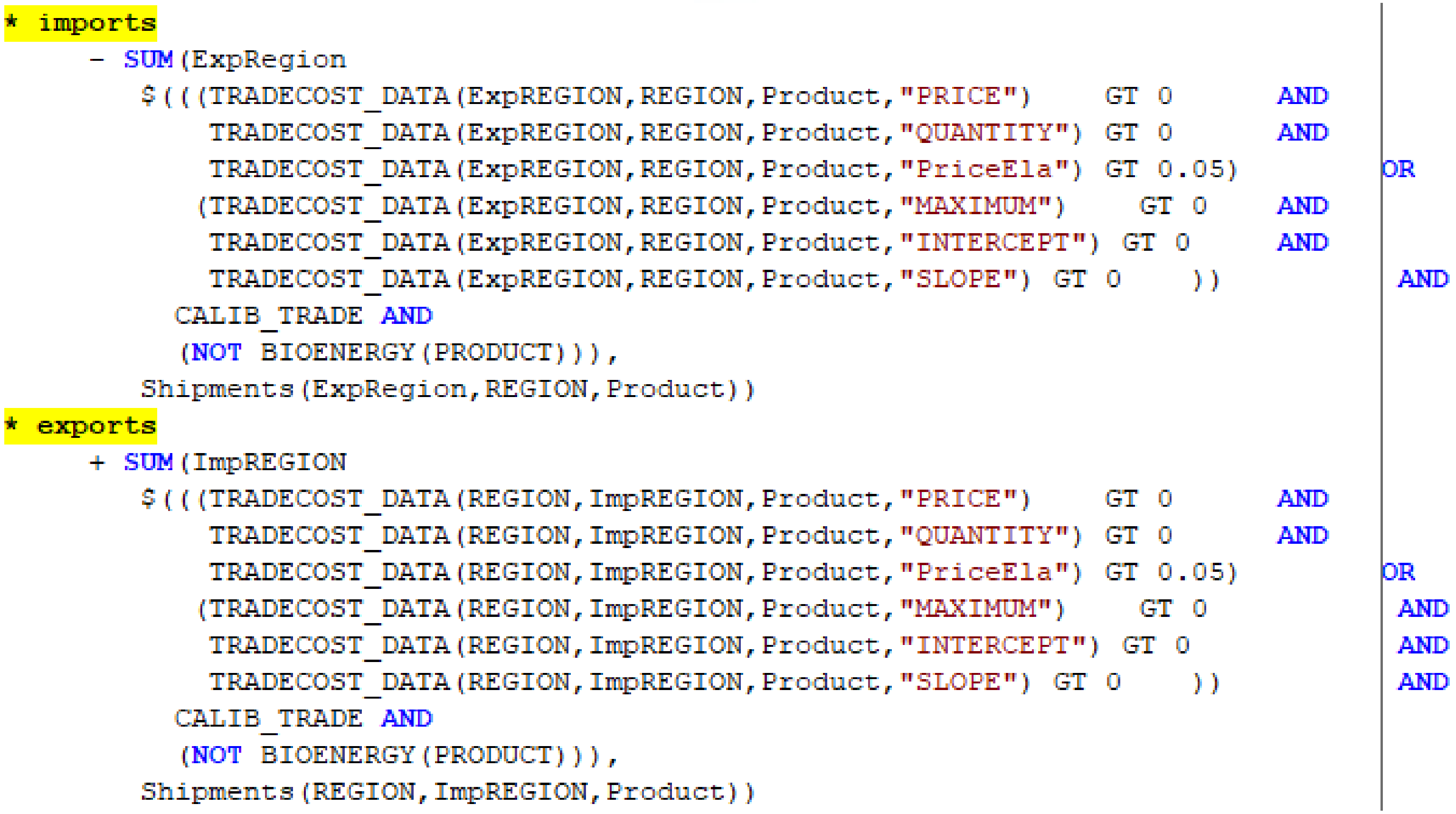
- Processing supply and demand



```
* Processing
     - Sum((Process)
        $ (ProcessData(REGION,Process,'Productioncost')
 AND

            ((SUM(REGIONs $((Demand_Tuple(REGIONs,Product) OR
                         SUM(CROPS $FeedReg_Tuple(REGION,CROPS),
                                 DisplacementRatio(CROPS,PRODUCT)))
                                       AND
                   (ProcessData(REGIONs,Process,PRODUCT) gt 0)),1)
              OR
           SUM((FINALP,REGIONs) $((Demand_Tuple(REGIONs,FINALP) OR
                         SUM(CROPS $FeedReg_Tuple(REGION,CROPS),
                                 DisplacementRatio(CROPS,FINALP))) AND
                    (ProcessData(REGIONs,Process,FINALP) gt 0) AND
                   (ProcessData(REGIONs,Process,PRODUCT) lt 0)),
                     ProcessData(REGIONs,Process,FINALP))

              OR
          ((SMIN((REGIONS,Process2),ProcessData(REGIONs,Process2,PRODUCT)) eq -1) AND
           (SMAX(REGIONS,ProcessData(REGIONs,Process,PRODUCT)) gt 0)))
      OR

            SUM(F_PROCESS $(SAMEAS(Process,F_PROCESS)), 1) AND
            SUM(REGIONs $((Product_Tuple(REGIONs,Product) AND
                  ((ProcessData(REGIONs,Process,PRODUCT) lt 0) OR
                   (ProcessData(REGIONs,Process,PRODUCT) gt 0)))),1)
            )),
        PQuantity(REGION,Process)*
        ProcessData(REGION,Process,Product))
```

Market balance

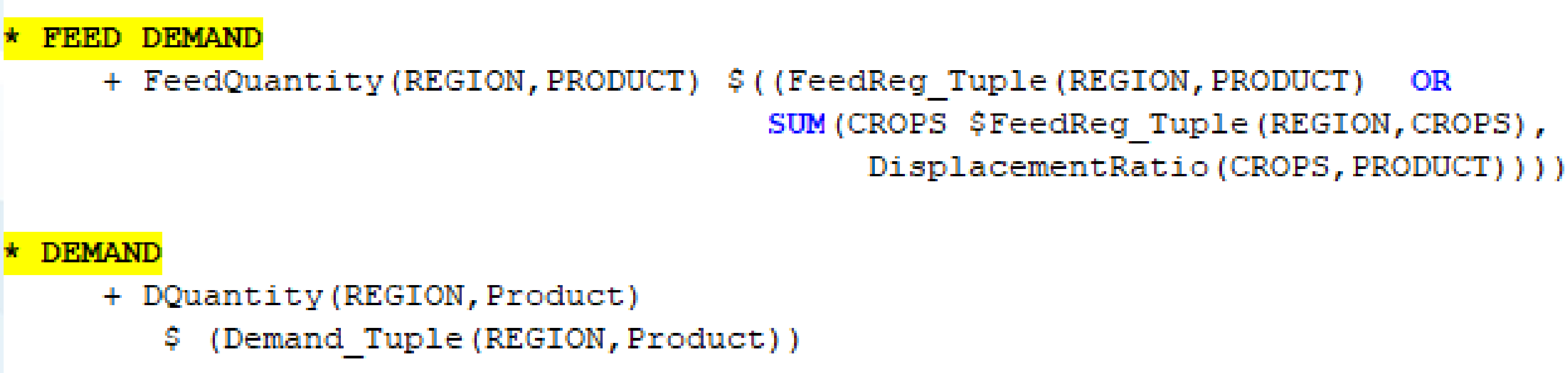
- Imports and Export



```
* imports
      - SUM(ExpRegion
         $(((TRADECOST_DATA(ExpREGION,REGION,Product,"PRICE")     GT 0       AND
             TRADECOST_DATA(ExpREGION,REGION,Product,"QUANTITY") GT 0       AND
             TRADECOST_DATA(ExpREGION,REGION,Product,"PriceEla") GT 0.05)         OR
            (TRADECOST_DATA(ExpREGION,REGION,Product,"MAXIMUM")    GT 0     AND
             TRADECOST_DATA(ExpREGION,REGION,Product,"INTERCEPT") GT 0      AND
             TRADECOST_DATA(ExpREGION,REGION,Product,"SLOPE") GT 0    ))            AND
           CALIB_TRADE AND
           (NOT BIOENERGY(PRODUCT))),
         Shipments(ExpRegion,REGION,Product))
* exports
      + SUM(ImpREGION
         $(((TRADECOST_DATA(REGION,ImpREGION,Product,"PRICE")     GT 0       AND
             TRADECOST_DATA(REGION,ImpREGION,Product,"QUANTITY") GT 0       AND
             TRADECOST_DATA(REGION,ImpREGION,Product,"PriceEla") GT 0.05)         OR
            (TRADECOST_DATA(REGION,ImpREGION,Product,"MAXIMUM")    GT 0           AND
             TRADECOST_DATA(REGION,ImpREGION,Product,"INTERCEPT") GT 0            AND
             TRADECOST_DATA(REGION,ImpREGION,Product,"SLOPE") GT 0    ))            AND
           CALIB_TRADE AND
           (NOT BIOENERGY(PRODUCT))),
         Shipments(REGION,ImpREGION,Product))
```

Market balance

- Feed and human demand



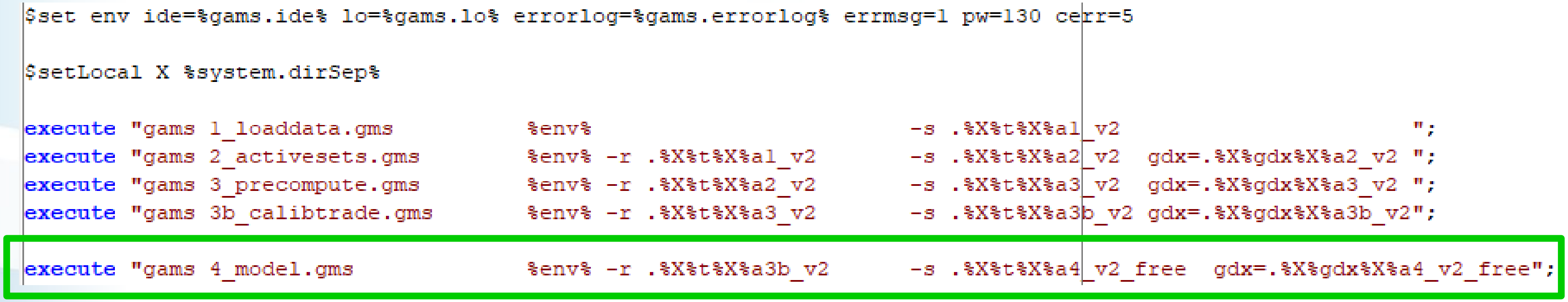
```
* FEED DEMAND
     + FeedQuantity(REGION,PRODUCT) $((FeedReg_Tuple(REGION,PRODUCT)   OR
                                      SUM(CROPS $FeedReg_Tuple(REGION,CROPS),
                                           DisplacementRatio(CROPS,PRODUCT))))

* DEMAND
     + DQuantity(REGION,Product)
        $ (Demand_Tuple(REGION,Product))
```

Market balance

- Artificial variables

Exercise 1: Run GLOBIOM!

- In **`0_executebatch.gms`** run **`4_model.gms`**



```
$set env ide=%gams.ide% lo=%gams.lo% errorlog=%gams.errorlog% errmsg=1 pw=130 cerr=5

$setLocal X %system.dirSep%

execute "gams 1_loaddata.gms          %env%                    -s .%X%t%X%a1_v2                      ";
execute "gams 2_activesets.gms        %env% -r .%X%t%X%a1_v2       -s .%X%t%X%a2_v2  gdx=.%X%gdx%X%a2_v2 ";
execute "gams 3_precompute.gms        %env% -r .%X%t%X%a2_v2       -s .%X%t%X%a3_v2  gdx=.%X%gdx%X%a3_v2 ";
execute "gams 3b_calibtrade.gms       %env% -r .%X%t%X%a3_v2       -s .%X%t%X%a3b_v2 gdx=.%X%gdx%X%a3b_v2";

execute "gams 4_model.gms             %env% -r .%X%t%X%a3b_v2      -s .%X%t%X%a4_v2_free  gdx=.%X%gdx%X%a4_v2_free";
```

- First, without calibration
- In **`4_model.gms`**

  - activate **`$setglobal without_calib`**

  - comment off **`$setglobal without_free`**

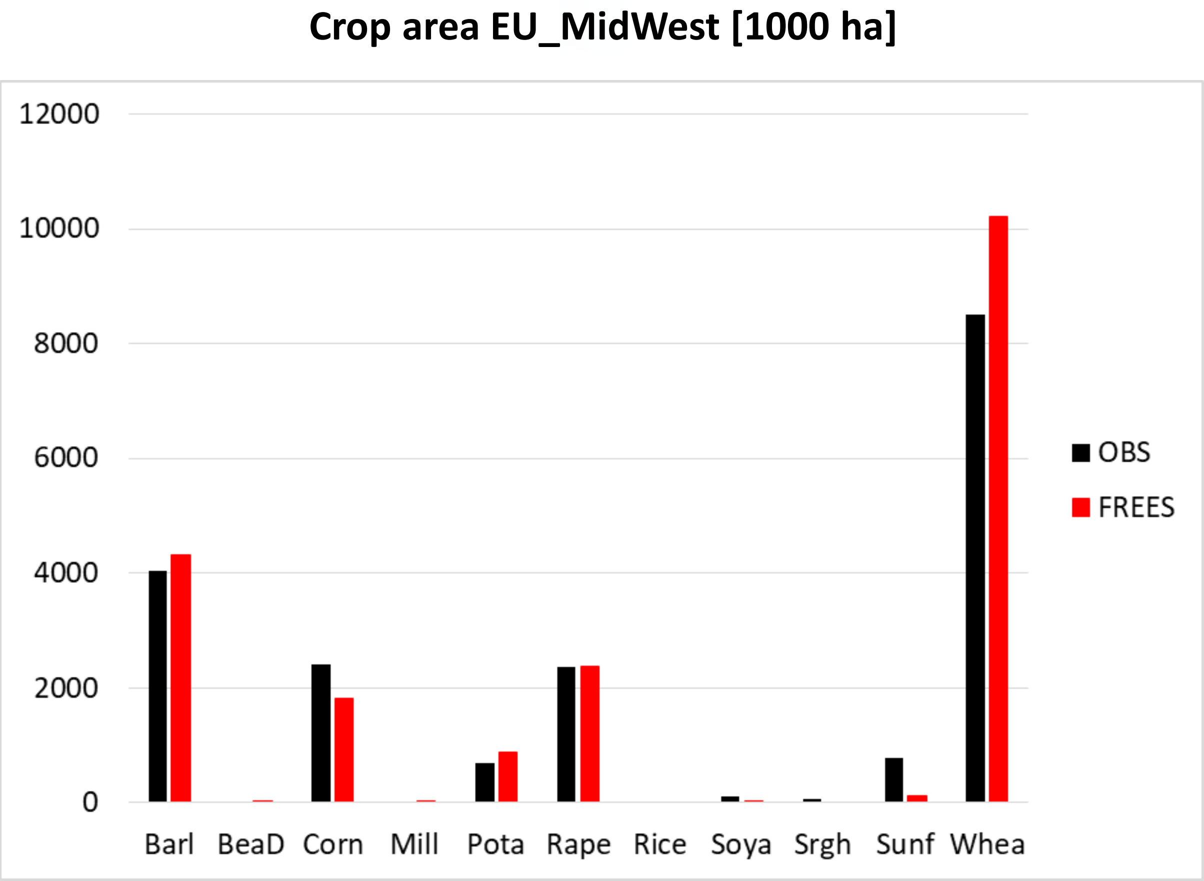
Exercise 1: Run GLOBIOM!

- Check the base year solution **FREES** compared to data **OBS**

- Parameters
  - **SOL_CROPLAND**
  - **SOL_CROPLAND_SYST**
  - **SOL_LIVESTOCK**
  - **SOL_LIVESTOCK_SYST**
  - **SOL_BIOMASS**
  - **SOL_PRICE**
  - **SOL_WELF**

Exercise 1: Run GLOBIOM!

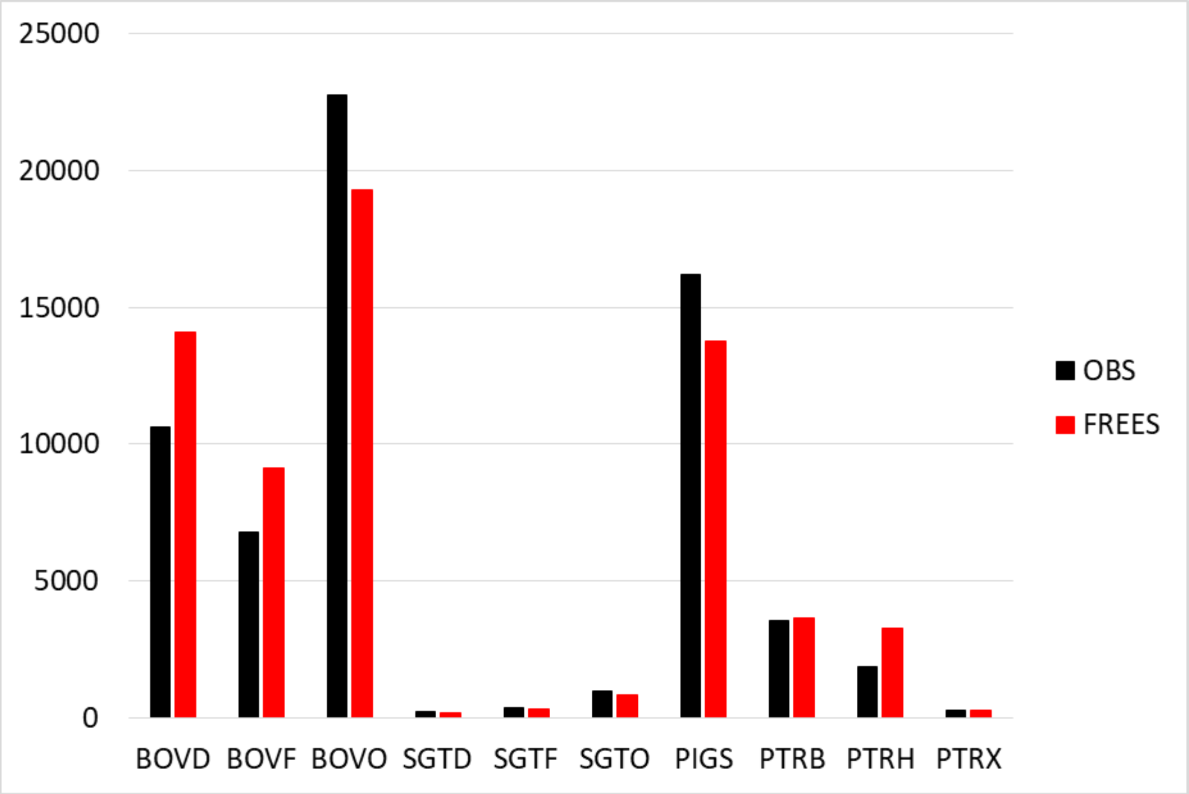
- Check crop areas (**SOL_CROPLAND**)



Exercise 1: Run GLOBIOM!

- Check livestock numbers (SOL_LIVESTOCK)

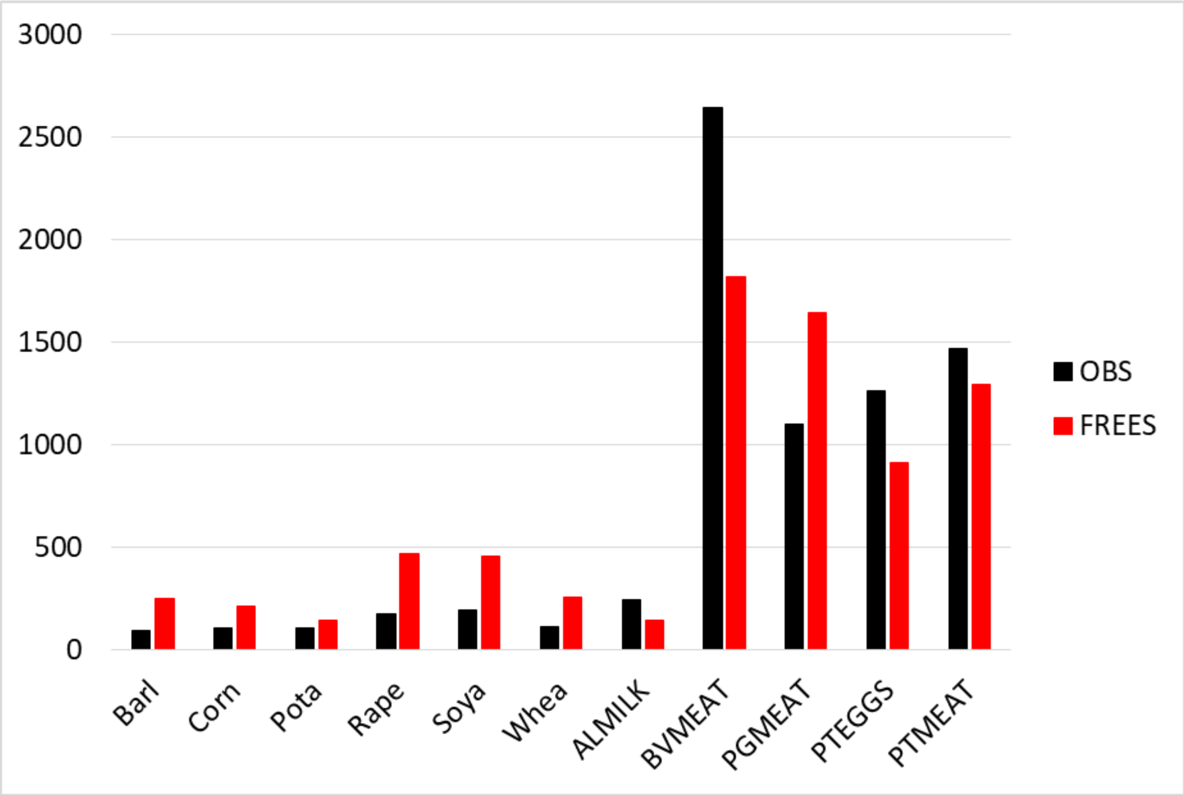
Livestock numbers EU_MidWest [1000 TLU]



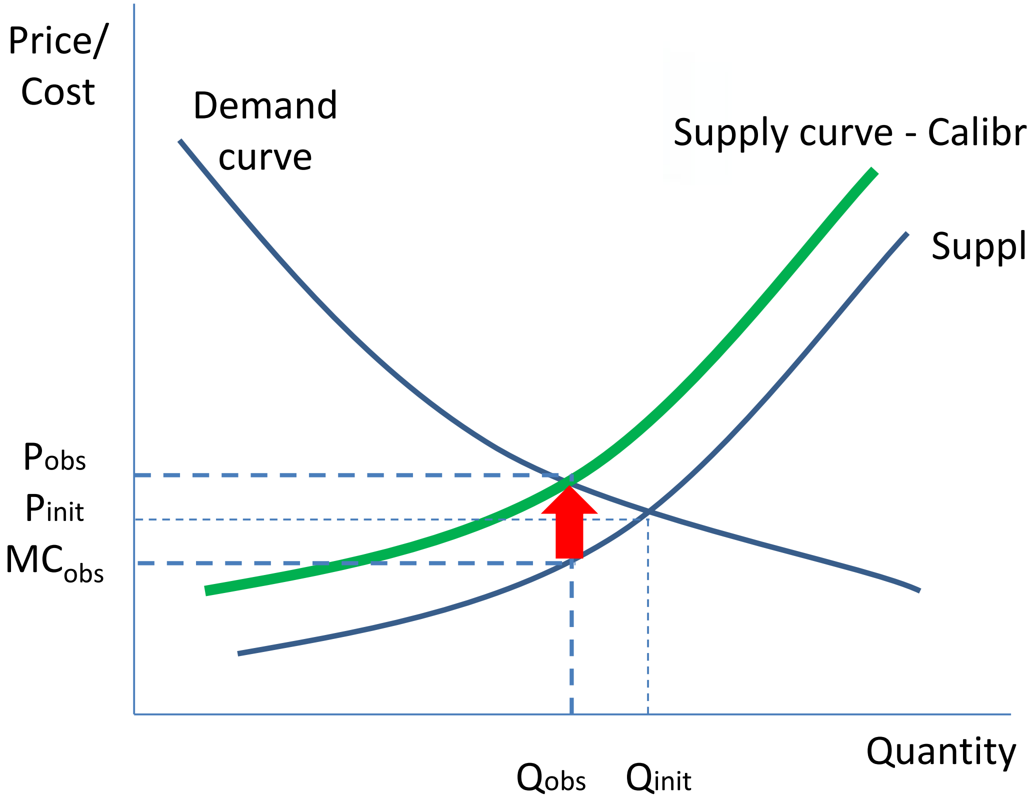
Exercise 1: Run GLOBIOM!

- Check commodity prices (**SOL_PRICE**)

Commodity prices EU_MidWest [USD/tonne]



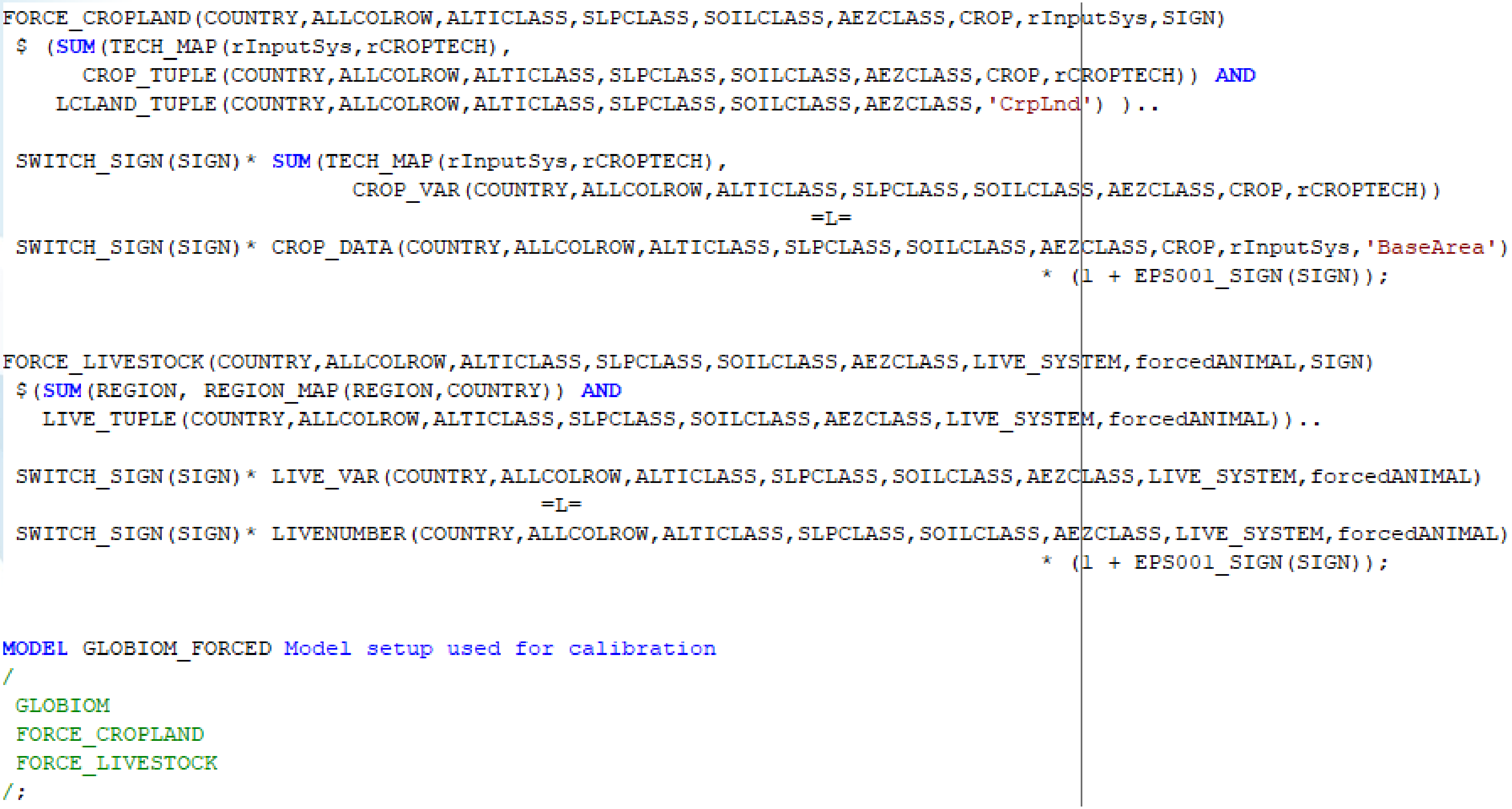
Calibration



MC=MR

Principle: Not all costs captured in data → add subsidies/costs to objective function to reproduce observed base year quantities

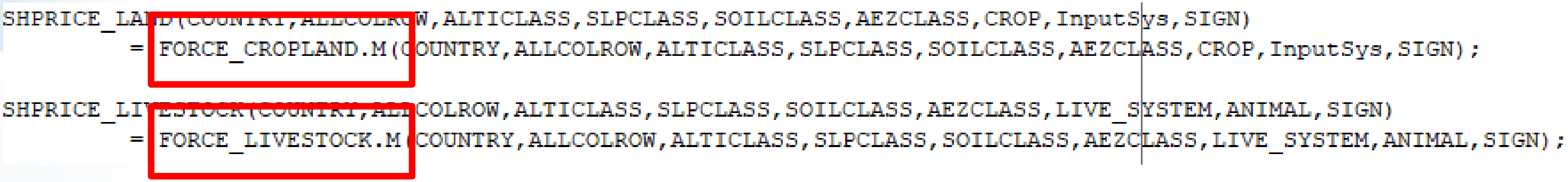
Calibration equations



```
FORCE_CROPLAND(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,rInputSys,SIGN)
 $  (SUM(TECH_MAP(rInputSys,rCROPTECH),
      CROP_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,rCROPTECH))  AND
    LCLAND_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,'CrpLnd')  )..

 SWITCH_SIGN(SIGN)*  SUM(TECH_MAP(rInputSys,rCROPTECH),
                          CROP_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,rCROPTECH))
                                                              =L=
 SWITCH_SIGN(SIGN)*  CROP_DATA(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,rInputSys,'BaseArea')
                                                                             *  (1  +  EPS001_SIGN(SIGN));


FORCE_LIVESTOCK(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,forcedANIMAL,SIGN)
 $(SUM(REGION,  REGION_MAP(REGION,COUNTRY))  AND
   LIVE_TUPLE(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,forcedANIMAL))..

 SWITCH_SIGN(SIGN)*  LIVE_VAR(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,forcedANIMAL)
                                          =L=
 SWITCH_SIGN(SIGN)*  LIVENUMBER(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,forcedANIMAL)
                                                                             *  (1  +  EPS001_SIGN(SIGN));


MODEL  GLOBIOM_FORCED  Model  setup  used  for  calibration
/
 GLOBIOM
 FORCE_CROPLAND
 FORCE_LIVESTOCK
/;
```

Calibration equations

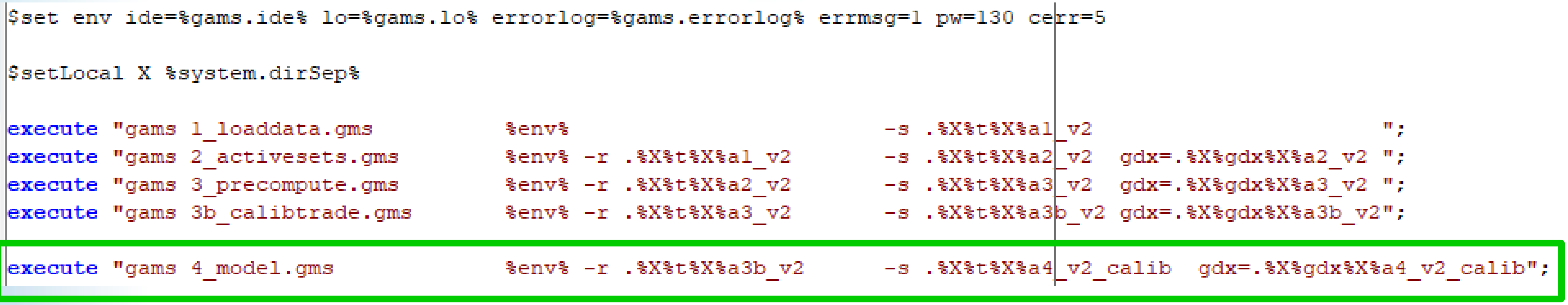
- Calibration cost



```
SHPRICE_LAND(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,InputSys,SIGN)
         = FORCE_CROPLAND.M(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,CROP,InputSys,SIGN);

SHPRICE_LIVESTOCK(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,SIGN)
         = FORCE_LIVESTOCK.M(COUNTRY,ALLCOLROW,ALTICLASS,SLPCLASS,SOILCLASS,AEZCLASS,LIVE_SYSTEM,ANIMAL,SIGN);
```

Exercise 2: CALIBRATE and run GLOBIOM!

- In **0_executebatch.gms** run **4_model.gms**

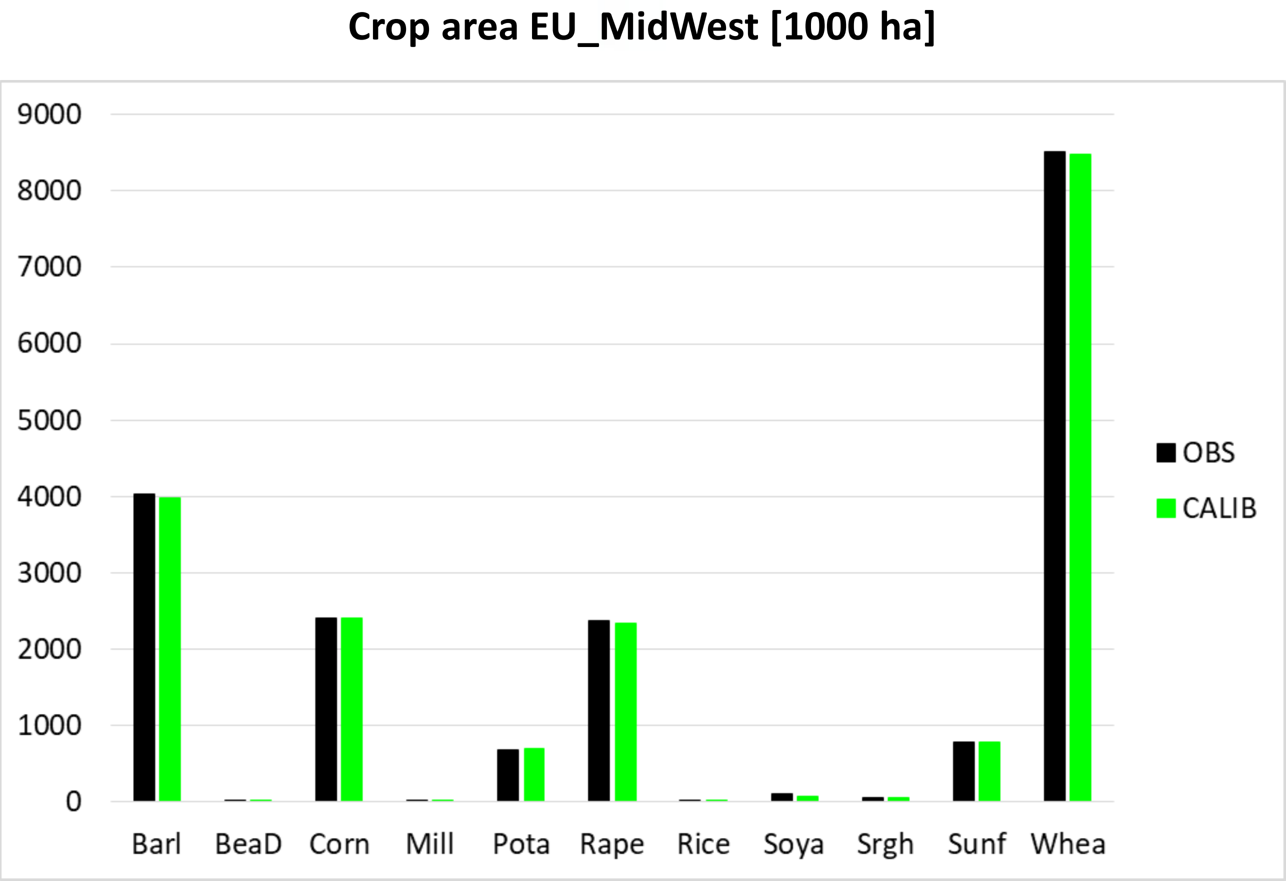


```
$set env ide=%gams.ide% lo=%gams.lo% errorlog=%gams.errorlog% errmsg=1 pw=130 cerr=5

$setLocal X %system.dirSep%

execute "gams 1_loaddata.gms          %env%                        -s .%X%t%X%a1_v2                      ";
execute "gams 2_activesets.gms        %env% -r .%X%t%X%a1_v2       -s .%X%t%X%a2_v2  gdx=.%X%gdx%X%a2_v2 ";
execute "gams 3_precompute.gms        %env% -r .%X%t%X%a2_v2       -s .%X%t%X%a3_v2  gdx=.%X%gdx%X%a3_v2 ";
execute "gams 3b_calibtrade.gms       %env% -r .%X%t%X%a3_v2       -s .%X%t%X%a3b_v2 gdx=.%X%gdx%X%a3b_v2";

execute "gams 4_model.gms             %env% -r .%X%t%X%a3b_v2      -s .%X%t%X%a4_v2_calib  gdx=.%X%gdx%X%a4_v2_calib";
```

- Now, with calibration
- In **4_model.gms**
  - comment off **$setglobal without_calib**
  - activate **$setglobal without_free**
- Check the base year solution **CALIB** compared to data **OBS**

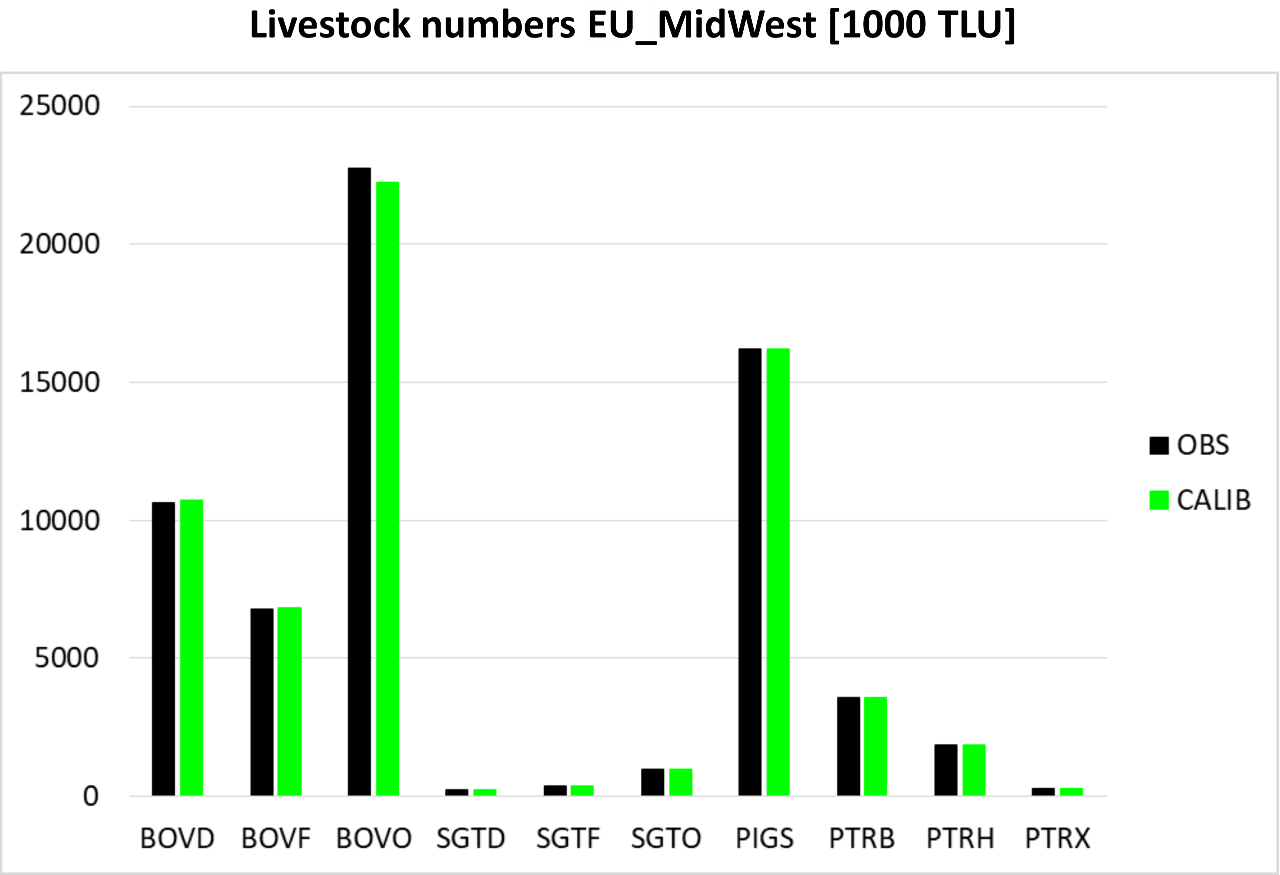
Exercise 2: CALIBRATE and run GLOBIOM!

- Check crop areas (**SOL_CROPLAND**)



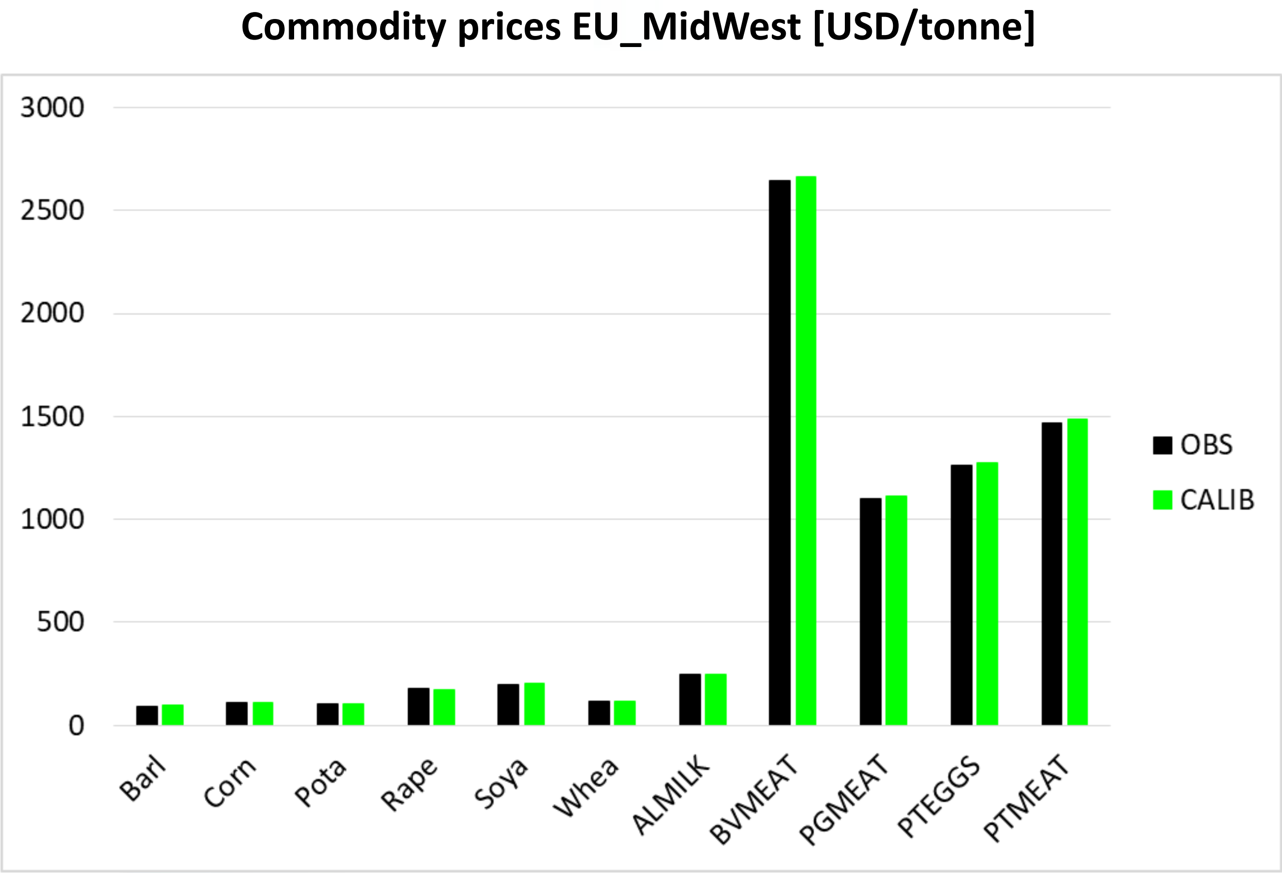
Exercise 2: CALIBRATE and run GLOBIOM!

- Check livestock numbers (`SOL_LIVESTOCK`)



Exercise 2: CALIBRATE and run GLOBIOM!

- Check commodity prices (SOL_PRICE)


Commodity prices EU_MidWest [USD/tonne]
Barl
Corn
Pota
Rape
Soya
Whea
ALMILK
BVMEAT
PGMEAT
PTEGGS
PTMEAT
OBS
CALIB