\*\*\*

\* .. \_parameter\_def:

\*

\* Parameter definition

\* ====================

\*

\* This file contains the definition of all parameters used in |MESSAGEix|.

\*

\* In |MESSAGEix|, all parameters are understood as yearly values, not as per (multi-year) period.

\* This provides flexibility when changing the resolution of the model horizon (i.e., the set ``year``).

\*\*\*

\*\*\*

\* .. \_section\_parameter\_general:

\* .. \_duration\_period:

\* .. \_duration\_time\_rel:

\*

\* General parameters of the |MESSAGEix| implementation

\* ----------------------------------------------------

\*

\* .. caution::

\*

\* Parameters written in \*\*bold\*\* are auxiliary parameters

\* that are either generated automatically when exporting a :class:`message\_ix.Scenario` to gdx

\* or that are computed during the \*pre-processing\* stage in GAMS (see the footnotes for more

\* individual details). These are \*\*not\*\* meant to be edited through the API when editing scenarios.

\*

\* .. list-table::

\* :widths: 25 20 55

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* - Explanatory comments

\* \* - interestrate

\* - ``year``

\* - Economy-wide interest rate or social discount rate

\* \* - duration\_time

\* - ``time``

\* - Duration of sub-annual time slices (relative to 1) [#duration\_time\_year]\_

\* \* - \*\*duration\_period\*\* (:math:`|y|`) [#short\_dur]\_

\* - ``year``

\* - Duration of multi-year period (in number of years) [#year\_auto]\_

\* \* - \*\*duration\_period\_sum\*\*

\* - ``year`` | ``year``

\* - Number of years between two periods [#df\_auto]\_

\* \* - \*\*duration\_time\_rel\*\*

\* - ``time`` | ``time``

\* - Relative duration between sub-annual time slices [#df\_auto]\_

\* \* - \*\*df\_period\*\*

\* - ``year``

\* - Cumulative discount factor over period duration [#df\_auto]\_

\* \* - \*\*df\_year\*\*

\* - ``year``

\* - Discount factor of the last year in the period [#df\_auto]\_

\*

\* .. [#duration\_time\_year] The element 'Year' in the set of subannual time slices ``time`` has the value of 1.

\* This value is assigned by default when creating a new :class:`ixmp.Scenario` based on the ``MESSAGE`` scheme.

\*

\* .. [#short\_dur] The short-hand notation :math:`|y|` is used for the parameters :math:`\text{duration\_period}\_y`

\* in the mathematical model documentation for exponents.

\*

\* .. [#year\_auto] The values for this parameter are computed automatically when exporting a ``MESSAGE``-scheme

\* :class:`ixmp.Scenario` to gdx.

\* Note that in |MESSAGEix|, the elements of the ``year`` set are understood to be the last year in a period.

\* See :doc:`/time`.

\*

\* .. [#df\_auto] These parameters are computed during the GAMS execution.

\*

\* .. \_duration\_period\_sum:

\*

\* duration\_period\_sum (dimensions :math:`y\_a, y\_b`)

\* This parameter measures the total time from the \*\*start\*\* of period :math:`y\_a`

\* until the \*\*start\*\* of any following period :math:`y\_b`

\* (equivalently: until the \*\*end\*\* of the period immediately preceding :math:`y\_b`).

\*

\* For example, with periods labelled '1000', '1010', '1015', and '1020':

\*

\* - The period '1000' ends on and includes the day 1000-12-31.

\* - The period '1010' starts on 1001-01-01 and ends on 1010-12-31.

\* - The period '1020' starts on 1016-01-01.

\* - Thus ``duration\_period\_sum(1010, 1020)`` measures the total time from 1001-01-01 to 1016-01-01,

\* which is 15 years.

\* - This is the same as ``duration\_period(1010) + duration\_period(1015)``.

\*

\* This parameter is used,

\* \*inter alia\*,

\* to populate |map\_tec\_lifetime| and compute |remaining\_capacity|.

\*\*\*

Parameters

\* general parameters

interestrate(year\_all) interest rate (to compute discount factor)

duration\_time(time) duration of one time slice (relative to 1)

duration\_period(year\_all) duration of one multi-year period (in years)

duration\_period\_sum(year\_all,year\_all2) number of years between two periods ('year\_all' must precede 'year\_all2')

duration\_time\_rel(time,time2) relative duration of subannual time period ('time2' relative to parent 'time')

df\_period(year\_all) cumulative discount factor over period duration

df\_year(year\_all) discount factor of the last year in the period

;

\*\*\*

\* .. \_section\_parameter\_resources:

\*

\* Parameters of the `Resources` section

\* -------------------------------------

\*

\* In |MESSAGEix|, the volume of resources at the start of the model horizon is defined by ``resource\_volume``. The quantity of the

\* resources that are extracted per year is dependent on two parameters. The first is ``bound\_extraction\_up``, which constraints

\* the maximum extraction of the resources (by grade) in a year. The second is ``resource\_remaining``, which is the maximum

\* extraction of the remaining resources in a certain year, as a percentage. Extraction costs for resources are represented by

\* ``resource\_cost`` parameter.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - resource\_volume

\* - ``node`` | ``commodity`` | ``grade``

\* \* - resource\_cost

\* - ``node`` | ``commodity`` | ``grade`` | ``year``

\* \* - resource\_remaining

\* - ``node`` | ``commodity`` | ``grade`` | ``year``

\* \* - bound\_extraction\_up

\* - ``node`` | ``commodity`` | ``level`` | ``year``

\* \* - commodity\_stock [#stock]\_

\* - ``node`` | ``commodity`` | ``level`` | ``year``

\* \* - historical\_extraction [#hist]\_

\* - ``node`` | ``commodity`` | ``grade`` | ``year``

\*

\* .. [#stock] Commodity stock refers to an exogenous (initial) quantity of commodity in stock. This parameter allows

\* (exogenous) additions to the commodity stock over the model horizon, e.g., precipitation that replenishes the water table.

\*

\* .. [#hist] Historical values of new capacity and activity can be used for parametrising the vintage structure

\* of existing capacity and implement dynamic constraints in the first model period.

\*

\*\*\*

Parameter

\* resource and commodity parameters

resource\_volume(node,commodity,grade) volume of resources in-situ at start of the model horizon

resource\_cost(node,commodity,grade,year\_all) extraction costs for resource

resource\_remaining(node,commodity,grade,year\_all) maximum extraction relative to remaining resources (by year)

bound\_extraction\_up(node,commodity,grade,year\_all) upper bound on extraction of resources by grade

commodity\_stock(node,commodity,level,year\_all) exogenous (initial) quantity of commodity in stock

historical\_extraction(node,commodity,grade,year\_all) historical extraction

;

\*\*\*

\* .. \_section\_parameter\_demand:

\*

\* Parameters of the `Demand` section

\* ----------------------------------

\*

\* .. list-table::

\* :widths: 30 70

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - demand [demand\_fixed] [#demand]\_

\* - ``node`` | ``commodity`` | ``level`` | ``year`` | ``time``

\* \* - peak\_load\_factor [#peakload]\_

\* - ``node`` | ``commodity`` | ``year``

\*

\* .. [#demand] The parameter ``demand`` in a ``MESSAGE``-scheme ``ixmp.Scenario`` is translated

\* to the parameter ``demand\_fixed`` in the |MESSAGEix| implementation in GAMS. The variable ``DEMAND`` is introduced

\* as an auxiliary reporting variable; it equals ``demand\_fixed`` in a `MESSAGE`-standalone run and reports

\* the final demand including the price response in an iterative `MESSAGE-MACRO` solution.

\*

\* .. [#peakload] The parameters ``peak\_load\_factor`` (maximum peak load factor for reliability constraint of firm capacity) and

\* ``reliability\_factor`` (reliability of a technology (per rating)) are based on the formulation proposed by Sullivan et al., 2013 :cite:`Sullivan-2013`.

\* It is used in :ref:`reliability\_constraint`.

\*

\*\*\*

Parameter

demand\_fixed(node,commodity,level,year\_all,time) exogenous demand levels

peak\_load\_factor(node,commodity,level,year\_all,time) maximum peak load factor for reliability constraint of firm capacity

;

\*\*\*

\* .. \_params-tech:

\*

\* Parameters of the `Technology` section

\* --------------------------------------

\*

\* Input/output mapping, costs and engineering specifications

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* .. \_input:

\* .. \_output:

\* .. \_technical\_lifetime:

\*

\* .. list-table::

\* :widths: 25 60

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - input [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` |

\* ``node\_origin`` | ``commodity`` | ``level`` | ``time`` | ``time\_origin``

\* \* - output [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` |

\* ``node\_dest`` | ``commodity`` | ``level`` | ``time`` | ``time\_dest``

\* \* - inv\_cost [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - fix\_cost [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act``

\* \* - var\_cost [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` | ``time``

\* \* - levelized\_cost [#levelizedcost]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``time``

\* \* - construction\_time [#construction]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - technical\_lifetime

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - capacity\_factor [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``time``

\* \* - operation\_factor [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act``

\* \* - min\_utilization\_factor [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act``

\* \* - rating\_bin [#rating]\_

\* - ``node`` | ``tec`` | ``year\_act`` | ``commodity`` | ``level`` | ``time`` | ``rating``

\* \* - reliability\_factor [#peakload]\_

\* - ``node`` | ``tec`` | ``year\_act`` | ``commodity`` | ``level`` | ``time`` | ``rating``

\* \* - flexibility\_factor [#flexfactor]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` | ``commodity`` | ``level`` | ``time`` | ``rating``

\* \* - renewable\_capacity\_factor [#renewables]\_

\* - ``node\_loc`` | ``commodity`` | ``grade`` | ``level`` | ``year``

\* \* - renewable\_potential [#renewables]\_

\* - ``node`` | ``commodity`` | ``grade`` | ``level`` | ``year``

\* \* - emission\_factor

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` | ``emission``

\*

\* .. \_input\_cap:

\* .. \_input\_cap\_new:

\* .. \_input\_cap\_ret:

\* .. \_output\_cap:

\* .. \_output\_cap\_new:

\* .. \_output\_cap\_ret:

\*

\* Input/output mapping related to technology capacities

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* .. list-table::

\* :widths: 25 60 55

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* - Explanatory comments

\* \* - input\_cap [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` |

\* ``node\_origin`` | ``commodity`` | ``level`` | ``time\_origin``

\* - Commodity input for operation of 1 unit of CAP.

\* \* - output\_cap [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` | ``year\_act`` |

\* ``node\_dest`` | ``commodity`` | ``level`` | ``time\_dest``

\* - Commodity output due to operation of 1 unit of CAP.

\* \* - input\_cap\_new [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` |

\* ``node\_origin`` | ``commodity`` | ``level`` | ``time\_origin``

\* - Commodity input per unit of |CAP\_NEW|.

\* \* - output\_cap\_new [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` |

\* ``node\_dest`` | ``commodity`` | ``level`` | ``time\_dest``

\* - Commodity output per unit of |CAP\_NEW|.

\* \* - input\_cap\_ret [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` |

\* ``node\_origin`` | ``commodity`` | ``level`` | ``time\_origin``

\* - Commodity input per unit of |CAP| retired.

\* \* - output\_cap\_ret [#tecvintage]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg`` |

\* ``node\_dest`` | ``commodity`` | ``level`` | ``time\_dest``

\* - Commodity output per unit of |CAP| retired.

\*

\* .. [#tecvintage] Fixed and variable cost parameters and technical specifications are indexed over both

\* the year of construction (vintage) and the year of operation (actual).

\* This allows to represent changing technology characteristics depending on the age of the plant.

\* Material flows can also vary based on the vintage and active years.

\*

\* .. [#levelizedcost] The parameter ``levelized\_cost`` is computed in the GAMS pre-processing under the assumption of

\* full capacity utilization until the end of the technical lifetime.

\* As these are calculated in the preprocessing, the reported ``levelized\_cost`` in the output GDX-file exclude fuel costs.

\*

\* .. [#construction] The construction time only has an effect on the investment costs; in |MESSAGEix|,

\* each unit of new-built capacity is available instantaneously at the beginning of the model period.

\*

\* .. [#rating] Maximum share of technology in commodity use per rating. The upper bound of a contribution by any technology to the constraints on system reliability

\* (:ref:`reliability\_constraint`) and flexibility (:ref:`flexibility\_constraint`) can depend on the share of the technology output in the total commodity use at

\* a specific level.

\*

\* .. [#flexfactor] Contribution of technologies towards operation flexibility constraint. It is used in :ref:`flexibility\_constraint`.

\*

\* .. [#renewables] ``renewable\_capacity\_factor`` refers to the quality of renewable potential by grade and ``renewable\_potential`` refers to the size of the renewable potential per grade.

\*

\*\*\*

Parameters

\* technology input-output mapping and costs parameters

input(node,tec,vintage,year\_all,mode,node,commodity,level,time,time) relative share of input per unit of activity

output(node,tec,vintage,year\_all,mode,node,commodity,level,time,time) relative share of output per unit of activity

# Commodity input and output associated with operation of capacity

input\_cap(node,tec,vintage,year\_all,node,commodity,level,time) 'Commodity input for operation of 1 unit of CAP'

output\_cap(node,tec,vintage,year\_all,node,commodity,level,time) 'Commodity output due to operation of 1 unit of CAP'

# Commodity input and output associated with construction of capacity

input\_cap\_new(node,tec,vintage,node,commodity,level,time) 'Commodity input per unit of CAP\_NEW'

output\_cap\_new(node,tec,vintage,node,commodity,level,time) 'Commodity output per unit of CAP\_NEW'

# Commodity input and output associated with retirement of capacity

input\_cap\_ret(node,tec,vintage,node,commodity,level,time) 'Commodity input per unit of retired CAP'

output\_cap\_ret(node,tec,vintage,node,commodity,level,time) 'Commodity output per unit of retired CAP'

inv\_cost(node,tec,year\_all) investment costs (per unit of new capacity)

fix\_cost(node,tec,vintage,year\_all) fixed costs per year (per unit of capacity maintained)

var\_cost(node,tec,vintage,year\_all,mode,time) variable costs of operation (per unit of capacity maintained)

levelized\_cost(node,tec,year\_all,time) levelized costs (per unit of new capacity)

\* engineering parameters

construction\_time(node,tec,vintage) duration of construction of new capacity (in years)

technical\_lifetime(node,tec,vintage) maximum technical lifetime (from year of construction)

capacity\_factor(node,tec,vintage,year\_all,time) capacity factor by subannual time slice

operation\_factor(node,tec,vintage,year\_all) yearly total operation factor

min\_utilization\_factor(node,tec,vintage,year\_all) yearly minimum utilization factor

rating\_bin(node,tec,year\_all,commodity,level,time,rating) maximum share of technology in commodity use per rating

reliability\_factor(node,tec,year\_all,commodity,level,time,rating) reliability of a technology (per rating)

flexibility\_factor(node,tec,vintage,year\_all,mode,commodity,level,time,rating) contribution of technologies towards operation flexibility constraint

renewable\_capacity\_factor(node,commodity,grade,level,year\_all) quality of renewable potential grade (>= 1)

renewable\_potential(node,commodity,grade,level,year\_all) size of renewable potential per grade

emission\_factor(node,tec,year\_all,year\_all,mode,emission) emission intensity of activity

;

\*\*\*

\* .. \_section\_parameter\_bounds:

\*

\* Bounds on capacity and activity

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* The following parameters specify upper and lower bounds on new capacity, total installed capacity, and activity. The bounds

\* on activity are implemented as the aggregate over all vintages in a specific period (:ref:`activity\_bound\_up` and :ref:`activity\_bound\_lo`).

\*

\* .. list-table::

\* :widths: 25 60

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - bound\_new\_capacity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - bound\_new\_capacity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - bound\_total\_capacity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act``

\* \* - bound\_total\_capacity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act``

\* \* - bound\_activity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``mode`` | ``time``

\* \* - bound\_activity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``mode`` | ``time``

\*

\*\*\*

Parameters

bound\_new\_capacity\_up(node,tec,year\_all) upper bound on new capacity

bound\_new\_capacity\_lo(node,tec,year\_all) lower bound on new capacity

bound\_total\_capacity\_up(node,tec,year\_all) upper bound on total installed capacity

bound\_total\_capacity\_lo(node,tec,year\_all) lower bound on total installed capacity

bound\_activity\_up(node,tec,year\_all,mode,time) upper bound on activity (aggregated over all vintages)

bound\_activity\_lo(node,tec,year\_all,mode,time) lower bound on activity

;

\*\*\*

\* .. \_section\_parameter\_dynamic\_constraints:

\* .. \_growth\_new\_capacity\_up:

\* .. \_initial\_new\_capacity\_up:

\*

\* Dynamic constraints on new capacity and activity

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* These parameters are used in the :ref:`dynamic constraint equations <dynamic\_constraints>` to limit the growth (or decline) of activity or of new capacity in each period, relative to the preceding period.

\* The ``soft\_`` parameters control ‘soft’ relaxation of these dynamic constraints, using the method of Keppo and Strubegger (2010) :cite:`Keppo-2010`.

\*

\* The ``growth\_`` and ``soft\_`` parameters are expressed as \*relative annual change\* and are unitless.

\* Because these are annual values, are compounded in the :ref:`constraint equations <dynamic\_constraints>` by ``duration\_period`` (:math:`|y|`) to obtain the relative \*inter-period\* change.

\*

\* \*\*Example:\*\* a value of 0.05 for ``growth\_activity\_up`` sets an upper bound of :math:`1 + 0.05 = 105\%` activity in one year relative to activity in the preceding year.

\* In a period with duration :math:`|y| = 5 \text{ years}`, the activity in the :doc:`representative year </time>` is bounded at :math:`(1.05)^5 = 128\%` of the activity in the representative year of the preceding period.

\*

\* Because these parameters do not have a ``mode`` (:math:`m`) dimension, they cannot be used to constraint the activity/new capacity of \*single modes\* of technologies; only the total across all modes.

\*

\* .. list-table::

\* :widths: 30 70

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - initial\_new\_capacity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - growth\_new\_capacity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - soft\_new\_capacity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - initial\_new\_capacity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - growth\_new\_capacity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - soft\_new\_capacity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - initial\_activity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - growth\_activity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - soft\_activity\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - initial\_activity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - growth\_activity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - soft\_activity\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\*

\*\*\*

Parameters

initial\_new\_capacity\_up(node,tec,year\_all) dynamic upper bound on new capacity (fixed initial term)

growth\_new\_capacity\_up(node,tec,year\_all) dynamic upper bound on new capacity (growth rate)

soft\_new\_capacity\_up(node,tec,year\_all) soft relaxation of dynamic upper bound on new capacity (growth rate)

initial\_new\_capacity\_lo(node,tec,year\_all) dynamic lower bound on new capacity (fixed initial term)

growth\_new\_capacity\_lo(node,tec,year\_all) dynamic lower bound on new capacity (growth rate)

soft\_new\_capacity\_lo(node,tec,year\_all) soft relaxation of dynamic lower bound on new capacity (growth rate)

initial\_activity\_up(node,tec,year\_all,time) dynamic upper bound on activity (fixed initial term)

growth\_activity\_up(node,tec,year\_all,time) dynamic upper bound on activity (growth rate)

soft\_activity\_up(node,tec,year\_all,time) soft relaxation of dynamic upper bound on activity (growth rate)

initial\_activity\_lo(node,tec,year\_all,time) dynamic lower bound on activity (fixed initial term)

growth\_activity\_lo(node,tec,year\_all,time) dynamic lower bound on activity (growth rate)

soft\_activity\_lo(node,tec,year\_all,time) soft relaxation of dynamic lower bound on activity (growth rate),

# Auxiliaries for growth\_new\_capacity\_up

gncu\_1(node,tec,year\_all) Auxiliary for growth\_new\_capacity\_up,

gncu\_2(node,tec,year\_all) Auxiliary for growth\_new\_capacity\_up,

k\_gncu(node,tec,year\_all2,year\_all) Auxiliary for growth\_new\_capacity\_up

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Add-on technology parameters \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_addon:

\*

\* Parameters for the add-on technologies

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* The implementation of |MESSAGEix| includes the functionality to introduce "add-on technologies" that are specifically

\* linked to parent technologies. This feature can be used to model mitigation options (scrubber, cooling). Upper and

\* lower bounds of add-on technologies are defined relative to the parent: ``addon\_up`` and ``addon\_lo``, respectively.

\*

\* .. note::

\* No default ``addon\_conversion`` factor (conversion factor between add-on and parent technology activity) is set.

\* This is to avoid default conversion factors of 1 being set for technologies with multiple modes, of which only a

\* single mode should be linked to the add-on technology.

\*

\* .. list-table::

\* :widths: 20 80

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - addon\_conversion

\* - ``node`` | ``tec`` | ``year\_vtg`` | ``year\_act`` | ``mode`` | ``time`` | ``type\_addon``

\* \* - addon\_up

\* - ``node`` | ``tec`` | ``year\_act`` | ``mode`` | ``time`` | ``type\_addon``

\* \* - addon\_lo

\* - ``node`` | ``tec`` | ``year\_act`` | ``mode`` | ``time`` | ``type\_addon``

\*

\*\*\*

Parameters

addon\_conversion(node,tec,vintage,year\_all,mode,time,type\_addon) conversion factor between add-on and parent technology activity

addon\_up(node,tec,year\_all,mode,time,type\_addon) upper bound of add-on technologies relative to parent technology

addon\_lo(node,tec,year\_all,mode,time,type\_addon) lower bound of add-on technologies relative to parent technology

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Storage parameters

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_storage:

\*

\* Parameters for representing storage solutions

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* The |MESSAGEix| formulation includes "storage" solutions to model sub-annual, inter-temporal storage of commodities in each period.

\* This feature can be used to model electricity storage (pumped hydro, batteries, compressed air energy storage, etc.), thermal energy storage,

\* demand side management, and in general any technology for storing commodities (gas, hydrogen, water, etc.) over sub-annual timesteps.

\* The user defines the chronological order of sub-annual time slices by assigning a number to them in parameter ``time\_order``.

\* This order is used by storage equations to shift the stored commodity in a correct timeline, e.g., from Jan through to Dec, and not vice versa.

\* The last sub-annual time slice is linked to the first one to close the loop of the year. Parameter ``storage\_initial`` is to set an initial amount

\* for the content of storage in any desirable time slice (optionally). This initial value is a cost-free stored media that storage can discharge

\* in the same or following time slices. ``storage\_self\_discharge`` represents the self-discharge (loss) of storage as % of the level of stored media

\* in each time slice. This allows to model time-related losses in storage separately, in addition to charging and discharging losses.

\*

\* .. list-table::

\* :widths: 20 80

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - storage\_initial

\* - ``node`` | ``tec`` | ``level`` | ``commodity`` | ``year\_act`` | ``time``

\* \* - storage\_self\_discharge

\* - ``node`` | ``tec`` | ``level`` | ``commodity`` | ``year\_act`` | ``time``

\* \* - time\_order

\* - ``lvl\_temporal`` | ``time``

\*

\*\*\*

Parameters

storage\_initial(node,tec,mode,level,commodity,year\_all,time) initial content of storage

storage\_self\_discharge(node,tec,mode,level,commodity,year\_all,time) self-discharge (loss) of storage as % of storage level in each time slice

time\_order(lvl\_temporal,time) sequence of subannual time slices

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Soft relaxations of dynamic constraints \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_soft\_constraints:

\*

\* Cost parameters for 'soft' relaxations of dynamic constraints

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* The implementation of |MESSAGEix| includes the functionality for 'soft' relaxations of dynamic constraints on

\* new-built capacity and activity (see Keppo and Strubegger, 2010 :cite:`Keppo-2010`).

\* Refer to the section :ref:`dynamic\_constraints`. Absolute cost and levelized cost multipliers are used

\* for the relaxation of upper and lower bounds.

\*

\* .. list-table::

\* :widths: 20 80

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - abs\_cost\_new\_capacity\_soft\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - abs\_cost\_new\_capacity\_soft\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - level\_cost\_new\_capacity\_soft\_up

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - level\_cost\_new\_capacity\_soft\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - abs\_cost\_activity\_soft\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - abs\_cost\_activity\_soft\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - level\_cost\_activity\_soft\_up

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\* \* - level\_cost\_activity\_soft\_lo

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``time``

\*

\*\*\*

Parameters

abs\_cost\_new\_capacity\_soft\_up(node,tec,year\_all) absolute cost of dynamic new capacity constraint relaxation (upwards)

abs\_cost\_new\_capacity\_soft\_lo(node,tec,year\_all) absolute cost of dynamic new capacity constraint relaxation (downwards)

level\_cost\_new\_capacity\_soft\_up(node,tec,year\_all) levelized cost multiplier of dynamic new capacity constraint relaxation (upwards)

level\_cost\_new\_capacity\_soft\_lo(node,tec,year\_all) levelized cost multiplier of dynamic new capacity constraint relaxation (downwards)

abs\_cost\_activity\_soft\_up(node,tec,year\_all,time) absolute cost of dynamic activity constraint relaxation (upwards)

abs\_cost\_activity\_soft\_lo(node,tec,year\_all,time) absolute cost of dynamic activity constraint relaxation (downwards)

level\_cost\_activity\_soft\_up(node,tec,year\_all,time) levelized cost multiplier of dynamic activity constraint relaxation (upwards)

level\_cost\_activity\_soft\_lo(node,tec,year\_all,time) levelized cost multiplier of dynamic activity constraint relaxation (downwards)

;

\*\*\*

\* .. \_section\_parameter\_historical:

\* .. \_historical\_new\_capacity:

\*

\* Historical capacity and activity values

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* To model the transition of an energy system, the initial energy system with its energy mix

\* needs to be defined first. The historical activity and the historical new capacity do this.

\* These parameters have to be defined in order to limit the capacity in the first model period.

\*

\* Historical data on new capacity and activity levels are therefore included in |MESSAGEix| for

\* correct accounting of the vintage portfolio and a seamless implementation of dynamic constraints from

\* historical years to model periods.

\*

\* .. list-table::

\* :widths: 35 65

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - historical\_new\_capacity [#hist]\_

\* - ``node\_loc`` | ``tec`` | ``year\_vtg``

\* \* - historical\_activity [#hist]\_

\* - ``node\_loc`` | ``tec`` | ``year\_act`` | ``mode`` | ``time``

\*

\* The activity in the historic period can be defined with

\*

\* .. math::

\* \sum\_{m} \text{ACT}\_{n,t,y^V,y,m,h} \leq \text{duration\_time}\_{h} \cdot \text{capacity\_factor}\_{n,t,y^V,y,h} \\

\* \cdot \text{CAP}\_{n,t,y^V,y} \quad t \ \in \ T^{\text{INV}}

\*

\* and the historical new capacity with

\*

\* .. math::

\* \text{CAP\_NEW}\_{n,t,y^V} = \frac{\text{CAP}\_{n,t,y^V,y}}{\text{duration\_period}\_{y}}

\*

\* Both equations are equally valid for model periods. However, to calculate ``historical\_new\_capacity``

\* and ``historical\_activity`` all parameters must describe the historic period.

\*

\* The ``duration\_period`` of the first period (historic period) is set to the value that appears

\* most frequently in the model. If, for example, the majority of periods in the model

\* consists of 10 years, the ``duration\_period`` of the first period is likewise 10 years.

\*

\*\*\*

Parameters

historical\_new\_capacity(node,tec,year\_all) historical new capacity

historical\_activity(node,tec,year\_all,mode,time) historical activity

;

\*\*\*

\* .. \_section\_parameter\_investment:

\* .. \_remaining\_capacity:

\*

\* Auxiliary investment cost parameters and multipliers

\* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*

\* Auxiliary investment cost parameters include the remaining technical lifetime at the end of model horizon

\* (``beyond\_horizon\_lifetime``) in addition to the different scaling factors and multipliers as listed below.

\* These factors account for remaining capacity (``remaining\_capacity``)

\* or construction time of new capacity (``construction\_time\_factor``),

\* the value of investment at the end of model horizon (``end\_of\_horizon\_factor``)

\* or the discount factor of remaining lifetime beyond model horizon (``beyond\_horizon\_factor``).

\* ``remaining\_capacity\_extended`` is identical to ``remaining\_capacity`` but extended to include historical periods.

\*

\* .. list-table::

\* :widths: 35 50

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index names

\* \* - construction\_time\_factor

\* - ``node`` | ``tec`` | ``year``

\* \* - remaining\_capacity

\* - ``node`` | ``tec`` | ``year``

\* \* - remaining\_capacity\_extended

\* - ``node`` | ``tec`` | ``year``

\* \* - end\_of\_horizon\_factor

\* - ``node`` | ``tec`` | ``year``

\* \* - beyond\_horizon\_lifetime

\* - ``node`` | ``tec`` | ``year``

\* \* - beyond\_horizon\_factor

\* - ``node`` | ``tec`` | ``year``

\*

\*\*\*

Parameters

construction\_time\_factor(node,tec,year\_all) 'Scaling factor to account for construction time of new capacity'

remaining\_capacity(node,tec,year\_all,year\_all) 'Scaling factor to account for remaining capacity in period'

remaining\_capacity\_extended(node,tec,year\_all,year\_all) 'Same as remaining\_capacity, but including historical periods'

end\_of\_horizon\_factor(node,tec,year\_all) 'Multiplier for value of investment at end of model horizon'

beyond\_horizon\_lifetime(node,tec,year\_all) 'Remaining technical lifetime at the end of model horizon'

beyond\_horizon\_factor(node,tec,year\_all) 'Discount factor of remaining lifetime beyond model horizon'

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Emissions \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_emissions:

\*

\* Parameters of the `Emission` section

\* ------------------------------------

\*

\* The implementation of |MESSAGEix| includes a flexible and versatile accounting of emissions across different

\* categories and species, with the option to define upper bounds and taxes on various (aggregates of) emissions

\* and pollutants, (sets of) technologies, and (sets of) years.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - historical\_emission [#hist]\_

\* - ``node`` | ``emission`` | ``type\_tec`` | ``year``

\* \* - emission\_scaling [#em\_scaling]\_

\* - ``type\_emission`` | ``emission``

\* \* - bound\_emission

\* - ``node`` | ``type\_emission`` | ``type\_tec`` | ``type\_year``

\* \* - tax\_emission

\* - ``node`` | ``type\_emission`` | ``type\_tec`` | ``type\_year``

\*

\* .. [#em\_scaling] The parameter ``emission\_scaling`` is the scaling factor to harmonize bounds or taxes across types of

\* emissions. It allows to efficiently aggregate different emissions/pollutants and set bounds or taxes on various categories.

\*

\*\*\*

Parameters

historical\_emission(node,emission,type\_tec,year\_all) historical emissions by technology type (including land)

emission\_scaling(type\_emission,emission) scaling factor to harmonize bounds or taxes across types

bound\_emission(node,type\_emission,type\_tec,type\_year) upper bound on emissions

tax\_emission(node,type\_emission,type\_tec,type\_year) emission tax

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Land-use model emulator \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_landuse\_emulator:

\*

\* Parameters of the `Land-Use model emulator` section

\* ---------------------------------------------------

\*

\* The implementation of |MESSAGEix| includes a land-use model emulator, which draws on exogenous land-use scenarios

\* (provided by another model) to derive supply of commodities (e.g., biomass) and emissions

\* from agriculture and forestry. The parameters listed below refer to the assigned land scenario.

\*

\* .. \_land\_input:

\* .. \_land\_output:

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - historical\_land [#hist]\_

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - land\_cost

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - land\_input

\* - ``node`` | ``land\_scenario`` | ``year`` | ``commodity`` | ``level`` | ``time``

\* \* - land\_output

\* - ``node`` | ``land\_scenario`` | ``year`` | ``commodity`` | ``level`` | ``time``

\* \* - land\_use

\* - ``node`` | ``land\_scenario`` | ``year`` | ``land\_type``

\* \* - land\_emission

\* - ``node`` | ``land\_scenario`` | ``year`` | ``emission``

\* \* - initial\_land\_scen\_up

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - growth\_land\_scen\_up

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - initial\_land\_scen\_lo

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - growth\_land\_scen\_lo

\* - ``node`` | ``land\_scenario`` | ``year``

\* \* - initial\_land\_up

\* - ``node`` | ``year`` | ``land\_type``

\* \* - dynamic\_land\_up

\* - ``node`` | ``land\_scenario`` | ``year`` | ``land\_type``

\* \* - growth\_land\_up

\* - ``node`` | ``year`` | ``land\_type``

\* \* - initial\_land\_lo

\* - ``node`` | ``year`` | ``land\_type``

\* \* - dynamic\_land\_lo

\* - ``node`` | ``land\_scenario`` | ``year`` | ``land\_type``

\* \* - growth\_land\_lo

\* - ``node`` | ``year`` | ``land\_type``

\*

\*\*\*

Parameters

historical\_land(node,land\_scenario,year\_all) historical land scenario assignment

land\_cost(node,land\_scenario,year\_all) costs of land-use scenario

land\_input(node,land\_scenario,year\_all,commodity,level,time) commodity input requirement of land-use scenario

land\_output(node,land\_scenario,year\_all,commodity,level,time) commodity output (yield) of land-use scenario

land\_use(node,land\_scenario,year\_all,land\_type) land type used in specific scenario

land\_emission(node,land\_scenario,year\_all,emission) emissions from land-use scenario

initial\_land\_scen\_up(node,land\_scenario,year\_all) initial bound on land-scenario change (upwards)

growth\_land\_scen\_up(node,land\_scenario,year\_all) relative bound on land-scenario change (upwards)

initial\_land\_scen\_lo(node,land\_scenario,year\_all) initial bound on land-scenario change (downwards)

growth\_land\_scen\_lo(node,land\_scenario,year\_all) relative bound on land-scenario change (downwards)

initial\_land\_up(node,year\_all,land\_type) initial bound on land-type use change (upwards)

dynamic\_land\_up(node,land\_scenario,year\_all,land\_type) absolute bound on land-type use change (upwards)

growth\_land\_up(node,year\_all,land\_type) relative bound on land-type use change (upwards)

initial\_land\_lo(node,year\_all,land\_type) initial bound on land-type use change (downwards)

dynamic\_land\_lo(node,land\_scenario,year\_all,land\_type) absolute bound on land-type use change (upwards)

growth\_land\_lo(node,year\_all,land\_type) relative bound on land-type use change (downwards)

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Share constraints \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_share\_constraints:

\*

\* Parameters of the `Share Constraints` section

\* ---------------------------------------------

\*

\* Share constraints define the share of a given commodity/mode to be active on a certain level. For the mathematical

\* formulation, refer to :ref:`share\_constraints`.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - share\_commodity\_up

\* - ``shares`` | ``node\_share`` | ``year\_act`` | ``time``

\* \* - share\_commodity\_lo

\* - ``shares`` | ``node`` | ``year\_act`` | ``time``

\* \* - share\_mode\_up

\* - ``shares`` | ``node\_loc`` | ``technology`` | ``mode`` | ``year\_act`` | ``time``

\* \* - share\_mode\_lo

\* - ``shares`` | ``node\_loc`` | ``technology`` | ``mode`` | ``year\_act`` | ``time``

\*

\*\*\*

Parameters

share\_commodity\_up(shares,node,year\_all,time) upper bound of commodity share constraint

share\_commodity\_lo(shares,node,year\_all,time) lower bound of commodity share constraint

share\_mode\_up(shares,node,tec,mode,year\_all,time) upper bound of mode share constraint

share\_mode\_lo(shares,node,tec,mode,year\_all,time) lower bound of mode share constraint

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Generic linear relations \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_generic\_relations:

\*

\* Parameters of the `Relations` section

\* -------------------------------------

\*

\* Generic linear relations are implemented in |MESSAGEix|. This feature is intended for development and testing only - all new features

\* should be implemented as specific new mathematical formulations and associated \*sets\* & \*parameters\*. For the formulation of the relations,

\* refer to :ref:`section\_of\_generic\_relations`.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - relation\_upper

\* - ``relation`` | ``node\_rel`` | ``year\_rel``

\* \* - relation\_lower

\* - ``relation`` | ``node\_rel`` | ``year\_rel``

\* \* - relation\_cost

\* - ``relation`` | ``node\_rel`` | ``year\_rel``

\* \* - relation\_new\_capacity

\* - ``relation`` | ``node\_rel`` | ``year\_rel`` | ``tec``

\* \* - relation\_total\_capacity

\* - ``relation`` | ``node\_rel`` | ``year\_rel`` | ``tec``

\* \* - relation\_activity

\* - ``relation`` | ``node\_rel`` | ``year\_rel`` | ``node\_loc`` | ``tec`` | ``year\_act`` | ``mode``

\*

\*\*\*

Parameters

relation\_upper(relation,node,year\_all) upper bound of generic relation

relation\_lower(relation,node,year\_all) lower bound of generic relation

relation\_cost(relation,node,year\_all) cost of investment and activities included in generic relation

relation\_new\_capacity(relation,node,year\_all,tec) new capacity factor (multiplier) of generic relation

relation\_total\_capacity(relation,node,year\_all,tec) total capacity factor (multiplier) of generic relation

relation\_activity(relation,node,year\_all,node,tec,year\_all,mode) activity factor (multiplier) of generic relation

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Fixed variable values \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_fixed:

\*

\* Fixed variable values

\* ---------------------

\*

\* The following parameters allow to set variable values to a specific value.

\* The value is usually taken from a solution of another model instance

\* (e.g., scenarios where a shock sets in later to mimic imperfect foresight).

\*

\* The fixed values do not override any upper or lower bounds that may be defined,

\* so fixing variables to values outside of that range will yield an infeasible model.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - fixed\_extraction

\* - ``node`` | ``commodity`` | ``grade`` | ``year``

\* \* - fixed\_stock

\* - ``node`` | ``commodity`` | ``level`` | ``year``

\* \* - fixed\_new\_capacity

\* - ``node`` | ``technology`` | ``year\_vtg``

\* \* - fixed\_capacity

\* - ``node`` | ``technology`` | ``year\_vtg`` | ``year\_act``

\* \* - fixed\_activity

\* - ``node`` | ``technology`` | ``year\_vtg`` | ``year\_act`` | ``mode`` | ``time``

\* \* - fixed\_land

\* - ``node`` | ``land\_scenario`` | ``year``

\*

\* Note that the variable :math:`\text{STOCK\_CHG}` is determined implicitly by the :math:`\text{STOCK}` variable

\* and therefore does not need to be explicitly fixed.

\*\*\*

Parameters

fixed\_extraction(node,commodity,grade,year\_all) fixed extraction level

fixed\_stock(node,commodity,level,year\_all) fixed stock level

fixed\_new\_capacity(node,tec,year\_all) fixed new-built capacity

fixed\_capacity(node,tec,vintage,year\_all) fixed maintained capacity

fixed\_activity(node,tec,vintage,year\_all,mode,time) fixed activity level

fixed\_land(node,land\_scenario,year\_all) fixed land level

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Auxiliary reporting parameters \*

\*----------------------------------------------------------------------------------------------------------------------\*

\*\*\*

\* .. \_section\_parameter\_auxiliary\_reporting:

\*

\* Auxiliary reporting parameters

\* ------------------------------

\*

\* The following parameters are used for reporting (post-processing) solved models. They assign monetary value to

\* the `net` total system costs from trading and emission taxes (``total\_cost``). Morevoer, they also assign a value

\* to the `total` trade of commodities (the difference between the revenues from exports and the costs of imports,

\* ``trade\_cost``) and to the costs from importing (``import\_cost``) and the revenues from exporting (``export\_cost``)

\* in each node and year.

\*

\* .. list-table::

\* :widths: 25 75

\* :header-rows: 1

\*

\* \* - Parameter name

\* - Index dimensions

\* \* - total\_cost

\* - ``node`` | ``year``

\* \* - trade\_cost

\* - ``node`` | ``year``

\* \* - import\_cost

\* - ``node`` | ``commodity`` | ``year``

\* \* - export\_cost

\* - ``node`` | ``commodity`` | ``year``

\*\*\*

Parameters

trade\_cost(node, year\_all) net of commodity import costs and commodity export revenues by node and year

import\_cost(node, commodity, year\_all) import costs by commodity and node and year

export\_cost(node, commodity, year\_all) export revenues by commodity and node and year

;

\*----------------------------------------------------------------------------------------------------------------------\*

\* Auxiliary parameters for GAMS workflow \*

\*----------------------------------------------------------------------------------------------------------------------\*

Parameters

cap\_comm 'Equivalent to MESSAGE\_CAP\_COMM' / %MESSAGE\_CAP\_COMM% /

ctr counter parameter for loops

status(\*,\*) model solution status parameter for log writing

;