Simple Lotka-Volterra like model

This example describes a simple predator prey model. The capability of **CEMoS** describing box models is not used here. However, every state variable must be indexed. Such the two states predator and prey gets the index 1 (see implementation below).

Then the model is described by the following ODE system

$$\begin{aligned}\dot{X} &= \alpha \cdot X \left(1 - \frac{X}{\kappa_1}\right) - \beta \frac{\cdot X}{X + \kappa_2} \cdot Y \\ \dot{Y} &= q_n \cdot \beta \frac{\cdot X}{X + \kappa_2} \cdot Y - \mu \cdot Y\end{aligned}$$

with

- α the maximum the growth rate of the prey X
- β, the maximum gross uptake rate of the predator Y
- q_n the net part of the uptake
- κ_1 the maximum capacity of the prey
- κ_2 the capacity of the prey, where the actual gross uptake rate is half of its maximum
- μ the mortality of the predator Y

The model is implemented in a directory called `LV`. The model code is located in the subfolder `main`. The following files are needed

`model.def`, model definitions in `LV/main`

`par.def`, parameter definitions in `LV/main`

`model.c`, **C**-code in `LV/main`

`cemos.par`, simulation control in `LV`

Additionally a file `batch.bat` is included which shows the structure of batch files.

model.def

The file `model.def` contains the central variables and parameters (in this example only the states):

```
%numeric double

%states
X[1]={100.0};
Y[1]={10.0};
```

The line `%numeric double` forces **CEMoS** to simulate the model in double precision, the identifier `%states` marks the definition block for state variables.
Remark: states must be defined as one dimensional arrays and initial values must be set.

par.def

The file `par.def` contains the definition of model parameters.

```
%real_par
alfa=0.5;      /* maximum birth rate of prey in 1/month*/
kappa1=1000.0; /* capacity of prey in kg*/
beta=0.5;      /* maximum gross uptake rate of predator in 1/month*/
qn=0.5;        /* net part of uptake*/
mue=0.1;       /* mortality of predator */
kappa2=200.0;  /* capacity of prey, where uptake rate of predator
                  is half of the maximum in kg*/
%change
```

The identifier `%real_par` marks the block of scalar real parameters. At the start of the simulation the here defined parameters will get their values. The identifier `%change` assigns an overruling of the values from the definition block by these values without recompiling the model. The content of this file can also be set in the file `model.def`, but the splitting into different files keeps care of the track, if the models become larger.

model.c

The file `model.c` contains the **C**-code of the model:

```
#include "struct.h"
#include "par.h"
#include <math.h>

void model(void)
{
 SX[1] = alfa*X[1]*(1.0  - X[1]/kappa1) - beta*X[1]/(X[1]+kappa2)*Y[1];
 SY[1] = qn*beta*X[1]/(X[1]+kappa2)*Y[1] - mue*Y[1];
}
```

With the `#include`-statements the variables defined in the previous files are made available, where the files `struct.h` and `par.h` are automatically generated by **CEMoS** from the files `model.def` and `par.def`.
The temporary derivatives ($\dot{X}$) are identified in **CEMoS** by a leading `S`, these `S` variables are automatically generated by **CEMoS** and available for all state variables.

cemos.par

The file `cemos.par` contains all information of the duration of a simulation, the integration method etc., the `\%change`-statement for variables and parameters defined in the `model.def` , as well as the `\%store`-statement for the information about variables to be stored.

```
#include "main/model.def"

%simulation_parameters
startim=0.0;
endtime=500.0;
storetime=0.;
outdelt=0.1;
year=0.0;
cycle=0.0;
model_dir main

%integration_par
mindelt=1.e-15;
maxdelt=0.01;
relrate=1.0;
relchange=0.5;
accuracy=0.001;
method=3;

%change
X[1]={50.0};
Y[1]={10.0};
%store
X[1];
Y[1];
```

For the description of the simulation parameters and the integration methods see **CEMoS** manual.