

World3 in Modelica: Creating System Dynamics Models in the Modelica Framework

François E. Cellier
ETH Zürich
Switzerland
FCellier@Inf.ETHZ.CH

Abstract

This paper introduces a new release of the SystemDynamics library of Modelica and shows how it is being used by discussing a fairly large application code: Meadows' World3 model. The newest version of that model has been made available in the library.

Keywords: *System Dynamics, World Dynamics, Soft Science Modeling*

1 Introduction

System Dynamics represents a fairly low-level modeling paradigm. Its implementation does not place heavy demands on the modeling software. Hence Modelica may in fact be a bit of an overkill for dealing with System Dynamics models. However, it is considerably better suited than the state-of-the-art software for this type of modeling, i.e., Stella [11], the code that most System Dynamics modelers use today.

A first version of a System Dynamics library for Modelica was released in 2002 [3]. In the present paper, a new release, SystemDynamics 2.0, is being discussed. SystemDynamics 2.0 is not an upgrade of SystemDynamics 1.0, but rather a re-implementation of the methodology. Inherited from SystemDynamics 1.0 were only two application codes, a small introductory model concerning lynxes eating hares, and a considerably more complex model borrowed from Forrester's Industrial Dynamics book [4].

As already mentioned above, the basic models implementing the System Dynamics methodology, levels and rates, are so simple that their implementation in Modelica requires very little time and effort. The value of the library is not in its basic models, but rather in its application codes.

Among other applications, SystemDynamics 2.0 offers two full World models, namely Forrester's World2 model [5], and Meadows' World3 model [7,8].

Whereas Forrester described his model in full in his World Dynamics book [5], Meadows' only talked in Limits to Growth about the results obtained with the model [8]. The model itself, originally coded in Dynamo [10], was described in a separate book [7].

Meadows' World3 model has seen two major upgrades since its original inception, one in 1992, i.e. after 20 years, and the second in 2002, i.e., after 30 years. The World3 application code contained in SystemDynamics 2.0 implements the 2002 version of the World3 model. In the code, we offer not only the basic model, but also all 10 scenarios that Meadows and co-workers are talking about in Limits to Growth: The 30-Year Update [8].

Although the work of Forrester and Meadows caused quite a stir in the early 70s when their books first appeared, world modeling became unfashionable fairly quickly, because essentially all sources of funding dried out for political reasons.

Only very recently, in the context of the looming *Peak Oil* event and because of the ongoing discussions concerning *Global Warming*, has world modeling become respectable again.

It turned out that Forrester and Meadows were essentially correct in their assessments, in spite of the fact that their models were very crude in comparison with real world dynamics.

With this paper, I wish to open up world modeling to the community of Modelica users.

2 Short History of System Dynamics

The *System Dynamics* approach to modeling dynamic systems was developed in the 1960s by Jay Forrester with the aim of creating a modeling and simulation tool that economists would be able to handle.

Instead of talking about differential equations, he talked about “levels,” the values of which were changed by “rates.” Level variables are variables that can accumulate. For example, population might be used as a level variable. It is controlled by two rate variables, the birth rate and the death rate.

Forrester would draw this relationship in a diagram similar to the one shown in Fig.1.

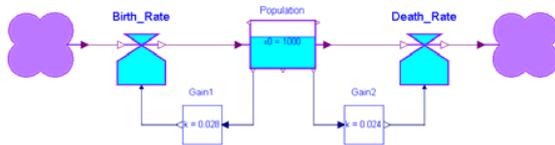


Figure 1: Population with birth and death rates

The blue square box represents a level. It requires an initial value. The blue icons to the left and right of the level represent rates. Both the birth and the death rate are proportional to the population. The two clouds represent sources and sinks of material. They are used for documentation purposes only. There are no equations associated with these models. The lilac lines represent material flows, whereas the blue lines represent information flows.

Of course, Forrester didn’t have a computer available with a graphical user interface. He drew his diagrams only by hand and then translated them manually (and quite mechanically) down to a set of equations that he then encoded in Dynamo [10], a simulation “language” that had been outdated already at the time of its creation.

Forrester explained to his disciples that every modeling exercise should always start with pondering, which are the most important accumulator variables that ought to be captured in the model. These variables should be declared as level variables. Subsequently, it needs to be decided, what other variables can be viewed as inflows and outflows to and from these levels. The inflows and outflows would then become the rate variables. Fig.2 shows a typical set of levels and their rates.

Levels	Rates	
	Inflows	Outflows
Population	Birth Rate	Death Rate
Money	Income	Expenses
Frustration	Stress	Affection
Love	Affection	Frustration
Tumor Cells	Infection	Treatment
Inventory on Stock	Shipments	Sales
Knowledge	Learning	Forgetting

Figure 2: Typical level and rate variables

The modeler would then need to decide, which other variables the rates depend on, and write these down in a so-called “laundry list.” A possible laundry list for the birth rate is offered in Fig.3.



Figure 3: Birth rate laundry list

So far so good, but now comes the most daring assumption, the “quantum leap” of System Dynamics.

The functional relationship represented by such a laundry list can be assumed to be a static non-linear function in multiple variables, e.g.:

$$Birth_rate = f(Population, Pollution, Food, Crowding, Material_Standard_of_Living)$$

Yet, since such a function may be too difficult to identify, Forrester chose to ignore the mutual relationship among the different input variables, and postulate the following model instead:

$$Birth_rate = BRN \cdot Population \cdot f_1(Pollution) \cdot f_2(Food) \cdot f_3(Crowding) \cdot f_4(Material_Standard_of_Living)$$

The birth rate is essentially computed as the average birth rate, *BRN*, multiplied by the population. All other dependencies are expressed as small signal deviations from the norm. The single-valued functions can most of the time be easily approximated using information from the open literature, e.g. from statistical yearbooks.

Forrester was wildly successful with his approach to modeling. Whereas engineers and physicists mostly ignored him, if they didn’t even sneer at his “methodology,” researchers from the soft sciences loved it. Already by the early 1980s, several thousands of papers making use of System Dynamics for a variety of modeling projects had been published [6].

By 1984, the Macintosh became available, and with it, programmers were for the first time offered an easily programmable graphical user interface. Within a short time, a graphical modeling environment for System Dynamics modeling, Stella [11], became available that quickly replaced Dynamo [10] as the tool of choice for System Dynamics modeling.

Today, more than 20 years later, Stella is still the most widely used tool for System Dynamics modeling. The language has seen a few improvements over the years, but by and large, it is still the same software that had been created in the mid 1980s.

A Stella model of population and its two rate variables is shown in Fig.4.

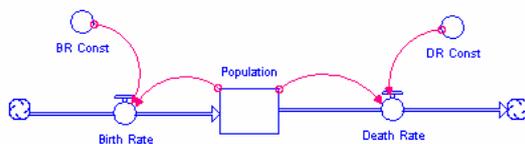


Figure 4: Stella model of population growth

3 The WORLD3 Model

World Dynamics became quickly one of the most prominent endeavors of System Dynamics modelers. Among the earliest world models created for the Club of Rome were Forrester's WORLD2 and Meadows' WORLD3 models. Both of these models are made available as part of the new SystemDynamics library.

Which are the most important drivers (accumulators) behind any world model? The list of levels ought to include at least:

- population
- pollution
- resource utilization
- invested capital
- work force
- food

Different world models vary in the degree of sophistication, with which they consider these sectors. In this paper, we shall primarily focus on the WORLD3 model, as this model has been upgraded several times, and therefore is still up-to-date.

3.1 Population Dynamics

The population dynamics model of WORLD3 is shown in Fig.5.

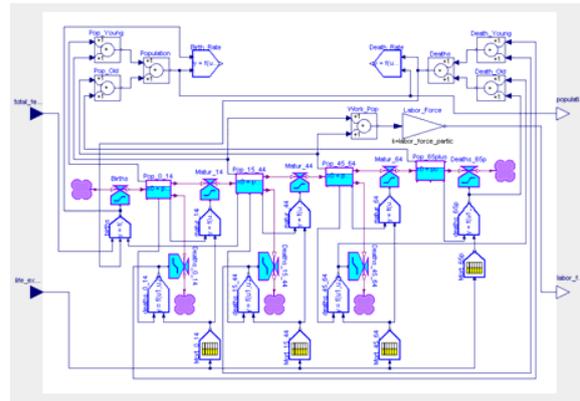


Figure 5: Population dynamics in WORLD3

The model is quite easy to read. The population is subdivided into four separate levels, representing:

1. children (until age 14)
2. young adults (until age 44)
3. older adults (until age 64)
4. seniors

This division makes sense, as the work force is comprised of groups #2 and #3 only, and people of reproductive age are those in group #2. The rates between the levels compute the maturation from one group into the next. Beside from the births and the final deaths, there are also people dying prematurely out of each of the four groups.

The birth rate depends on the fertility, which is computed by another module. The death rates in the four groups are modeled as tabular functions of the life expectancy, which is also computed elsewhere.

The model exports the total population and the labor force, as these variables are used by other modules.

Notice that WORLD3 is a global model. All variables are averaged over the entire globe. The model does not distinguish between Europe and Africa, for example. This limits the types of questions that may be answered by it.

3.2 Pollution Dynamics

The pollution dynamics model of WORLD3 is depicted in Fig.6.

The pollution model contained originally a single state variable: the accumulated pollution. New pollution is being generated in proportion to the total resource utilization and in proportion to the arable land used for agriculture. Pollution is being assimilated again in proportion to the accumulated pollution by the self-regulating mechanisms of this planet.

ments, the work force, land fertility, the human ecological footprint, the human welfare index, life expectancy, and last but not least the utilization of non-recoverable natural resources. Three of those were presented in the previous sections of this paper. The overall model invokes one of each of the 13 sector models and connects the terminal variables of those sector models among each other.

We are now ready to simulate the model. The compiled model contains 41 state variables and 265 algebraic variables. A few simulation results are shown in Figs.9 and 10.

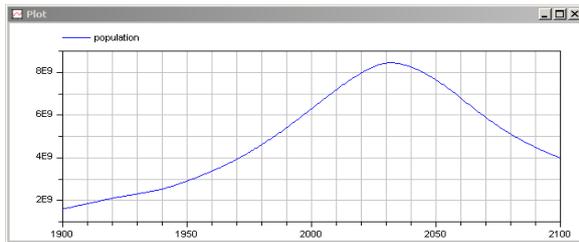


Figure 9: Population as a function of time

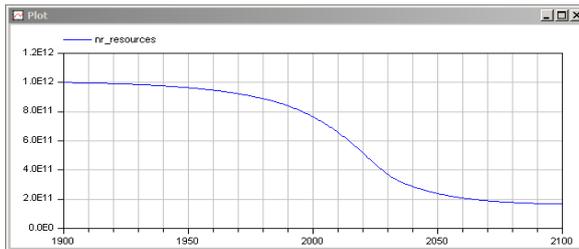


Figure 10: Natural resources as a function of time

The simulation results are identical to those shown in the book *Limits to Growth* [8]. The population grows until roughly 2030. At that time, the non-recoverable resources have been depleted to an extent where production can no longer proceed as before. In particular, less food gets produced, which leads to a decline in the population.

Can we trust these results? To answer this question, it may be useful to look at scenario #2. In this scenario, Meadows and his co-workers postulated that the amount of the remaining non-recoverable natural resources had been massively underestimated. The amount still available in 1900 is thus doubled. Furthermore, it is proposed that, in 2002, money is being invested in producing the remaining resources more efficiently.

Some simulation results of this scenario are shown in Figs. 11 and 12. The results from scenario #1 are superposed for comparison. We would expect that, since resource depletion won't occur as quickly,

the population can continue to grow for some time after 2030.

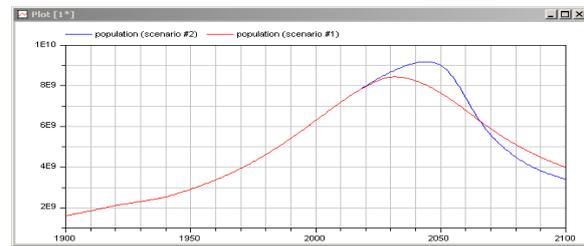


Figure 11: Population as a function of time (scenario #2)

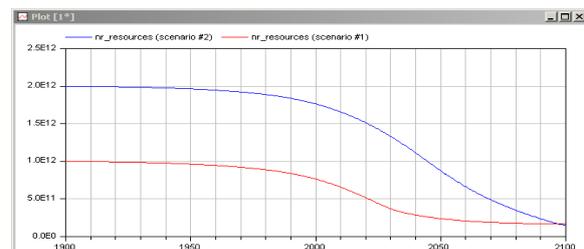


Figure 12: Resources as a function of time (scenario #2)

In this scenario, the population is indeed able to grow for a little while longer, but now it starts shrinking at 2045, although the resources aren't getting depleted until 2080. This time around, the cause of the die-off is the pollution. Pollution is allowed to continue to increase unabated, which eventually hampers our ability to grow food.

Whereas scenario #1 suffers (in a general sense) the effects of *Peak Oil*, scenario #2 is plagued by *Global Warming*. Similar results were shown in earlier editions of *Limits to Growth* [8]. The main difference between the models is the year, in which corrective action is being taken in the different scenarios. In the first edition of the book, corrective actions were taken in 1972. However, we already know that this didn't happen. Hence, the 3rd edition proposes corrective actions to take place in 2002 only. By postponing the intervention, the window of opportunity for still influencing the simulation results in a significant way shrinks.

Why do I believe these results? It is, because they aren't very sensitive to the scenario chosen. Whatever we do, if it is not one factor that brings us to the limits of growth, it is another ... and irrespective of what we do, it always happens in the 21st century. It may happen a few years earlier or a few years later, but the general picture doesn't change at all.

Also the (much simpler) *WORLD2* model that features a different set of state variables and different

interactions between them essentially paints the same picture.

Since the 1980s, we are consuming more resources per time unit than the planet is able to regrow [1,12]. We are living beyond our means. This is not sustainable. It cannot continue indefinitely.

So, will the decline take place? Maybe it won't. Maybe the moon is made out of Swiss cheese.

3.5 Analysis of Simulation Results

Meadows and co-workers found two scenarios that look a bit more hopeful. These are scenarios #6 and #9. Let me analyze these two scenarios in more detail. To this end, we shall continue the simulation all the way until 2500.

In scenario #6, a whole palette of interventions was enacted in 2002. These include the interventions of scenario #2. In addition, money was invested in improved pollution control technology (scenario #3), in enhanced land yield (scenario #4), in increased land erosion control (scenario #5), and in augmented resource utilization efficiency (scenario #6).

Some simulation results are depicted in Fig.13.

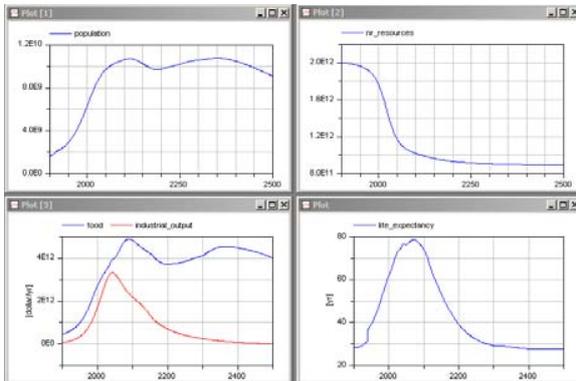


Figure 13: Simulation results of scenario #6

This scenario is indeed sustainable. The world population hovers at approximately 10 billion people. The remaining natural resources get no longer consumed.

Yet, humanity is paying a heavy price for insisting on maintaining such a large population. It spends all of its resources in producing food, and does so with the most primitive of means. The industrial output, and also the service sector output get reduced to almost zero. This is also why the remaining natural resources are no longer being consumed. Worst of all, the life expectancy is back at a value as it was experienced prior to the industrial age. The

average human dies before age 30 due to huge infant mortality.

Let us now look at scenario #9. In that scenario, additional interventions are chosen. The scenario starts out with scenario #6, but in addition enforces strict population control (scenario #7), and products are being built that last 25% longer on average (scenarios #8 and #9).

Some simulation results are depicted in Fig.14.

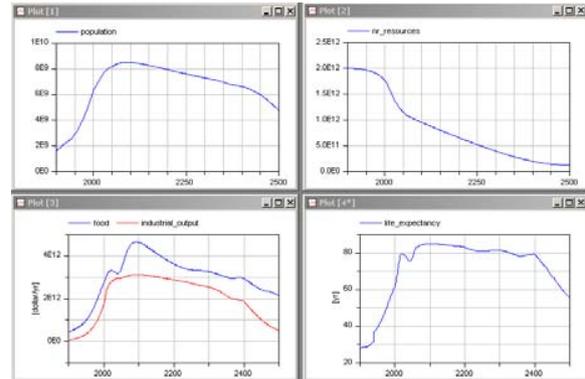


Figure 14: Simulation results of scenario #9

By enforcing strict population control, the world population is kept at a maximum value of 8 billion people. The scenario promises a golden age that will last for 400 years. Unfortunately, the scenario is not fully sustainable, as the natural resources continue to be used up, and by the year 2400, the industrial output, and with it also the population and life expectancy start declining again.

4 Dymola vs. Stella

What have we gained by offering a System Dynamics modeling capability in Dymola and by porting the WORLD3 model to that new environment?

Stella, contrary to Modelica, is not truly object-oriented. Large models are handled in Stella by supporting the concept of a virtual canvas. The physical screen can be scrolled over the virtual canvas, enabling the user to look at parts of the model separately. However, there is no feature available that would help a user find a particular spot, such as the population dynamics model, on the canvas.

Stella furthermore does not offer an icon editor. Stella only supports three types of icons that are all displayed in Fig.4. The square boxes represent levels (or “stocks,” as they are being called in Stella); the circles with the tap on top denote the rates (or “flows,” as they are being named in Stella), and the circles without a tap are everything else (linear and

non-linear functions, tabular functions). For this reason, Stella diagrams don't offer mnemonic hints. They look all the same, irrespective of what they represent (just like a bond graph [2]).

The numerical ODE solvers offered by Stella are rather poor. Also, Stella computes internally with an accuracy of 2 digits after the comma only (triggered by the fact that Stella is frequently used by economists who think in terms of dollars and cents).

On the other hand, Stella offers better support in dealing with tabular functions. Each 1D table is immediately plotted in the parameter window, and the user can tweak the curve by moving supporting values around using the mouse.

Furthermore, Dymola forces the user to create a separate block for each non-linear function and program the non-linear relationship either graphically in its diagram window or alphanumerically in its equation window. In contrast, Stella offers a generic non-linear function block that enables the user to create the non-linear relationship interactively in the parameter window of that generic block. The user doesn't even need to retype the names of the input variables. The parameter window of that generic function offers a scroll-down list of the names of all input variables, and the user can simply click on any of those in order to get them included in the expression.

Finally, Modelica has been designed by engineers for engineers. It is based heavily on SI units. Whereas the user can declare types based on these units, he cannot declare new units. Whereas this works well for most engineering endeavors, it causes problems when dealing with soft science models.

Sometimes, new derived units are needed. For example, time in System Dynamics models is often measured in years rather than seconds. Whereas Dymola offers the possibility to declare new display units, the user cannot change the units used in computations. This is inconvenient. Of course, the types encoded in the SIunits library are based on SI units. Thus, if a user wishes to declare his own units, he will have to declare his own types based on these units also.

Even worse, however, are those units that cannot be expressed at all in terms of SI units. For example, many System Dynamics models operate on units of money. Dollars cannot be expressed in terms of SI units at all.

The most important advantage of Dymola is the fact that the entire System Dynamics knowledge is encoded at the level of Modelica. The interface can therefore be easily modified and enhanced by the

user. In contrast, Stella's user interface is completely hard-wired. The user cannot modify the syntax or semantics of Stella in any way, and therefore, new ideas cannot be incorporated into the code except by talking the designers of the tool into including them with their next software release.

5 Conclusions

In this paper, a new release, or rather re-implementation, of the System Dynamics library of Modelica was presented.

The System Dynamics methodology is very easy to use, and consequently, does not really require much of an introduction. The most important value of a System Dynamics library is the knowledge encoded in its application examples. Currently, the by far most valuable part of the new library are its world models.

What future additions are in the works? In today's world of dwindling fossil fuel reserves, it becomes important to track how much energy we are actually using. Whereas classical System Dynamics is designed to track material flows, it does not track energy flows. This is a major drawback of the methodology.

For this reason, a second version of the System Dynamics library has also been released as a sub-library of BondLib [2], our bond graph library. In that version, all material flows are represented internally by bond graphs. A bond graph naturally tracks energy flows. Each energy flow, in that version of the library, is represented as the product of a specific enthalpy and a mass flow. Hence we can track material flows and energy flows simultaneously.

When I drive my car from home to work, I am not only spending energy in the form of the gas that my car consumes. Some energy was also spent in producing the car, and more energy will be spent in discarding it at the end of its lifecycle and in recovering those materials from it that can be recycled.

The accumulated energy that accounts for all of those indirect uses of energy is called *emergy* [9]. The specific enthalpy can be used to encode in the model the *specific emergy*, i.e., the emergy per unit of mass.

I plan on porting examples of emergy modeling, as described in the publications by Howard Odum, over to the bond graph implementation of the System Dynamics library, but this work has not yet been completed.

Acknowledgments

I wish to express my gratitude to Dennis Meadows and Jørgen Randers. Without their generous help and support, I would have been unable to port the WORLD3 model and all of its 10 scenarios over from Stella to Dymola (Modelica).

References

- [1] Cellier, F.E.: [Ecological Footprint, Energy Consumption, and the Looming Collapse](#). *The Oil Drum*, May 16, 2007
- [2] Cellier, F.E., Nebot, A.: [The Modelica Bond Graph Library](#). In: *Proceedings of the 4th International Modelica Conference*, Hamburg-Harburg, Germany (2005) Vol. 1, 57-65
- [3] Fabricius, S.M.O.: [SystemDynamics Modelica Library: Brief Feature and Example Documentation](#). Modelica Website, 2002
- [4] Forrester, J.W.: [Industrial Dynamics](#). M.I.T. Press, 1961
- [5] Forrester, J.W.: [World Dynamics](#). Pegasus Communications, 1971
- [6] Lebel, J.D.: System Dynamics. In: [Progress in Modelling and Simulation](#) (F.E. Cellier, ed.), Academic Press, London (1982) 119-158
- [7] Meadows, D.L., Behrens III, W.W., Meadows, D.H., Naill, R.F., Randers, J., Zah, E.K.O.: [Dynamics of Growth in a Finite World](#). Wright-Allen Press, 1974
- [8] Meadows, D.H., Randers, J., Meadows, D.L.: [Limits to Growth: The 30-Year Update](#). Chelsea Green, 2004
- [9] Odum, H.T.: [Environmental Accounting: Emergy and Environmental Decision Making](#). John Wiley, 1995
- [10] Richardson, G.P., Pugh III, A.L.: [Introduction to System Dynamics Modeling with DYNAMO](#). M.I.T. Press, 1981
- [11] Richmond, B., Peterson, S., Vescuso P.: [An Academic User's Guide to STELLA](#). High Performance Systems, Inc., Lyme, N.H., 1987
- [12] Wackernagel, M., Rees, W.: Our Ecological Footprint. *Green Teacher*, 45 (1995) 5-14



François E. Cellier received his BS degree in electrical engineering in 1972, his MS degree in automatic control in 1973, and his PhD degree in technical sciences in 1979, all from the Swiss Federal Institute of Technology (ETH) Zurich. Dr. Cellier worked at the University of Arizona as professor of Electrical and Computer Engineering from 1984 until 2005. He recently returned to his home country of Switzerland. Dr. Cellier's main scientific interests concern modeling and simulation methodologies, and the design of advanced software systems for simulation, computer aided modeling, and computer-aided design. Dr. Cellier has authored or co-authored more than 200 technical publications, and he has edited several books. He published a textbook on Continuous System Modeling in 1991 and a second textbook on Continuous System Simulation in 2006, both with Springer-Verlag, New York.