ModelWorks

A. Fischlin   May 2001

What Is ModelWorks?

ModelWorks is a modelling and simulation environment programmed in Modula-2 (WIRTH, 1985, 1988) specifically designed to be run interactively on modern working stations and personal computers and is one of the tools contained in the set RAMSES (Research Aids for the Modelling and Simulation of Environmental Systems) (FISCHLIN, 1991). It offers a handy user interface allowing for efficient alterations of model and simulation run parameter values and other specifications interesting for executing simulation experiments.

Moreover ModelWorks supports modular modelling by featuring a coupling mechanism between submodels and an unrestricted number of so-called model objects, such as state variables, model parameters etc. (FISCHLIN *et al.*, 1994). This allows to build models of any complexity in a well structured way. The software imposes no limits on the number of models, nor model objects, nor on the structure of the model equations or the model architecture; limits are solely determined by the physical memory and execution speed of the used computer. Yet, rigorous mathematical formalisms are available and provide a solid basis, despite the pitfalls lurking because of the complexity of the system structure.

What Makes ModelWorks Unique?

- First ModelWorks supports modular modeling. With ModelWorks you can formulate and integrate models of the three standard model formalisms, i.e. systems of coupled ordinary differential equations (DESS - Differential Equation System Specification), difference equations (SQM - Sequential Machines), or continuous-time discrete event systems (DEVS - Discrete Event System Specification). Hence models of a single or mixed type in any combination, such as continuous time or discrete time alone, or continuous and discrete time mixed with each other can be easily implemented. Each model can have its own integration routine, functioning on a different time step, depending on precision and efficiency requirements. Furthermore it is possible to declare or remove dynamically models and model objects in the middle of a simulation run (models can generate new models with a structure depending on the obtained simulation results and can be added to the simulation experiment without having to interrupt the ongoing runs etc.).

- Another of ModelWorks main strength is its open system architecture, which allows you to extend it freely in any way. ModelWorks does not feature a simulation language, but offers the means to formulate models and to control simulation experiments within a high level programming language. The hereby gained access to the power of an ordinary programming language is particularly important if you work with non-standard model formalisms, e.g. recursively defined model equations, or if you want to customize the display of simulation results, or if you wish to perform a particular sensitivity analysis or parameter identification at the end of a modelling and simulation study. With ModelWorks your investments are protected, since you will not have to switch to completely new simulation tools in order to accomplish such tasks.

- Thirdly, ModelWorks provides interactive simulation, since it is based on the Dialog Machine (FISCHLIN & SCHAUFELBERGER, 1987; FISCHLIN *et al.*, 1987; KELLER, 1989) , a user dialog controlling software layer between application and system software. In general the Dialog Machine substantially facilitates the programming of interactive programs on modern working stations with a windowing system, a pointing device such as a mouse, and a graphical user interface (FISCHLIN *et al.*, 1987). Thanks to the open system architecture and the Dialog Machine, you can also freely customize the user interface by extending the provided standard user interface or by completely replacing the latter with your own, satisfying your specific needs. This may facilitate the controlling of simulation experiments or the programming of special purpose animations, or allow you to quickly transform a research model into a demo version with a simple and robust user interface, which may then be used by practitioners and extension services. Thanks to the Dialog Machine, a port of ModelWorks to a new machine where the Dialog Machine is available is relatively simple. Currently it is possible to port –

without having to change the source code at all – large computer models developed on either a Mac, an IBM PC, or a Unix machine among all three platforms.

- Finally, there is also the so-called Easy ModelWorks, a simple tool for full interactive modeling of DESS and SQM systems. Easy ModelWorks provides only a small subset of the functionality of the full fledged ModelWorks, but the latter is often only needed by the sophisticated simulationist. Since Easy ModelWorks models can be transformed into an ordinary ModelWorks model with a single command, Easy ModelWorks provides also an easy and efficient entry into the world of ModelWorks.

Easy ModelWorks

How to Learn More on ModelWorks?

A full fledged manual which describes the both the implementation of ModelWorks for the Macintosh as well as the IBM PC is available as part of this software release (see folder «MW» within folder «Docu» and double click the file «ModelWorks Manual.sea»). It can be printed on any laser or inkjet printer (instructions provided when you unpack the sea).

Be warned, we do not expect you to be able to use ModelWorks without reading the essential parts of the manual! Although you may comprehend the standard user interface of ModelWorks without a manual (we also avoid reading boring manuals), modeling and simulation are based on concepts which require a bit of study. We made lots of efforts to provide you a manual worth-reading.

The following explanations on the manual may help you to decide how to use the ModelWorks manual. It consists mainly of the following four parts:

1) The **Tutorial** explains the basic concepts and features of ModelWorks together with a step by step guided tour to learn the elementary usage of ModelWorks. **The Tutorial is a MUST for everybody interested in using ModelWorks.**

2) For any serious modelling work to be done with ModelWorks, e.g. if you have decided to use it throughout a research project such as your dissertation, you should also read the second section **Theory**. It explains all theoretical concepts ModelWorks is based on. This section is **highly recommended for researchers** planning to use ModelWorks in a large research projects**.**

3) The third section **Reference** contains a detailed description of all features of ModelWorks. It is only to be consulted as needs arise.

4) The fourth section **Appendix** contains first a most useful section featuring sample models, from very basic to rather advanced, complex models (full listings featuring every detail). Again, this section is **highly recommended for researchers** planning to use ModelWorks in their research projects. This section contains also detailed explanations of auxiliary libraries, e.g. mathematical functions, random number generators, statistical analysis routines, graphic utilities etc., which are likely to be of high demand in a simulation environment. At the end come quick reference listings for the convenient usage of all software, i.e. of all auxiliary library modules, the Dialog Machine, and ModelWorks for the readers convenience.

Finally note, there is also a separate document (FISCHLIN *et al.*, 1995) with the file name «RAMSES Install&Tech Report» (distributed again in electronic form as part of RAMSES) wich explains the installation, the usage of the underlying MacMETH language system (WIRTH *et al.*, 1992), information on the use of the Dialog Machine (FISCHLIN & SCHAUFELBERGER, 1987; FISCHLIN *et al.*, 1987; KELLER, 1989) together with ModelWorks, how to make stand-alone applications etc.

Licensing, Copyrights and Other Legal Stuff

ModelWorks is **FreeWare** (but not public domain software) and is distributed on a non-profit and non-commercial basis only! We just thought that there might be some other members of the scientific community out there, who would be interested to share with us some of the more practical results from our research.

ModelWorks may not be redistributed for profit or sold by any means without prior written consent by the authors.

You are entitled to give a copy of this software to anybody you wish, but make sure to distribute it unmodified only. Please redistribute it exactly in the form as you received it on the disk where you found this document. All rights are reserved and any modifications are handled solely by the listed persons respectively institutions.

© Authors and ETH Zurich claim all copyrights.

ModelWorks for the Macintosh comes as part of the RAMSES package. It contains all software needed to edit, compile and execute models (see also topic «Where to obtain ModelWorks?)».

For the IBM PC version however, you have to purchase additional licenses, in particular you require Windows and the Logitec Modula-2 development system (There is also an old, GEM version available where you require other licenses). However, all other underlying software, i.e. the Dialog Machine and the auxiliary library, are also available as freeware together with ModelWorks.

Hardware Requirements

In order to run ModelWorks on a Macintosh™ you require at least 512K of RAM memory and at least two 800 KB floppies (for serious work, however we recommend a hard disk (the RAMSES package to which ModelWorks belongs requires uncompressed about 10 MB of hard disk space) and at least 1 MB RAM (preferably 4 MB)).

For the IBM PC version (under Windows) you require at least 8 MBytes RAM and a graphics monitor with at least EGA resolution and a large hard disk.

For the IBM PC version (under GEM) you require at least 640 KByte RAM and a graphics monitor with at least EGA resolution and a hard disk.

For more details on hard- and software requirements see (FISCHLIN *et al.*, 1995).

Where to Get ModelWorks?

ModelWorks can be obtained via anonymous ftp from the host

ftp.ito.umnw.ethz.ch

(formerly baikal.ethz.ch / Internet address: 129.132.80.130) in the directory pub/mac/RAMSES or pub/pc/MW-WINDOWS or pub/unix/RASS.

Furthermore, ModelWorks is also available via www. Please visit our home page

http://www.ito.umnw.ethz.ch/SysEcol/

Finally the Macintosh version is available on

CD «Apprentice»

at a bargain price from

Celestin Company Inc., 5652 NE Meadow Road, Kingston, Washington 98346-9505 (Phone: (360) 297-8091/Fax: (360) 297-8092; http://www.celestin.com/; For a list of products send and E-mail to info@celestin.com

How to Feedback?

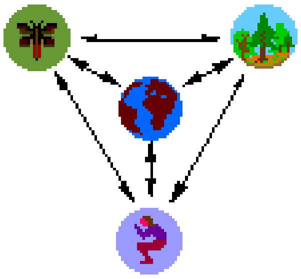
If you do use ModelWorks, please let us know. We from the systems ecology group would like to hear from you and your experiences. We welcome bug reports, but please send us bug reports only by using electronic mail. Send messages to the following internet address:

ramses@ito.umnw.ethz.ch

Attach if possible an enclosure containing the full source code of the model definition program which produced the bug. If possible, we will attempt to resolve the problem and answer your request. However, please be patient, we are only a research institution. Thank you!

For nostalgic users, our "ordinary" mail address would be:

Systems Ecology ETHZ (re: ModelWorks)
Institute of Terrestrial Ecology
Grabenstr. 3
CH-8952 **Schlieren/Zurich**
SWITZERLAND



Cited References

FISCHLIN, A., & SCHAUFELBERGER, W. 1987. Arbeitsplatzrechner im technisch-naturwissenschaftlichen Hochschulunterricht. Bulletin SEV/VSE, 78(1): 15-21.

FISCHLIN, A., MANSOUR, M.A., RIMVALL, M. & SCHAUFELBERGER, W. 1987. Simulation and computer aided control system design in engineering education. In: Troch (ed.), Simulation of Control Systems, IFAC/IMACS Proceedings, 13: 51-60.

FISCHLIN, A., 1991. Interactive Modeling and Simulation of Environmental Systems on Working Stations. In: Möller, D.P.F. & Richter, O. (eds.), Analysis of dynamic systems in medicine, biology, and ecology. Informatik-Fachberichte 275: 131-145.

FISCHLIN, A., GYALISTRAS, D. & LOEFFLER, T.J., 1995. Installation Guide and Technical Reference of the RAMSES Software (Version 2.2) For Apple® Macintosh® Computers. Technical Systems Ecology Report, Swiss Federal Institute of Technology, Zurich, 44pp.

FISCHLIN, A., GYALISTRAS, D., ROTH, O., ULRICH, M., THOENY, J. & NEMECEK, T., BUGMANN, H. & THOMMEN, F., 1994. Model Works – an interactive simulation environment for personal computers and workstations. Systems Ecology Report No. 14, 324pp.

KELLER, D., 1989. Introduction to the Dialog Machine. Report No. 5, Project Centre IDA, ETHZ, 37 pp.

WIRTH, N., 1985. Programing in Modula-2. Springer, Berlin a.o. 3rd. corr. edition. 202 pp.

WIRTH, N., 1988. Programing in Modula-2. Springer, Berlin a.o. 4th. edition. 182pp.

WIRTH, N., GUTKNECHT, J., HEIZ, W., SCHÄR, H.R., SEILER, H., VETTERLI, C. & FISCHLIN, A., 1992. MacMETH 3.2. A Fast Modula-2 Language System For the Apple Macintosh - User Manual. Departement Informatik ETH Zürich. 4th, completely revised edition, Release 3.2. 116 pp.

On the History of ModelWorks

ModelWorks has been designed originally by Markus Ulrich and Andreas Fischlin, the later versions by Andreas Fischlin, Olivier Roth, and Dimitrios Gyalistras, all from the Swiss Federal Institute of Technology Zürich in Switzerland (ETHZ). Thanks to the thorough testing and using by Thomas Nemecek many important improvements and extensions of ModelWorks became possible. Later several other Ph.D. students and many scientific collaborators, in particular Frank Thommen, Jürg Thoeny, Harald Bugmann, Daniel Perruchoud, and Thomas J. Loeffler, made partly extensive use of ModelWorks and helped to enhance the implementation and further its quality.

Together with the Dialog Machine the first versions of ModelWorks have been developed during the pilot project CELTIA (Computer Aided Explorative Learning and Teaching with Interactive Animated simulation) under the auspices of Prof. Walter Schaufelberger at the Project-Centre IDA at ETH. The Dialog Machine has been designed by Andreas Fischlin and been implemented by Andreas Fischlin, Alex Itten, Olivier Roth, and Klara Vancso during the pilot project CELTIA at the Project-Centre IDA at ETHZ.

The later versions of ModelWorks have been developed during a research project funded by two grants from the Swiss National Science Foundation and the Systems Ecology Group of the ETHZ under the direction of Andreas Fischlin. The two IBM PC implementations (under GEM respectively Windows) have been developed by Daniel Keller (Project-Centre IDA/Institute of Automatic Control Theory at ETHZ) and Thomas Wegmüller, who also made the port of ModelWorks from the Macintosh to the IBM PC several times. ModelWorks can be easily ported to every machine on which the 'Dialog Machine' is available, since it is simply a 'Dialog Machine'-program.

The following chapters of this ReadMe contain all Release Notes which have been added in the past to every release of ModelWorks. It may be useful to trace exact properties of the various versions of ModelWorks.

Release Notes for Version 3.0 May 2000

• Support for the Batch Dialog Machine added: When launching any MDP (**M**odel **D**efinition **P**rogram) written for ModelWorks in the so-called batch mode of the 'Dialog Machine', either a single simulation run or the structured experiment, if installed, is executed. Upon termination of this experiment, the MDP is automatically quit. This means, even on an interactive personal computer like a Macintosh or an IBM PC any ModelWorks MDP simulation experiment can be solved without any user interaction.

The 'Dialog Machine' 3.0 features now a batch mode, which behaves on an interactive machine exactly the same way as it does on a RASS host (RASS: **RA**mses **S**imulation **S**erver / RAMSES: **R**esearch **A**ids for **M**odeling and **S**imulation of **E**nvironmental **S**ystems). Note, in the batch mode any interactive elements of the graphical user interface are still fully managed and displayed, if the host runs an intercative 'Dialog Machine'. In contrast to the ordinary interactive mode however, any dialog is answered by the 'Dialog Machine' itself, given there is a default answer possible. Otherwise the execution comes to a standstill. The mode supports an easier development and/or debugging of MDPs, which are planned to be executed under RASS. Moreover, interactive machines can be treated like simulation servers running only in batch mode.

Release Notes for Version 2.2 January 1997 - March 1999:

- No changes on the functionality of ModelWorks were made. Each release is only due to bug fixes.

Release Notes for Version 2.2 April 1996:

- Supports now continuous-time discrete event systems (DEVS - Discrete Event System Specification) (see new optional module SimEvents from ModelWorks' client interface and the auxiliary library modules Queues, StateEvents, RandGen, RandGen0, RandGen1, StatLib, Confidence, Histograms, and StochStat which support the implementation of DEVS) (see e.g. sample model Diversity, CarPollution, or ForestYield).

- Full dynamic model and model object management, i.e. declaration and removal of models plus model objects at run-time in the middle of simulation experiments (see e.g. sample model ForestYield).

- Complex experiments like interactive parameter identification, interactive construction of phase portraits are now supported.

- Client and user interface provide same functionality.

Release Notes for Version 2.02 November 1990:

•This version is upward compatible with the previous version 1.3a from January 1990. It also uses the Dialog Machine version 2.02 and MacMETH version 2.6.2 (also compatible with V 2.6+).

•It supports now multi-level programming as for instance needed by the RAMSES shell (RAMSES stands for Research Aids for Modeling and Simulation of Environmental Systems). For instance with this new version it is now possible to call a sub-program which contains in its body another call to RunSimMaster. Previous versions of ModelWorks did not allow to call RunSimMaster from within RunSimMaster.

This offers the possibility to load respectively unload dynamically several layers of a model, e.g. sub-models, one on top of the other into the simulation environment. The RAMSES shell offers the following five sessions: Programming, Modeling, Experiment Definition, Simulation, and Post-simulation Analysis. RAMSES makes extensively use of multi-level programing. This allows for particular efficient model loading. Because of the substantial advantages, use if possible the RAMSES shell to run ModelWorks.

Release Notes for Version 2.0 May 1990:

•This version is upward compatible with the previous version 1.3a from January 1990. It also uses the Dialog Machine version 2.0 and MacMETH version 2.6+.

•It offers the new possibility declare models and model objects as well as to remove models and model objects dynamically. Thus a stage for interactive modelling is set. Note that this implies that it is now possible to call RunSimMaster from SimMaster without passing as actual parameter the procedure which declares models. Instead you may pass a procedure which installs menus and menu commands, where the latter allow to declare models and model objects. Calls to the procedures which declare model objects (DeclSV, DeclP, and DeclMV) can be preceeded by a call to the new procedure SelectModel. This procedure specifies the model to which subsequently declared model objects will belong.

Release Notes for Version 1.3a January 1990:

•Compared with version 1.3 from October 1989, this version has only minor differencies to the previous version in respect to the functionality of ModelWorks. The major difference is that ModelWorks 1.3a is now based on the Dialog Machine version 2.0 (which offers substantial enhancements compared with older versions of the Dialog Machine) and MacMETH version 2.6+.

•Many auxiliary modules have been added to the run-time library, such as calendar functions, portable random number generators, a module to read measured data into a model etc..

•ModelWorks is now distributed on two diskettes. The first contains only all documentations, the sample models, and a system folder. The system folder is only of use if you have no hard disk. The file System contains the desk accessory MockWrite, which is distributed as shareware (If you use it, please do not forget to pay the author). Print whatever you need, copy any sample model you wish into your work folder, but otherwise just archive this disk. The second disk is organized for development work. Complement it with the compiler from the MacMETH distribution disk and you have a fully functional ModelWorks. If you have a hard disk, simply copy the content of this disk into a folder on your hard disk and you are all set to go.

•Although there are no version conflicts to older ModelWorks libraries of this release, it is recommended to recompile the model definition programs in order to warrant correct functioning of models. The latter is especially important if your model definition programs use some of the newer features, which were not yet present in release 1.1.

Release Notes for Version 1.3 October 1989:

•The new version is fully compatible with previous versions but offers many new features in particular via the client interface to fully support batch simulations, e.g. you can dump the graph onto the stash file for later inspection.

•The client interface of ModelWorks Version 1.3 is fully upward compatible (same symbol files) with the previous versions. However this interface has been substantially extended (For Modula-2 programmers: Please don't ask us how we accomplished this!). This means that no ModelWorks model definition program made with ModelWorks 1.1 needs to be recompiled or changed in any way in order to work with version 1.3 (Note, this is not true for the release 1.2; to be on the safe side recompile your models made with version 1.2). ModelWorks 1.3 will work on any Macintosh computer with at least 512 KB memory (also on the new 68030 Macs like the MacIIx, cx, and SE/30), with a system version 2.0 or later and Finder version 4.1 or later, and is compatible with MacMETH V2.5.1 (needed for 68030 Macs) and the latest MacMETH V2.6+. Simply install this version (follow same procedure as described under «Release Notes for Version 1.1») and continue your work, that's it.

•The new features are too numerous and too powerful to explain here, please consult the manual. Most of them are avialable either from within the simulation environment or via the client interface (whenever meaningful, the user and the client interface offer the same functions). To mention just some of the most important new features, they are:
-Full control over curve attributes (color (here called stain), style of drawing lines, plotting symbols etc.) to associate permanently certain colors with particular variables or to hide temporarily a curve section etc.
-Full control over appearance of all windows (size, position, columns displayed etc.)
-Customization of graph: The new optional module SimGraphUtils featuring drawing of observation results independently of simulation runs but in the same graph as ModelWorks draws. Error bars can be displayed and it is also possible to draw anything you want into the graph by means of module DMWindowIO for customization of simulation results (even pictures made with MacPaint etc.).
-Installation of client monitoring.
-Installation of a consistency testing procedure. It tests before any simulations are started whether parameter settings do satisfy consistency conditions, otherwise no simulations are executed.
-Full support for the programming of experiments is now offered via the client interface. For instance, a sensitivity analysis can be easily programmed with new procedures accessing any value associated with models or model objects, i.e. so-called Get and Set procedures to change an integration method of a model, e.g. between runs, the initial value of a state variable etc. As a rule, everything you could previously control and edit interactively from within the simulation environment can now also be programmed via the client interface. This means full support for lengthy simulations (at night) no longer requiring attendance by the simulationist.
-Stash filing format is made such that a post-run analysis utility can read it for later, interactive inspection of simulation results (next morning). (This utility will be available from us on special request). Even graphs can be written onto the stash file (Implementation restriction: graphs are written in the so-called RTF-format; this requires the use of applications like WriteNow™, Microsoft® Word, or MacWrite II to open and interprete fully a stash file which contains a graph).

Release Notes for Version 1.2 April 1989:

•The new version is fully compatible with previous versions but offers the following new features in the simulation environment:

-Printing of the graph with the new menu commands "Print graph..." and "Page setup" under menu "File" (previously "Control").

-Support of transfer of the graph into the clipboard via the newly introduced menu "Edit" with the standard commands "Undo", "Cut", "Copy", and "Paste".

-To obtain more space on the menubar for the installation of additional menus by the modeller, the previous menus "Setup" and "Reset" have been merged into a single menu "Settings".

-It is now possible to enter and define so-called project descriptions, i.e. a project title, comments on the current project, and a project footer with the menu command "Project description..." under menu "Settings". The project footer is by default updated by ModelWorks and contains the date, time, and simulation run number of the current simulation session. This feature is optional and can be turned on or off. Moreover this command also allows to control the new project documentation on the stash file (see below) for the various ModelWorks object classes.

-ModelWorks will now write additional information and data onto the stash file. Besides the project descriptors, current date, time, and simulation run, it will record the global simulation parameters and all data for the models, state variables, model parameters, and monitorable variables (by default the latter is not included) before it starts to record the simulation results. This allows to store all information and specifications necessary to define a single simulation run or a whole experiment on the stash file. In particular, different from earlier versions, all results produced by an experiment consisting of several simulation runs are now fully written to the stash file.

-Windows can now be arranged in two different modes: stacked and tiled. The commands are "Tile windows" and "Stack windows" and are available under the menu "Windows". The stacked window arrangement corresponds to the start up arrangement except for the opening of the graph and table window. The latter will always be opened by the command "Stack windows". The command "Tile windows" will rearrange the windows so that they do no longer overlap and make optimal use of the used computer screen. This implies a deactivation of several columns shown in the stacked state; just the most important columns are shown, i.e. for all objects just the identifiers and the current values: for models the integration method, for state variables the initial values, for parameters their values, and for monitoring variables the currently activated monitoring. The actual tile arrangement varies from screen to screen. Neither tiling nor stacking of the windows will affect the current values of the model objects (different from menu command "Reset all" in previous versions, which was a mean to rearrange the windows to the original start-up arrangement, but which also implied a resetting of all object values).

•The client interface of ModelWorks Version 1.2 is fully upward compatible (same symbol files) with the previous version 1.1, i.e. no ModelWorks model definition program made with ModelWorks 1.1 needs to be recompiled or changed in any way in order to work with version 1.2. ModelWorks 1.2 will work on any Macintosh computer with at least 512 KB memory (also on the new 68030 Macs like the MacIIx, cx, and SE/30), with a system version 2.0 or later and Finder version 4.1 or later, and is compatible with MacMETH V2.5.1 (needed for 68030 Macs) and the latest MacMETH V2.6. Simply install this version (follow same procedure as described under «Release Notes for Version 1.1») and continue your work, that's it.

Release Notes for Version 1.1 October 1988:

•The version is based on the version 1.0 of the Dialog Machine (same as ModelWorks V1.0) but uses the new MacMETH 2.5. This new version of ModelWorks offers several new features, among others:

-an additional Runge-Kutte integration method with variable step length
-variable coincidence interval length
-a new module called TabFunc for functions only defined by a set of data pairs to be linearly interpolated during simulation (replaces module Tables released with ModelWorks 1.0). It may be used to declare table functions, to interpolate values with a particular method of extrapolation or no extrapolation whatsoever, plus an interactive editor to change any of the values defining the function from within the simulation environment (features a graph of the table function).

•ModelWorks Version 1.1 is fully upward compatible with the previous version 1.0, i.e. no ModelWorks model definition program made with ModelWorks 1.0 needs to be recompiled or changed in any way in order to work with version 1.1. Simply install this version and continue your work, that's it.

•In order to run the ModelWorks sample model Logistic as described in the ModelWorks Tutorial, you don't have to install first ModelWorks; simply run the RMSMacMETH shell contained on the disk on which you found this document (double-click or open the application RMSMacMETH as any other Macintosh application). Then choose the command Execute under the menu File and open in the folder Sample Models the file Logistic.OBM.

•To fully install ModelWorks follow one of the procedures described below. Depending whether you have additional disk space available, such as a hard disk, follow procedure A otherwise (only one 800K Byte floppy disk drive available) follow procedure B:

Procedure A (Hard disk system):

Step 1)Copy the content of the ModelWorks distribution diskette fully onto your Hard disk into a new folder you could name something like RAMSES.

STORE the original ModelWorks disk in a safe place! Make sure the write protection latch on the distribution diskette is locked (open whole). Never work on your original distribution diskette.

Step 2)In case you should not have the Desk Accessory MockWrite in your System on the hard disk included (check under your Apple menu) use the Font/DA Mover to move it from the System contained on the ModelWorks disk into your System on your hard disk.

Step 3)Destroy the System Folder contained in the newly created folder RAMSES on your hard disk by dragging it into the trash can.

Step 4)If you wish to develop your own models you need also the MacMETH compiler. This compiler is contained on the MacMETH distribution diskette, which you have to obtain separately (sorry!) from the following address:

Institut für Informatik ETH Zürich
ETH-Zentrum
CH-8092 Zürich
Switzerland

Do copy the file Compile contained in the folder SysProg on the MacMETH distribution disk into the folder Tools within your folder RAMSES. Make sure you copy only the file Compile and no other file into your RAMSES folder. Check also that the MacMETH version is 2.5 or later!

Procedure B (Floppy disk only system):

Step 1)Make first a working copy of the original ModelWorks distribution disk.

STORE the original ModelWorks disk in a safe place! Make sure the write protection latch on the distribution diskette is locked (open whole). Never work on your original distribution diskette.

Step 2)Drag all files contained in the folder Sample Models except for GrassCows.MOD, Logistic.MOD, and Logistic.OBM from your ModelWorks working disk into the trash-can. In case you would need any of these files at some later time, for instance as a template to start writing a new model, you will have to copy that file from your original disk onto your working disk again.

Step 3)If you wish to develop your own models you need also the compiler from MacMETH. This compiler is contained on the MacMETH distribution diskette, which you have to obtain separately (sorry!) from the following address:

Institut für Informatik ETH Zürich
ETH-Zentrum
CH-8092 Zürich
Switzerland

Do copy the file Compile contained in the folder SysProgs on the MacMETH distribution disk into the folder Tools on your ModelWorks working disk. Make sure you copy only the file Compile and no other file onto your working disk. Check also that the MacMETH version is 2.5 or later!

Note, there will not be much working space left on your working disk. This will require you to remove older files first from your working disk before you can start with a new project. Otherwise there is no way than to add some more disk space onto your system. In the latter case don't forget to modify the User.Profile file accordingly (Instructions on how to do it can be found in the MacMETH manual and in the ModelWorks Manual Appendix).

Procedure A as well as B:

Step F)This step is common to both installation procedures described above. Depending on your preferences it is possible to install ModelWorks with a full run-time system or only with a smaller portion of it (in case of memory limitations, i.e. if your Mac should have less than 1 MByte RAM) and it is also possible to switch between a MathLib which uses SANE (Standard Apple Numerics Environment) or not.   Note these installations are not final; they can be reverted any time you wish.

- The folder RAMSESLib contains the following modules:

SimLibS.OBMCurrent default version of basic integration routines; on original disk identical to the file SimLibS.OBM (no Aux) [see below]
SimLibS.OBM (no Aux)Version of basic integration routines which will NOT call SimLibAux0.OBM [see below]; results in a ModelWorks requiring little memory
SimLibS.OBM (Aux0)Version of basic integration routines which will call SimLibAux0.OBM [see below]; results in a ModelWorks requiring more memory
SimLibAux0.OBMcontains a variable step- length integration method

You may now simply choose your preferred version of ModelWorks (large or small) by removing first SimLibS.OBM (drag to trash), second duplicate either SimLibS.OBM (Aux0) or SimLibS.OBM (no Aux) depending on your preference, and finally renaming the new copy to SimLibS.OBM. That's it. All the rest will be taken care of by ModelWorks automatically. For instance the dialog box which lets you select an integration method will look differently accordingly to your current installation, i.e. it will you let choose from additional integration methods if you have decided for the larger version.

• The folder M2 MiniLib contains the following modules:

MathLib.OBMcontains fast routines which do not use SANE (useful on Macs without a mathematical coprocessor). Is the default.
MathLib.OBM(SANE)contains slow but precise routines which do use SANE

Depending on your preference you may choose between a fast but less precise (does not use SANE) or slow but precise (uses SANE) method to compute the MathLib functions like Ln, Sin, Cos etc. In order to work with SANE or without follow these steps: Rename first MathLib.OBM (the version present on the original disk) to MathLib.OBM(NOSANE) and then rename the file MathLib.OBM(SANE) to MathLib.OBM. That's it. From now on all computations using MathLib routines will be made either with SANE or without SANE. You may any time switch back and forth between the two versions by renaming the file you wish to use to MathLib.OBM., thats the one ModelWorks will be using. In case you should get confused which file is which, read the Information on the files by selecting the Get Info command under the Finder menu File.

That's it, you are now ready to work through the tutorial and you will be able to develop your own models with ModelWorks. Have fun!

•The module TabFunc is optional and will be linked to your model program only if you use it. The following ModelWorks model definition program uses this new model:

The growth rate R in the model is not a constant model parameter but varies with the temperature T and is expressed as a function of T. The function R(T) is only determined by some discrete measurements which are contained as data pairs in this table:

```
T [°C]R(T) [/h]
0.00.00
5.00.04
10.00.07
20.00.17
30.00.19
40.00.26
50.00.25
```

T is a sinusoidal forcing function of the form:

```
        T(t) = Tm + Ta*Sin(6.2832*t/24.0)
```

where Tm is the mean temperature and Ta is the amplitude within the temperature fluctuates during its diurnal cycle. The model equation (continuous-time):

```
        dx(t)/dt = R(T(t))*x(t)
```

The following statements contained in the procedure which declares the model will install the function R(T(t)):

```
VAR
  RTt: TabFunc;  (* the table function R(T(t)) *)
  tempVect, grVect: ARRAY [0..7-1] OF REAL;
   (* used to pass the data to the function *)
...
tempVect[0]:= 0.0;  grVect[0]:= 0.0;
tempVect[1]:= 5.0;  grVect[1]:= 0.04;
tempVect[2]:= 10.0;  grVect[2]:= 0.07;
tempVect[30]:= 20.0;  grVect[3]:= 0.17;
tempVect[4]:= 30.0;  grVect[4]:= 0.19;
tempVect[5]:= 40.0;  grVect[5]:= 0.26;
tempVect[6]:= 50.0;  grVect[6]:= 0.25;
DeclTabF(RTt, tempVect, grVect, 7,  TRUE, "RTt: growth rate vs.
temperature", "Temperature", "Growth rate", "°C", "/h", 0.0, 60.0, 0.0, 1.0);
```

This will allow to model the differential equation with the following statement in procedure Dynamic:

```
temp:= tm + ta*Sin(6.2832*CurrentTime()/24.0);
xDot:= Yie(RTt,temp)*x;    (* interpolates growth rate for any temp *)
```

While running this model you may edit the function RTt as you can modify a value of any ordinary model parameter. E.g. you may want to change the last data pair [50.0, 0.25] to temp = 60.0 °C and a growth rate gr = 0.1. This can be easily acchieved by means of the menu command Edit under the added ModelWorks menu TableFunctions just beside the Simulation menu. Try it to see it.

•All known bugs have been fixed.

Release Notes for Version 1.0 April 1988:

•The version has been based on the new version 1.0 of the Dialog Machine. This has certain consequences:

-Zoom boxes are available for all windows
-Module DMFiles supports now fully the HFS (Hierarchical file system). In particular, searching for already existing files is done according to the paths given within the MacMETH User.profile and if the file can be found in any of the folders, its path is available in the returned file name
-It is now possible to use module DMEditFields featuring modeless dialogs
-Module DM2DGraphs is distributed too. You may use it to plot your own charts of the simulation results ModelWorks computes
-There are less bugs within the new version of the Dialog Machine
-However, the new version works best with the new Finder and Systems files and therefore requires more memory and disk space, and finally
-The Macintosh with only 64K ROMs are no longer supported.

•If only discrete time models are installed, the global simulation parameters (simulation time begin and end, integration step and monitoring interval) are all rounded to the nearest integer and it is no longer possible to enter fractional values (either via the user or the client interface).

•All known bugs have been fixed, in particular the procedure SimBase.SetIntegrationStep functions now properly and the integration is performed correctly if only discrete time models are installed and the monitoring interval is different from 1.

Release Notes for Version 0.6 Januar 1988:

•The user interface has been changed by altering existing and introducing new menu commands. Simulations on the Macintosh II with a color monitor are now supported.

•The client interface has been modified by introducing the following identifiers:

```
SimMaster.SimRun
SimBase.DeclExperiment
```

These two new procedures exported both by module SimBase and SimMaster respectively; they can be used to define any kind of experiment calling any times a whole simulation run. The client (modeller) has to install into ModelWorks an experiment procedure by calling DeclExperiment. This experiment can be executed once ModelWorks is active by the user by selecting the menu command "Execute Experiment" under menu "Simulation". This feature may be used for instance to construct a phase portrait by changing the initial conditions of state variables during the experiment or to identify model parameters while computing a performance index based on a simulation run.

The experiment procedure can be freely programmed and contains typically calls to the procedure SimRun. The following example shows an experiment used to run several simulations while changing the model parameter p1 from run to run.

```
CONST maxRuns = 3; VAR runNo: [1..maxRuns];


PROCEDURE MyExperiment;
BEGIN
  FOR runNo:= 1 TO maxRuns DO SimRun END; runNo:= 0;
END MyExperiment;
```

Whereby the procedure to initialize a run may have been programmed as follows:

```
PROCEDURE Init;
BEGIN
  CASE runNo OF
   1: p1:= 0.0;
 | 2: p1:= 1.5;
  | 3: p1:= 3.0;
 ELSE
    p1:= 3.14;
  END (*CASE*);
END Init;
```

Calling SimRun results in the same action as if the user would have selected the menu command "Run simulation" under menu "Simulation". All the program modes apply as well to an experiment as to a simple simulation run; however, note that the experiment may only be stopped during a simulation run, not during another lengthy modeller defined computation, such as a parameter identifcation.