*Journal of Universal Computer Science, vol. 13, no. 7 (2007), 1012-1021*
*submitted: 7/3/07, accepted: 25/7/07, appeared: 28/7/07 © J.UCS*

# HME: a Handheld Model Editor for Educational Contexts

**Maria José Marcelino**
(Universidade de Coimbra, Portugal
zemar@dei.uc.pt)

**Abstract:** Handheld devices are becoming more and more popular in education. Educational simulation and modelling are not new soil, but for handhelds they are still much under explored. Due to the difficulties teachers usually face in developing computer models and simulations and the lack of adequate tools for building them, we developed an authoring-tool for handheld educational simulation and modelling, called Sim-H (SIMulation for Handhelds). Sim-H is made by several modules each one relating to a type of simulation application that can be used in an educational context. One of these modules, that we describe thoroughly herein, is the Handheld Model Editor, a modelling tool for handheld devices that can be used to build models to use as such or as the core of educational handheld simulations.

**Keywords:** handheld devices, modelling, simulation, authoring-tools
**Categories:** K.3

## 1 Introduction

The next technology that will affect education and learning is mobile and handheld technology [Lockitt 2005]. Handheld computers, also called PDA (Personal Digital Assistant) or Palmtop computers, are small and light machines, for personal use, capable of performing organization, productivity and communication tasks. In the last years, they have gained increasingly functionalities (digital photo and video cameras, sound recorders, e-mail, etc.) and adepts. Some important advantages of handheld devices are their price (they are cheaper than laptops), storage capacity (they allow saving and transferring increasing quantities of information), flexibility (they can be used without a physical connection, thus favouring mobility), connectivity (they allow direct access to wireless networks), ubiquity (they have the possibility of anytime, anywhere access) [Lockitt 2005, Savill-Smith and Kent 2003].

Software for handhelds is becoming rather common too. We already find in the market several examples of administrative, collaborative, communication and educational applications [Naismith et al. 2005]. In this paper we are particular interested in this last category, where we can find text editors, graphical editors, simulations, educational games, etc.

Simulation and modelling have been used in education and recognized as valuable for quite a long time. From simple closed programs that represent a specific situation (known as an educational simulation program) to authoring-tools that assist the user in building them or modelling tools that allow a student/teacher to create/use a model (or representation) of a real or imaginary system and play with it, we find

several examples and studies in the literature [Aldrich 2005, Brigas and Marcelino 2004, Robinson 2004].

Many simulation programs or tools that exist are for a specific subject area or age group, while others are quite general. There are tools for desktop computers, as well as for the Web, but a few for handheld devices, where we have clearly a deficit.

Next we review some of the current efforts in this chapter [see Section 2]. After [Section 3], we describe briefly the Sim-H system, or SIMulation for Handhelds, an authoring system for handheld modelling and simulation. Then [Section 4], the Handheld Model Editor, one of Sim-H tools, main characteristics are overviewed, along with an illustrative example. We end the paper with some conclusions and future work perspectives.

## 2 Handheld modelling and simulation

In handheld modelling and simulation we find mainly three approaches:

- Simulations or simulation programs, where, through a model, it is possible to interact and represent the evolution of a particular situation in a dynamic way,
- Modelling tools where the user can build and test his/her own models and
- Simulation tools where a simulation can be constructed and run, giving the possibility of interacting with its parameters [Soloway et al. 2001, Roschelle 2003, GoKnow 2006, Geney 2003, ParticipatorySimulations 2005].

Among simulations for handhelds we can mention Cooties Game [GoKnow 2006], Geney [Geney 2003] and the Live Long and Prosper Game [ParticipatorySimulations 2005]. But we could refer many others.

Cooties Game is a simulation where students study cooperatively the spread of a disease. The teacher defines a set of simulation parameters, the students 'infect' themselves using their handhelds and can study after the propagation of the disease.

Geney was developed to help students explore genetic concepts in a game like environment. It simulates a fish population, distributed by several handhelds. Fishes develop at a constant rate and can mate in this 'lake'. The new fishes can be exchanged among handhelds and the game objective is the cooperative work between students with the aim of producing fishes with pre-determined characteristics.

Live Long and Prosper Game is also for genetic concepts and shares some similarities with Geney. It is also a game where these concepts must be understood and applied in order to 'survive'.

Although we can build these simulations using a general purpose programming language, this approach is quite hard and time-consuming for a common teacher. A better one is using a specific tool to support him/her in this role, a simulation authoring-tool. On the other hand, if we want to use models in education a suitable approach is to have a tool to express these models straightforwardly or what is called a modelling tool.

Modelling and simulation tools in this sense for handheld devices are scarce. We can mention Pocket Model-It, from Hi-CE (Centre for Highly Interactive Computing,

University of Michigan), in the first category and C5, from Agentsheets, Inc., in the second, although it does not involve only handheld devices.

Pocket Model-It is the handheld version of Model-It, a modelling visual tool for qualitative models for desktop computers, and was developed after it. The first phase of model development is planning. To build a model it is necessary to identify objects, variables and relations among them. After, a model can be run. Model results are presented in graphical form for better visualization [See Fig. 1].

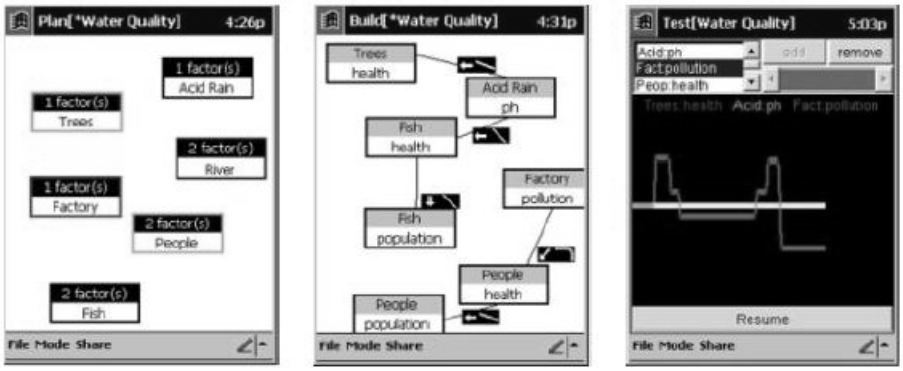


*Figure 1: Pocket Model-It, model plan, execution and test.*

C5 (Distributed Simulations for Handhelds), a research project aimed at several educational levels, proposes a new architecture, the C5 (compact, connected, continuous, customizable, collective simulations), that uses handhelds, desktop computers, and Internet to support students' work. Trough a simulation server, it is possible to simulate a part of a model with a handheld device and after compare results, reflect and analyze them in group.

We could also mention AgentSheets, a product of AgentSheets, Inc., in this category, although it is not a tool for handhelds, but as we can create in it an application that can after be transferred to a handheld device (PDA, phones, etc.), a Web page or a desktop computer, we can say that it supports, at least partially, the handheld paradigm. In AgentSheets a model is represented by a grid of agents or objects with behaviours that can be programmed by the user. We can build with this tool little simulations and interactive games in a graphical and intuitive way. However, to model some agents' behaviours the task it is not as easy as it may seem as the user has to know quite deeply the tool embedded 'programming language', that is he/she must have some technical skills.

## 3 Sim-H modelling and simulation authoring-tools

We have analyzed the tools mentioned above and found that some important characteristics for a modelling/simulation authoring-tool for handheld devices were still missing. We can refer, for instance, that existing tools are only for simulation (C5) or for modelling (Pocket Model-It) and not for both, in others the tool

environment is not for handheld devices (AgentSheets), others require complex infrastructures (C5), or the user has to have basic knowledge of the built-in programming language to use them properly (AgentSheets), etc.

As a consequence a set of key requisites for such an authoring-tool have been identified. Among them we mention:

- To allow building several types of models (continuous, discrete, qualitative, etc.) in a transparent way.
- To allow the construction of simulations based on these models.
- To have an easy to use metaphor based on graphical objects (images, icons, etc.).
- To interact mainly using direct manipulation techniques.
- To integrate easily multimedia elements, like sound, photos, etc., in simulations.
- To integrate easily animations of several types in simulations.
- To be easy and quickly to make changes.
- To be portable to similar devices and among languages and cultures.
- To make it possible to be used individually or collaboratively.
- To make it possible to be used "anyway anywhere", in the classroom, at home, in the lab, etc.
- To be cheap [Ferrão and Marcelino 2005].

Based on them we are developing a prototype of a tool to support handheld modelling and simulation that we called Sim-H, or SIMulation for Handhelds. This tool is made by several editors for in a simple, effective and efficient way build handheld models and simulations. One of its components is precisely a modelling tool that will be described in detail in the rest of this paper, the Handheld Model Editor (HME). Another one is the Handheld Scenario Editor (HSE), a tool to build scenarios or interfaces for these models. For some learners the direct interaction with a model can be rather hard. What they need is a simpler way to do it. This is exactly what the HSE scenarios allow. Yet another tool is the Handheld Learning Editor (HLE), a tool for developing learning layers to aggregate to models or scenarios and build simulations. A learning layer can include mainly content information, package objectives, operational rules, forms of interaction with the underlying model, guidance, learner activities, adequate feedback to students' actions, help.

## 4 The Handheld Model Editor

A modelling tool for handheld devices should possess, among others, the following characteristics:

- to allow the representation of several types of models (continuous, discrete, etc.),
- to permit the introduction of the model in several ways (like graphical and textual forms),
- to have a graphical interface,

- to be easy to use and make changes,
- to allow the re-use of models in different perspectives or situations, etc.

HME is a modelling tool that allows implementing continuous models of generic systems, those described by differential equations. However, we plan to extend it to discrete models soon (those described by events). HME has a graphical interface that allows a simple and friendly interaction, as the description of a model is done by direct manipulation techniques and using an iconic diagram that represents the model entities and the relationships among them. This has clear advantages, although a model can be input also by typing directly its equations.

To build a model the user will have to identify the situation to be modelled, the model objectives, the entities that make the model, their relationships, the model initial conditions, the calculus method and the forms of visualizing model results [Brigas and Marcelino 2004].

In HME a continuous model is represented by a diagram of entities that represent model state variables, constants, derivatives, etc. The relationships among the model elements are made by connecting them with links [Fig. 2].

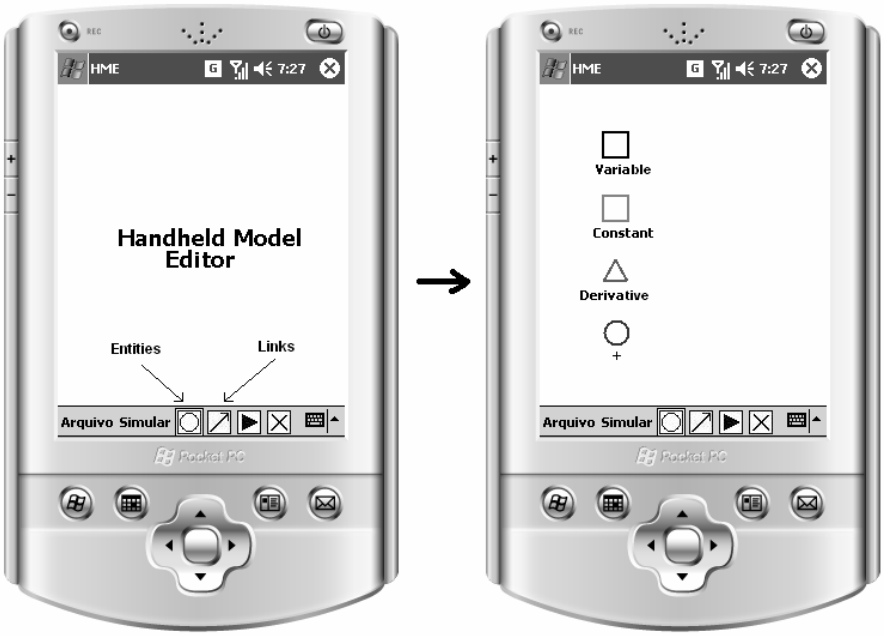


*Figure 2: HME, Handheld Model Editor initial screen and types of model entities.*

When building a model in HME the user has the possibility of inserting entities of several types, namely variable, constant, derivative and junction point.

To create a state variable in a model the user has just to select the Entities icon in the toolbar [see Fig. 2] and point in the white canvas. A symbol of a small square

appears. For all state variables created we can change any of their properties: name (nome), initial value (Valor inicial) and applying it a function (Função) [see Fig. 3].

*Figure 3: Changing the properties of a state variable in HME.*

Model constants are represented with a similar icon as state variables, but with a different colour – while variables are represented in black, constants are represented in gray to distinguish them easily. To create a constant we just need to select the Constante button in the variable properties dialog box [see Fig. 3].

Model derivatives are created selecting the Derivadas option in the properties dialog box and they are represented through a green triangle. The teacher or the student has the possibility of changing its name and initial value in a way similar to state variables and constants.

Junction points are entities that represent arithmetic operations (a sum, a subtraction, a multiplication or a division) applied to their input entities. They let us relate state variables, constants and derivatives to each other. To create one we choose the Junção option in the dialog box.

These relationships of the several entities that make a model are specified also through connections or links. For that the user has to select the Links icon in the toolbar and point in the canvas to the entities he/she wants to link.

Entities can be deleted easily by selecting the toolbar X icon. Of course all the related links will be deleted too, to maintain model consistency.

Pointing to the toolbar ► icon a model will be calculated or run. The result can be seen in tabular form [see Fig. 4] or in graphical form [see Fig. 5].

The calculation of a continuous model is done basically using numeric integration. Among numeric integration methods we have the 1$^{st}$ order Euler method

(the simplest one), several Runge-Kutta methods, etc. In HME the default integration method is this Euler method, although the tool allows calculating the models using two more methods, namely a 2nd and a 4th order Runge-Kutta methods. Certain models can give numerical instability with the Euler method. In these cases it can be important to have more accuracy and that is precisely what Runge-Kutta methods give.

In most models the independent variable is the time, but we can have in HME any other variable specified by the user. The model independent variable can be altered very easily, along with its initial and final values and integration step.

In the next section we will clarify all these concepts with an example.

## 5 Using the Handheld Model Editor

One of the most widely known models of coexistence of predator-prey species in the same territory is the following model that was created by Lotka-Volterra [Evans and Findley 1999]:

$$\begin{cases} \dfrac{dR}{dt} = aR - bRF \\ \dfrac{dF}{dt} = -cF + dRF \end{cases} \quad (\mathbf{1})$$

Where:

| | |
|---|---|
| *dR/dt* | Prey population at instant *t*; |
| *dF/dt* | Predator population at instant *t*; |
| *a* | Rate of natural growth of preys in the absence of predators; |
| *b* | Rate of decrease of preys due to the presence of predators; |
| *c* | Rate of natural decrease of predators if there is not preys; |
| *d* | Constant of the efficiency with which predators transform encounters with preys into food (well-succeeded hunts). |

This model allows a student to analyse in what way the growth of one population species interferes with the growth of the other. The system initial conditions are:

- Only one species of preys exists that serves as food for the population of predators;
- Only one predator species exists that hunts the population of preys;
- Other data that could alter model evolution (like prey food, diseases, man intervention, etc.) are not considered in the model.

Using HME the model can be represented in the following way [see Fig. 4]:

- Constants a, b, c and d will be represented by the constant symbol.
- State variables R and F are symbolized through state variable entities.
- Derivatives DR/dt and DF/dt will be represented by the derivative icon.

- Besides these, junction points are necessary to describe the equations relationships, namely the multiplications and subtractions.

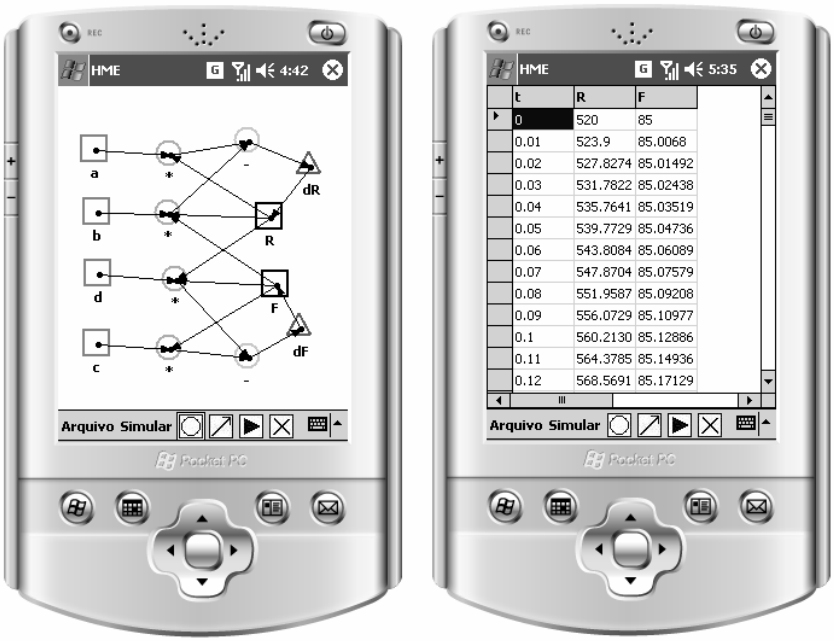


*Figure 4: Lotka-Volterra model in HME with model results presented in tabular form.*

After constructing the model and assigning the values of constants and the initial values of state variables it is necessary to choose the calculus method (by default it is the Euler method) and to define values to the independent variable (in this case the time, t). Finally the user can verify the behaviour of the model in a table [see Fig. 4] or in a graph. [see Fig. 5]. As it can be observed the typical sine waves of this model behaviour are plotted.

With this model several activities can be planned with the students. For instances, we can use it to study the impact of the raise in the rate of natural growth of preys or in the rate of natural growth of predators in both populations. Students would have to vary it systematically and watch and register the evolution of the two species. This corresponds to using the model in a so-called exploratory way. But we can program others forms, like asking students to extend this model to accommodate other factors, as the presence of another population in the same territory or the inclusion of food for preys. That would correspond to an expressive way of model utilization. This is usually a richer approach, but harder too as one has to devise his/her own model or representation of reality. It asks for capacities like abstraction, generalization, critical thinking, transfer, etc., which are not easy to exercise for anyone, but that are essential to learning.

Of course we can use other models to support the study of the same topic. As a model can be built in a few minutes, it is quite easy and straightforward to do it with HME.

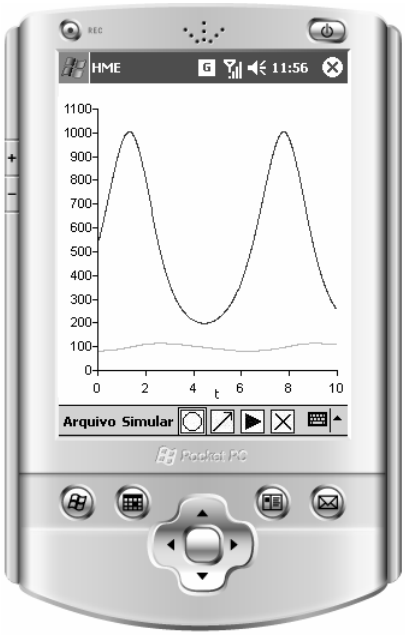


*Figure 5: Lotka-Volterra model in HME with model results shown in graphical form.*

## 6 Conclusion

With the spread of handheld devices, it is reasonable to believe that more software for these devices will appear. We find already several commercial programs for many areas (Linguistics, Engineering, Medicine, Social Sciences, Physics, Geography, etc.), but little that allow us to create models or simulations.

In this perspective, we presented, in this paper, an authoring-tool for handheld modelling and simulation, Sim-H. This tool consists of several modules, one of which is the Handheld Model Editor (HME). With HME the teacher or the student can create and run a model that represents a system without having any knowledge of programming languages or mathematical formalisms behind. Through the Sim-H tools we intend to create an environment that will allow the creation of models and simulations by the teacher or by the learner in a simple and useful way. We already have tested the usability of HME with a small group of potential users. As the results were positive, in a near future, we plan to use it to support a few students' activities in the scope of an Informatics Engineering simulation course.

In the meantime Sim-H development will continue. At present a first prototype of the Handheld Scenario Editor (HSE) exits and the Handheld Learning Editor (HLE) is

in a final stage of development. Also the design and implementation of another tool to support evaluation while using a model/simulation will take place soon.

## References

[Aldrich 2005] Aldrich, C.: "Learning by Doing: A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in e-Learning and Other Educational Experiences", John Wiley & Sons, Inc. (2005).

[Brigas and Marcelino 2004] Brigas, C., Marcelino, M.: "SimulWeb: Web-based Modelling and Simulation Authoring-tools", Proc. Ed-MEDIA'04, AACE Publishing, Switzerland (2004), 801-807.

[Evans and Findley 1999] Evans, C. and Findley, G.: "Analytic solutions to a family of Lotka-Volterra related differential equations", Journal of Mathematical Chemistry, 25, 2 (1999), 181-189.

[Ferrão and Marcelino 2005] Ferrão, S. and Marcelino, M.: "Modelar e Simular na Palma da Mão?", Proc. 5º Simpósio Internacional de Informática Educativa, Portugal (2005).

[Geney 2003] Geney, What-If, 2003, http://guir.berkeley.edu/geney.

[GoKnow 2006] GoKnow – Powerful Learning Tools for Today's Digital Kids, University of Michigan, 2006, http://www.goknow.com/Products.

[Lockitt 2005] Lockitt, B.: "Mobile Learning", 3T Productions Limited (2005) 3.

[Naismith et al. 2005] Naismith, L. et al.: "Learning with Mobile Technologies", NESTA Futurelab Report 11 (2005).

[ParticipatorySimulations 2005] PDA Participatory Simulations@MIT, 2005, http://education.mit.edu/pda/index.htm.

[Robinson 2004] Robinson S.: "Simulation: The Practice of Model Development and Use", Warwick Business School (2004), 1-8.

[Roschelle 2003] Roschelle, J.: "Unlocking the learning value of wireless mobile devices", Journal of Computer Assisted Learning, 19, 3 (2003), 260-272.

[Savill-Smith and Kent 2003] Savill-Smith, C. and Kent, F.: "The Use of Palmtops computers for Learning". Learning and Skills Development Agency (2003) 4.

[Soloway et al. 2001] Soloway, E. et al.: "Log on education: handheld devices are ready-at-hand", Communications of the ACM, 44, 6 (2001), 15-20.