

The monist project

A system for learning and teaching with educational simulations

Sören Lorenz, Wolfram Horstmann, Markus Oesker, Hubert Gorczytza, Andreas Dieckmann, Martin Egelhaaf
Department of Neurobiology, Faculty of Biology, Bielefeld University, Germany
soeren.lorenz@uni-bielefeld.de

Abstract: The monist-project has committed itself to the task of developing recipes for the use of simulations in higher education in neural and cognitive science. The basis of the monist-project is a software package, the monist console, designed for providing compact, ready to use educational simulations with variable application context. The monist console is a client-server application supporting online and offline learning with simulations. Simulations become educational as they are complemented by topical texts, instructional design, concrete tasks and an editor for individual notes and solutions. This ‘format’ is integrated in the monist-console. Educational simulations can be used in nearly every learning context found in higher education. They can selectively supplement lectures and tutorials or can be compiled to complete courses. Online file-management and communication tools as well as an learning management (via monist-server) help to organize personalized courses.

1. Introduction

In recent years, there has been great scientific benefit in explaining brain function by model simulation [1, 2]. In higher education, simulations are introduced only sporadically – ‘recipes’ for applying simulations in education are widely missing. The monist-project has committed itself to the task of supporting the use of model simulations in higher education. Primary goals are to improve the design for educational simulations and to develop an appropriate environment for learning with and authoring of this type of educational simulations. This paper describes the system design, its modules and contents in detail and demonstrates their possible application in higher education.

Though the main topics are neural and cognitive sciences, the system is designed for the integration of educational

(RPC) and is connected to a MySQL-database both, locally and on the server. The latter assures online *and* offline learning with simulations. That comply the demand of learners and teachers to use the functionality offline as well.

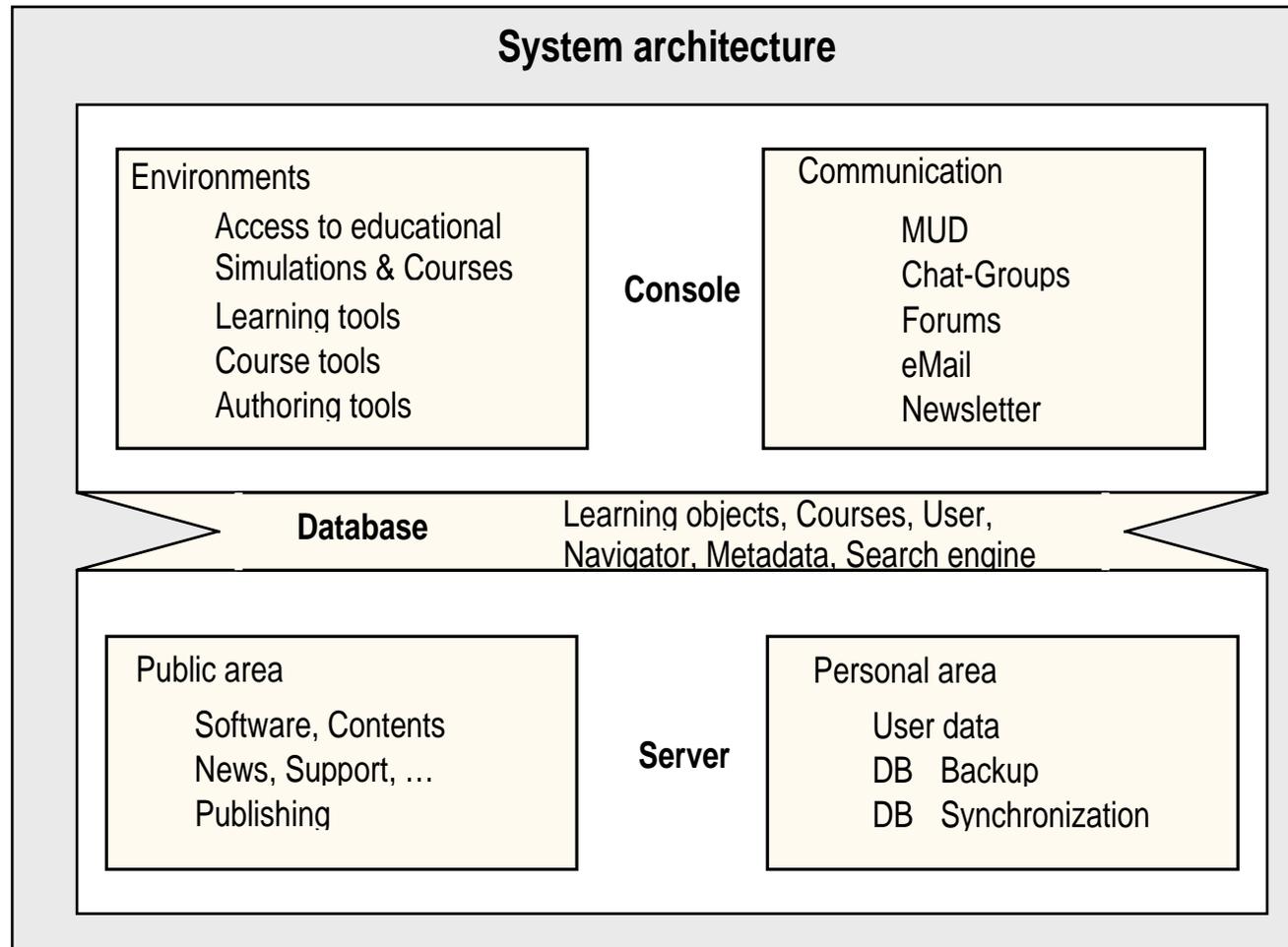


Fig.1: Architecture of the monist system. The figure illustrates the main modules of the monist-console.

There are mainly three parts, the console, the server and the database (locally and on the server) with its access tools. In combination with the database, the console provides access to all contents of the monist-system.

The server provides general tools, a public area and the personal area for every user. For details of functionality see section 2.

The monist-console consists of several modules, which are interconnected (see figure 1). The essential items of the system are compact, ready to use *educational simulations* with variable application context (see section 3 for detail). They can selectively supplement lectures and tutorials or can be compiled to complete courses. A *navigator* provides tree-structured access to all contents stored in the *monist-database*, e.g. courses, educational simulations, general

RPC-mechanism which allows communication between different monist-applications. The talker can provide a virtual world with rooms that users can explore and use to communicate. The talker unites both basic features of typical synchronous (chats, messaging system) and asynchronous (email, news) communication-media. The communication functionality can be accessed either by command-line or by graphical user-interface (GUI) for a more intuitive approach to the basic features of common communication tools.

The screenshot shows the 'Monist Konsole' application window. The title bar includes 'Navigator', 'Edit', 'Options', and 'Help'. The main interface is divided into several sections:

- Top Navigation:** A series of buttons for navigating through content: 'Dynamics of', 'Coding descr', 'Images', 'Monist', and 'Coding descr'.
- Left Panel (Course Tree):** A hierarchical tree structure under the 'Monist' root. It includes folders for 'Courses' (with sub-items like 'Introduction to the theo...', 'Basic Neural Simulations', 'Systems and Signals', 'workspace', 'Introduction to the theo...', 'Signals and Systems', 'Measurement and Model', 'Basic Neural Simulations', 'Introduction', 'Passive Membrane: Restin', 'Passive Membrane: Dynan', 'Active Membrane: Spikes', 'Coding', 'Synapse', 'Networks', 'RC-Circuit') and 'Units' (with sub-items like 'Neuron', 'Networks', 'Sensory Systems', 'Motor Systems', 'Behaviour', 'Learning', 'Cognition', 'Methods', 'Glossary').
- Right Panel (Presentation Slide):**
 - Title:** "Characteristics": the basic relation between input and output signal
 - Text 1:** Consider a photoreceptor that receives a light stimulus. When the signal of the analyzed in an experiment, e.g. by registering electric changes at the phtotreceptor membrane, we observe the signals on an oscilloscope.
 - Text 2:** It is possible to conclude from the receptor response what the stimulus looked like?
 - Text 3:** Given, the photoreceptor was in a steady (adapted) state this should be possible to a certain degree: usually we assume that a higher stimulus amplitudes -- e.g. a brighter light flash will cause higher response amplitudes -- e.g. a greater depolarization. However, this is not the only way a signal can be encoded in another signal -- there are many more ...
 - Diagram:** A complex diagram illustrating the relationship between input and output signals. It features:
 - Eingangssignal Stufe:** A section with various waveform icons and a slider for 'Amplitude' ranging from 'min' to 'max'.
 - Aktivierungs-funktion Logistisch:** A section with a sigmoidal curve icon and a slider for 'Steigung' (slope) ranging from 0 to 'max'.
 - Graph:** A plot of 'Antwort f(t)' vs. 't' showing a sigmoidal curve. The y-axis ranges from 0 to 1, and the x-axis ranges from 0 to 1.
 - Oscilloscope:** A digital oscilloscope showing a step function (input) and a smoothed, sigmoidal response (output). A binary sequence '<10110' is displayed next to it.
 - Control Panel:** A vertical column of icons for simulation control, including a yellow 'NW' icon, a red 'STOP' icon, and various signal and measurement tools.

commentaries. Additionally, students are not restricted to an actual course, but have the opportunity to view all offered contents for self-studying or exam preparation.

The *teaching environment* is equipped with additional functionality. Teachers can organize their courses and course members. They have access to a learning management. All uploaded task solutions can be viewed as well as comments can be sent via the communication system. Teachers have access to all educational content provided by the monist-database. They can use precompiled courses, organize personalized courses, or mix up several existing materials to a new course via the navigator. And if the current collection is not sufficient, they can produce their own material by using the content editor (see below). Using the communication system during an online-course, teachers can support learning or administrate via chat, mail and news. Additionally, they can create course rooms and newsgroups in the talker.

To support an easy production process for educational simulations, an *authoring environment* is provided for designing and modifying educational material and complete courses. It consists of the content editor, supporting the format for educational simulations as described above. Authors can write topical texts, design instructions and tasks. Existing glossary and literature entries can be linked or new entries added. Any type of media can be integrated into the unit, including different simulation formats. Already existing simulations, like applets, programs and the like, can easily be embedded in monist learning units via external links.

An integrated *simulation construction kit* is under development. It will also allow for designing interactive elements beyond the simulation itself, like visualizations or data analysis tools. Any new object can (and should) be equipped with metadata descriptions provided by the monist *metadata tool*. This assures finding and reuse of objects stored in the monist-database. Authored material can be submitted to the server for publication under the monist-content license.

3. Educational simulations

Scientific simulations are usually not functional for students. There is a lack of textual information and a clear description of the instructional use or the educational context, a simulation is intended for [4, 5]. A lot of examples can be found in the internet. The reuse of hundreds of simulations as applets suffers from this lack. They are rarely equipped with contextual information, are hard to handle and the data-visualizations are not self-explaining. In most

Another crucial feature is the simulations interface design. Simulations often provide unfiltered access to all model parameters, easily more than 10, to allow a wide range of experiments and exploration of the models behaviour. Learners, who are unfamiliar with model simulations, are often unable to cope with this wide range of possible parameter sets. Therefore, the effective reduction of changeable parameters *per page* is an important concern of these educational simulations introduced here. This leads to the concept of restricted model access. For example, this can be achieved by offering a single parameter per page. In this way, the focus lies on this and only this parameter and its influence on the model's behaviour. Combinations of parameter variation can be introduced in an intelligible manner. The model access can get more complicated with each page.

The main advantages of scientific simulations are, in general, visualization possibilities of the models data. Evidently, this is also true for education. To describe the model's behaviour in words is inadequate. Even formulas as such will not give an insight to the way dynamics evolve in this kind of systems. Therefore, an appropriate visualization of the spatiotemporal dynamics is essential for understanding the processes going on in the simulation. The typical educational simulations provided in monist are equipped with several visualizations that can be applied in parallel, e.g. colour-coded spatial representation of the models overall activity (see figure 2), a virtual oscilloscope (allowing for the parallel observation of time courses) or static and dynamic plots. These visualizations can be applied in parallel to support the learners imagination of what happens to signals on different levels of processing.

4. Educational settings

The general disciplines involved in the field monist seeks to serve are neurobiology, psychology and informatics. In biology, series of experiments during practical trainings are one of the most common educational settings taking place in university laboratories with specific experimental setups. They are the most time expensive part of a degree. Other settings, like classical lectures or seminars, are comparatively less time consuming and their main function is restricted to the provision theoretical background. Therefore, first and foremost, using simulations in biology means their integration in practicals, which is, fortunately, the most active and intense educational setting.

The education in psychology and informatics mainly deals with classical lectures, seminars and tutorials. Contrary to the education of biology, the practical aspects of learning are attached to the theoretical background. Compared with the total time involved chalk and talk is still more common than practical learning, like dealing with educational

5. Collection of Contents

Concerning the content, the monist group represents a wide range of disciplines involved in neural- and cognitive sciences. Each participating department represents one topical and one methodological area in its field. This composition ranges from biophysics to neurobiology and cognitive science as well as more formally oriented disciplines like neuroinformatics and robotics. Table 1 gives a general overview.

Topical focus	Methodological focus	Department
Sensory Systems	Experimental Neurobiology	Neurobiology, Bielefeld University
Motor Systems	System Theory	Biological Cybernetics/ Theoretical biology, Bielefeld University
Learning/Behaviour	Experimental Neuroethology	Neurobiology, FU Berlin
Neuronal Coding	Data analysis/statistics	Neurobiology & Biophysics, University of Freiburg
Spatial Cognition/Psychophysics	biol. Robotics, VR	Cognitive Neurobiology, University of Tuebingen
Cognition/Complex Problem Solving	Experimental Psychology	Psychology, University of Bamberg
Neuronal Network Theory	Neuroinformatics & Simulation Techniques	Neuroinformatics, Bielefeld University
Applied Neuronal Networks	»neural engineering«, Robotics	Neuroinformatics, University of Ilmenau
Didactics & Evaluation	Mental Models & Instructional Design	Educational science, University of Freiburg

Table 1: General overview of the topical and methodological fields covered by the members of the monist-group.

Within the scope of the monist-project, 180 ready-made educational simulations have been produced, covering the main disciplines involved. This collection is divided into 8 different more general topics. Table 2 gives a coarse overview of the main contents offered by the monist-system. Apart the category *Methods* every topic describes one specific level of information processing in the brain. The last category deals with those methods that are essential for neuronal modelling.

and will mainly be used in educational context by those departments that designed them for scientific use. Nevertheless, the monist-console offers tutorials and documentations for all users to give them the opportunity of operating these tools in a meaningful manner. All these tools are downloadable at the monist-homepage [3]. Especially for NEO/NST [6] monist is equipped with a plug-in, to view ready-made NEO/NST-simulations like applets. A collection of RUBIN-simulations [7] is integrated as well. All programs mentioned are stand-alone applications and therefore also executable without monist.

Tool	Main purpose	Main topic	GUI	Language	Main users
NEO/NST	Fast prototyping of artificial neural networks	Neuroinformatics	Visual programming	C/C++	Scientist
PSI	Implementation of a theory of human action and cognition	Psychology	Parameter manipulation	Delphi	Scientist, Students
DAS	Artificial Neural network modeller	Psychology	Visual programming	Delphi	Scientist, Students
GraSim	Prototyping of robot control systems	Neural engineering	Visual programming	C++	Scientist
SNNAP	Modelling single neurons & small networks	Neurobiology	System manipulation	Java	Scientists/ Students
RUBIN	Collection of high end educational simulations	Neural information processing	System manipulation	Java	Students

Table 3: Additional tools offered by the monist system.

6. Evaluation

An essential part of the monist-project is the formative and summative evaluation of the use of simulations in higher education. This evaluation is four-fold. It belongs to courses and educational contexts as well as to the educational

For now, the monist group refers to the use of educational simulations, as they are introduced in this paper. It provides a clear structure and can be combined with many different simulation formats, e.g. applets and applications. Units typically combine instructions and explorative elements. Hence, the explorative character of simulations is the ultimate learning objective but is also guided by instructions. Because of the intuitive interfaces used in most of the monist simulations, instructions will become less important as the user's experience during ongoing learning grows. The less intuitive the embedded simulations are the more detailed the instructions should be. This implies that an effective use of the educational simulations lies in the responsibility of authors and teachers, respectively. To strengthen the efficiency, the system offers recommendations for authors and teachers based on the experiences of the monist group. This includes an educational tutorial, sample units and a help system for general system features.

Teachers and authors from other disciplines, who are interested in designing educational simulations for their lectures with monist, are invited to study the available educational simulations for the neurosciences and to use them as an example work. They surely will benefit from the inspiration coming with the educational simulations in monist.

8. Conclusions and Outlook

The monist console is a system for learning and teaching with educational simulations. It also supports the authoring of educational simulations. Several tools are integrated in the system, that allow for comprehensive introduction of educational simulations in the neural- and cognitive sciences. The offered contents can be regarded as a good practice example, demonstrating the integration of simulations in higher education. Even though simulations in the neural and cognitive sciences are highly specific contents, the monist-console readily supports learning with any kind of educational simulations. The system can be adopted by any discipline, using simulations in science and education. Understanding simulation as educational media [5], monist provides a framework for a broader use and integration in education in general, not only in university context.

Acknowledgements